

## ON THE LOCAL SOLUTION METHOD AND THE SOLUTION OF NONLINEAR ORDINARY DIFFERENTIAL EQUATIONS

### Erb Ferreira Lins

Programa de Engenharia Mecânica, COPPE, UFRJ, C.P. 68503, 21945-970  
Rio de Janeiro, Brasil  
erb@mecsol.ufrj.br

### Jerson Rogério Pinheiro Vaz

Universidade Federal do Pará – Departamento de Engenharia de Materiais  
Folha 17, Qd 04, Lote Especial, Nova Marabá, Marabá, PA, 68505-080, Brasil  
jerson@ufpa.br

### Daniel Onofre de Almeida Cruz

Universidade Federal do Pará – Departamento de Engenharia Mecânica  
Av. Augusto Correa, s/n - Belém, PA, 66075-900, Brasil  
doac@ufpa.br

**Abstract.** *In this work, a new numerical method for solution of ordinary differential equations is presented. This technique uses a simpler ODE form, derived from the original ODE by the assumption that, in a small region, some terms could be approximated by a constant value or by a straight line. The new ODE could be integrated analytically and could be used to obtain the value of the function and derivatives to the next step, resulting in piecewise continuous and smooth solutions. This method will be compared to the classical 4<sup>th</sup> order Runge-Kutta method and analytical solutions (where available) in some differential equations. The results show very good agreement, even for larger steps size.*

**Keywords.** *Ordinary differential equations, numerical methods, linearization techniques.*

### 1. Introduction

The development of numerical methods to solve ordinary differential equations has become, in the past decades, a very intense field of research (Butcher, 2000). In spite of the wide range of applicability of the most used methods, such as Runge-Kutta and others (Butcher, 2000, Iavemaro and Mazzia, 1998, Shawagfeh and Kaya, 2004), there are still some situations where those methods, are not adequate or simply diverge, for certain values of the step size. The principal characteristic of most of the above cited methods is that, once the value of a function at a certain point  $y_i$  is known, the estimate of the next point  $y_{i+1}$  is obtained through an straight line approximation. The slope of the straight line is calculated to minimize the error. It is well known that the above described approach does not furnish good results for stiff problems and, of course, is unstable when a coarse mesh is used. That type of problems appears to be an intrinsic characteristic of the numerical procedure, since the solution of the ordinary differential equation is locally approximated by a straight line.

An alternative procedure that can be used to improve the performance of the numerical methods is to search for locally valid, more precise, solutions (Ramos, 2005, Ramos and García-López, 1997). Those solutions must be able to predict, in a small region of the domain, the approximate behavior of the solution of the differential equation. As a first guess, a polynomial function could be used to obtain the required approximation. But a polynomial function is not in fact a truly local approximation from the numerical methods point of view, since; the higher the degree of the approximation, higher is the number of points that must be used to obtain the derivatives used in the Taylor series expansion.

Generally, the differential equations solved by numerical methods, are very complex non linear expressions, and possess no analytical solutions. But, considering that the idea is to find a local expression that furnishes a behavior closer to the exact solution than a straight line, it is not necessary to solve the original differential equation but just an “approximate” differential equation, obtained from the “original” differential equation. That equation must be more amenable to mathematical treatment than the original one. In other words, the local “approximate” equation must be chosen in such a way that an analytical solution can possibly be found.

This work proposes a new numerical method for solution of ordinary differential equations. The main idea is to formulate a locally valid approximate differential equation, in order to obtain a locally valid approximate analytical solution. The solutions procedure and some applications of the method are described in the following sections. Two special cases with specially chosen differential equations are chosen to describe the characteristics of the method. In the first case, a comparison between the proposed method and the Runge-Kutta method is made. It is show that the present

approach is more stable, and provides better results, even for large values of the step size. In the second example the Blasius equation is solved by two different formulations of the present method in order to show its flexibility.

## 2. Description of the method

The local solution method consist in find an locally approximate valid analytical solution which can be used to evaluate the function value (solution of the equation) at the next step of the calculation. The approximate solution is in fact the solution of an approximate equation which is obtained from the original differential equation as described below.

Consider an ordinary non linear differential equation:

$$f_{ap}^{(N)} = G\left(f_{ap}^{(N-1)}, \dots, f_{ap}^{(1)}, x\right) \quad (1)$$

An approximate equation can be constructed from Eq. (1) by considering some of parameters of the argument of the  $G$  function as constants at some small interval of the domain:

$$f_{ap}^{(N)} = G\left(f_{ap_o}^{(N)}, f_{ap}^{(N-1)}, f_{ap_o}^{(N-1)}, \dots, f_{ap}^{(1)}, f_{ap_o}^{(1)}, x, x_o\right)$$

where  $f_{ap}^{(N)}$  represents the  $n^{\text{th}}$  derivative of the approximate function. The parameters with the subscribed  $O$  are assumed to be constants and their values are obtained from the calculations preformed at the previous step. In order to clarify the procedure an example can be used:

The Duffing equation,

$$\frac{d^2 f}{dx^2} + k_1 f + k_2 f^3 = 0 \quad (2)$$

With the boundary conditions

$$\begin{aligned} f(0) &= f_0 \\ f'(0) &= f'_0 \end{aligned}$$

is a non linear differential equation, with no analytical solution (so far). That equation can locally be linearized by the following procedure:

I - In a small neighborhood near  $x=0$ ,  $f(x)$  is not very different from  $f_0$  ;

II - The non linear term in Eq. (2) is  $k_2 f^3$ . That term can be approximated (linearized) using the statement I as follows:

$$k_2 f^3 \cong k_2 f f_0^2 \quad (3)$$

III - Inserting expression (3) into Eq. (2) we obtain:

$$\frac{d^2 f}{dx^2} + k_1 f + k_2 f f_0^2 = \frac{d^2 f}{dx^2} + f(k_1 + k_2 f_0^2) = 0 \quad (4)$$

Equation (4) is a linear equation with a analytic solution. The solution of Eq. (4) is not very different of the solution of Eq. (2) at least in the neighborhood of  $x=0$ . Eq. (4) can now be solved and used to evaluate the next step value of  $f$  at  $x = \Delta x$ . The derivative of  $f$  at  $x = \Delta x$  can be obtained analytically, (since we have a simple function) and now  $f|_{\Delta x}$  and  $df/dx|_{\Delta x}$  can be used as boundary conditions for the next step. The approach can be generalized for any step of  $x$  by the equation below,

$$\frac{d^2 f_{N+1}}{dx^2} + f_{N+1}(k_1 + k_2 f_N^2) = 0 \quad (5)$$

With the boundary conditions:

$$f(x_N)_{N+1} = f_N$$

$$\left. \frac{df(x)_{N+1}}{dx} \right|_{x_N} = \left. \frac{df(x)_N}{dx} \right|_{x_N}$$

Equation (5) is valid for  $x_N = N\Delta x < x < x_{N+1} = (N+1)\Delta x$ . An improved version of Eq. (5) can be obtained if the mean value of  $f$ , calculated at  $x = x_N + \Delta x/2$  is used according to the following expression:

$$\frac{d^2 f(x)_{N+1}}{dx^2} + f(x)_{N+1} \left( k_1 + k_2 \left( \frac{f_N + f_{N+1}}{2} \right)^2 \right) = 0$$

The boundary conditions remain the same.

### 3. Comparison with others Numerical Methods

In order to evaluate the performance of the Local Solution Method (LSM) two equations are chosen to be numerically solved. A 4<sup>th</sup> order Runge-Kutta (RK4) method will be used and the results will be compared with the LSM approach. The 4<sup>th</sup> order Runge-Kutta is a very popular method, due to its relative simple numerical implementation and accuracy. There are some others implicit procedures, like Adams-Bashforth-Moulton or BDF (backward differentiation formula) methods, but they require some evaluations using explicit methods to start. As long as we are interested in a method that could give good results in largers steps and stiff problems, the use of implicit methods should not give good results (Press *et al.* 1992, Heath, 2002).

Nevertheless, there are some situations where the RK4 method does not furnish good results. One example is the following equation:

$$\frac{df(x)}{dx} = 1 - 4f(x)^2$$

$$f(0) = 0$$
(6)

Equation (6) was chosen because, despite its apparent simplicity, the RK4 method fails to solve it for some values of the step size. Besides that, this equation has an analytical solution given below:

$$f(x) = \frac{1}{2} \sin \left( -\frac{\pi}{2} + 2 \arctan \left( e^{2x + \frac{5\pi}{4}} \right) \right)$$
(7)

The LSM furnishes the following linearized equation for each  $\Delta x$  step:

$$\frac{df_{N+1}(x)}{dx} = 1 - 4 \frac{(f_N + f_{N+1})}{2} f_{N+1}(x)$$

$$f_{N+1}(N\Delta x) = f_N$$
(8)

The analytical solution of eq. (8) is:

$$f_{N+1}(x) = \frac{1 - e^{-4 \frac{(f_N + f_{N+1})}{2} (x+c)}}{4 \frac{(f_N + f_{N+1})}{2}}$$

with  $c = -N\Delta x - \ln \left( \frac{1 - 4 \frac{(f_N + f_{N+1})}{2} f_N}{4 \frac{(f_N + f_{N+1})}{2}} \right)$

(9)

To calculate the function  $f(x)$  at  $x = (N+1)\Delta x$ , it is necessary to use the Eq. (9) twice. First to initially evaluate  $f_{N+1}$ , this calculation is performed assuming  $f_{N+1} = f_N$ . Then, equation (9) is used, now using the previous obtained estimate of  $f_{N+1}$  to calculate the mean value of  $f$  on the interval, in order to obtain a better approximation of the linearization coefficient.

The results of the calculation using the LSM are compared with the RK4 and the analytical solution, for several of the step size values. Figure (1) show the comparison for  $\Delta x = 0.9$ , for this step size the RK4 diverges. Although the LSM does not give very accurate results at the initial steps, the results obtained with the proposed method are by far superior to the RK4. The LSM not only does not diverge but also gives the correct asymptotic value of the function.

A more tight mesh is used in Fig. (2) ( $\Delta x = 0.7$ ) in this case the RK4 exhibits a strange behavior, the calculations does not diverge, but the asymptotic values furnished by the method is completely wrong. Again the solution obtained by the LSM is much better. At this point it is important to call the attention to the fact that the LSM is very stable, due to the fact that the calculations are performed using actually, an expression that is locally very near of the exact solution, this approach stabilizes the process and is also very useful if large values of the step size must be used.

Figures (3) to (5) show the calculations for  $\Delta x = 0.6$ ,  $\Delta x = 0.5$   $\Delta x = 0.1$  respectively, as expected the results of the RK4 are greatly improved as the step size gets smaller. It is worth to mention that for small values of the step size, the results of the RK4 are more accurate than the LSM, as it can be seen in the Tables (1), (2) and (3). The accuracy of the LSM is not established for this case but Tab. (1) show that it should less than  $\Delta x^3$ .

Table 1 - Values obtained for a step  $\Delta x = 0.1$ .

X	YPM Proposed Method	YAS Analytical solution	YRK Runge-Kutta	Error (YPM-YAS)	Error (YRK-YAS)
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.100000	0.099009	0.098688	0.098686	0.000321	0.000001
0.200000	0.190522	0.189974	0.189971	0.000548	0.000003
0.300000	0.269166	0.268525	0.268519	0.000642	0.000006
0.400000	0.332640	0.332018	0.332009	0.000621	0.000010
0.500000	0.381331	0.380797	0.380785	0.000534	0.000012
0.600000	0.417251	0.416827	0.416813	0.000423	0.000014
0.700000	0.442994	0.442676	0.442662	0.000318	0.000014
0.800000	0.461065	0.460834	0.460821	0.000230	0.000013
0.900000	0.473566	0.473403	0.473391	0.000163	0.000012
1.000000	0.482127	0.482014	0.482004	0.000113	0.000010
1.100000	0.487949	0.487872	0.487864	0.000077	0.000008
1.200000	0.491890	0.491837	0.491831	0.000053	0.000006
1.300000	0.494549	0.494514	0.494509	0.000036	0.000005
1.400000	0.496340	0.496316	0.496312	0.000024	0.000004
1.500000	0.497544	0.497527	0.497525	0.000016	0.000003
1.600000	0.498352	0.498341	0.498339	0.000011	0.000002
1.700000	0.498895	0.498887	0.498886	0.000007	0.000002
1.800000	0.499259	0.499254	0.499253	0.000005	0.000001
1.900000	0.499503	0.499500	0.499499	0.000003	0.000001
2.000000	0.499667	0.499665	0.499664	0.000002	0.000001

Table 2 - Values obtained for a step  $\Delta x = 0.5$ .

X	YPM Proposed Method	YAS Analytical solution	YRK Runge-Kutta	Error (YPM-YAS)	Error (YRK-YAS)
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.500000	0.393502	0.380797	0.373352	0.012705	0.007445
1.000000	0.484081	0.482014	0.461302	0.002067	0.020712
1.500000	0.497812	0.497527	0.487237	0.000284	0.010291
2.000000	0.499703	0.499665	0.495751	0.000039	0.003914
2.500000	0.499960	0.499955	0.498584	0.000005	0.001371
3.000000	0.499995	0.499994	0.499528	0.000001	0.000466
3.500000	0.499999	0.499999	0.499843	0.000000	0.000157
4.000000	0.500000	0.500000	0.499948	0.000000	0.000052

Table 3 - Values obtained for a step  $\Delta x = 0.7$ .

X	YPM Proposed Method	YAS Analytical solution	YRK Runge-Kutta	Error (YPM-YAS)	Error (YRK-YAS)
0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.700000	0.446103	0.442676	0.381834	0.003427	0.060842
1.400000	0.496249	0.496316	0.379132	0.000067	0.117183
2.100000	0.499770	0.499775	0.377020	0.000006	0.122755
2.800000	0.499986	0.499986	0.375399	0.000000	0.124587
3.500000	0.499999	0.499999	0.374175	0.000000	0.125824
4.200000	0.500000	0.500000	0.373261	0.000000	0.126739

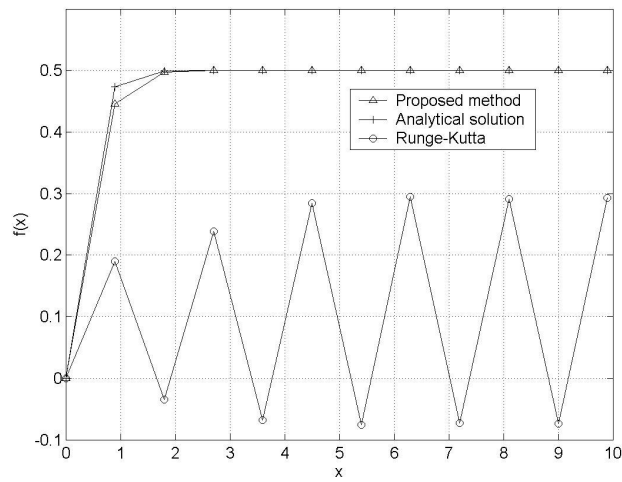


Figure 1 – Result obtained using a step  $\Delta x = 0.9$ . Observe that the Runge-Kutta method fails abruptly. The proposed method converges, even for this large step.

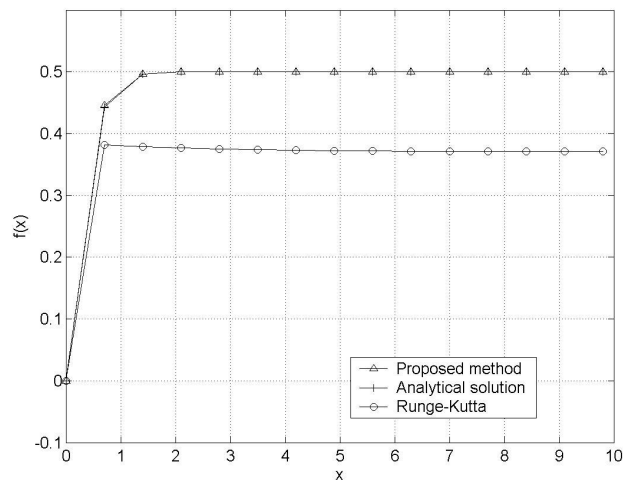


Figure 2 – Obtained result using a step  $\Delta x = 0.7$ . In this case the Runge-Kutta method fails.

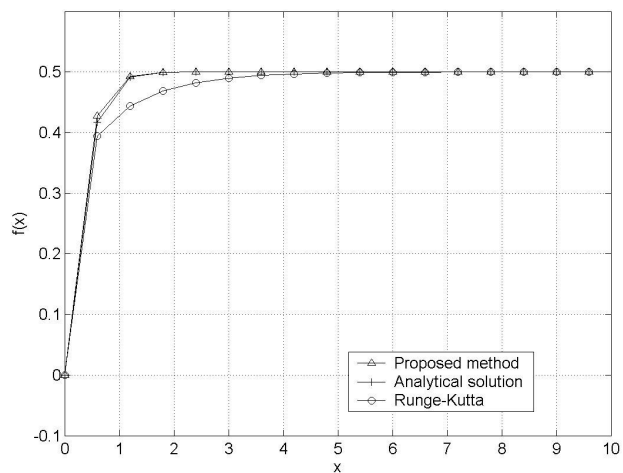


Figure 3 – Result obtained using a step  $\Delta x = 0.6$ . The Runge-Kutta method tends to get better.

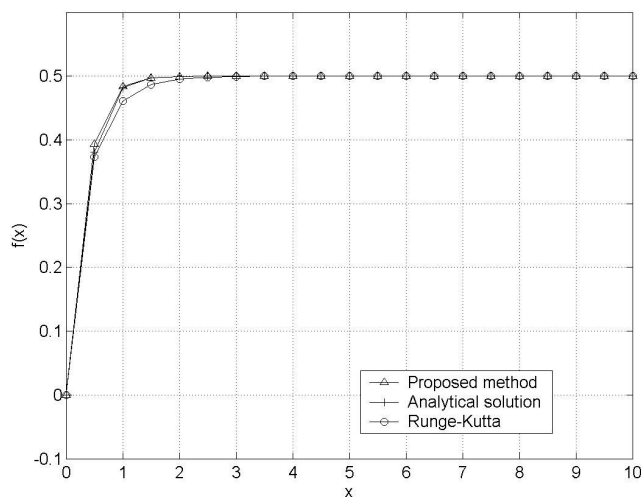


Figure 4 – Result obtained using a step  $\Delta x = 0.5$ . Here, the Runge-Kutta method presents good proximity of the analytical solution; however, the proposed method presents better convergence.

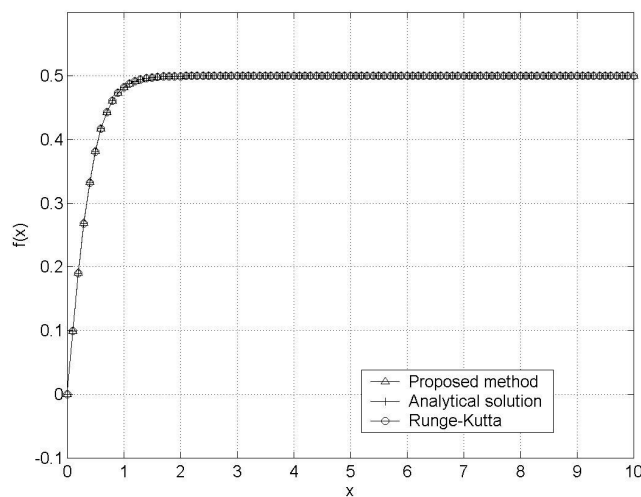


Figure 5 – Result obtained using a step  $\Delta x = 0.1$ . The solutions are practically coincident. The Runge-Kutta method has smaller error (see Table 1), but the proposed method presents a quite good approximation.

The accuracy is strongly connected with the solution of local approximate equation. The second test case is used to analyze that relation. The Blasius equation is a well know non linear ordinary third order differential equation. This equation with the boundary conditions is presented below.

$$\frac{d^3 f}{d\eta^3} - f \frac{d^2 f}{d\eta^2} = 0 \tag{10}$$

With the boundary conditions:

$$\begin{aligned} f(0) &= 0 \\ \left. \frac{df}{d\eta} \right|_{\eta=0} &= 0 \\ \left. \frac{d^2 f}{d\eta^2} \right|_{\eta=0} &= f_0'' \end{aligned} \tag{11}$$

Usually the third boundary  $f_0''$  condition is to consider that the derivative of the function asymptotically approaches 1 as  $\eta \rightarrow \infty$ . But since the value of the second derivative at the wall is well established, and that the two boundary conditions are in fact equivalent, the above boundary conditions are used here.

We consider here two different approximations to the Blasius equation. The first one is very similar to previous approach, used in Eq. (6). In this case, we consider that the values of function  $f(\eta)$  in a small region can be approximated using a constant value. Then, the Blasius equation can be rewritten as:

$$\frac{d^3 f}{d\eta^3} - f_N \frac{d^2 f}{d\eta^2} = 0 \tag{12}$$

And the boundary conditions in a point  $\eta_N = N\Delta\eta$  in the domain

$$\begin{aligned} f(N\Delta\eta) &= f_N \\ \left. \frac{df}{d\eta} \right|_{\eta=N\Delta\eta} &= f_N' \\ \left. \frac{d^2 f}{d\eta^2} \right|_{\eta=N\Delta\eta} &= f_N'' \end{aligned} \tag{13}$$

If  $f_N$  is a constant value, the Eq. (12) has the following solution:

$$f(\eta) = \frac{e^{-f_N\eta} + C_1}{f_N^2} + C_2\eta + C_3 \tag{14}$$

Applying the boundary conditions given by Eq. (11) results:

$$f = \frac{f_N''}{f_N^2} \left( e^{-f_N(\eta-\eta_N)} - 1 + f_N(\eta-\eta_N) \right) + f_N'(\eta-\eta_N) + f_N \tag{15}$$

Where  $N\Delta\eta \leq \eta \leq (N+1)\Delta\eta$ .

In an alternative approach, we consider that the values of  $f(\eta)$  could be approximated using a straight line. The Blasius equation could be written as

$$\frac{d^3 f}{d\eta^3} - (k_1 + k_2\eta) \frac{d^2 f}{d\eta^2} = 0 \tag{16}$$

The solution is:

$$\begin{aligned} f(\eta) = \frac{1}{2k_2^{3/2}} \left\{ e^{-(k_1+k_2\eta)^2/2k_2} \left[ 2\sqrt{k_2} e^{k_1^2/2k_2} \left( f_N'' e^{k_1\eta_N + k_2\eta_N^2/2} + e^{k_1\eta + k_2\eta^2/2} \left( k_2 [f_N'(\eta-\eta_N) + f_N] - f_N'' \right) \right) \right] + \right. \\ \left. \sqrt{2\pi} (k_1 + k_2\eta) f_N'' e^{((k_1+k_2\eta)^2 + (k_1+k_2\eta_N)^2)/2k_2} \left[ \operatorname{erf} \left( \frac{k_1 + k_2\eta}{\sqrt{2k_2}} \right) - \operatorname{erf} \left( \frac{k_1 + k_2\eta_N}{\sqrt{2k_2}} \right) \right] \right\} \end{aligned} \tag{17}$$

And the boundary conditions given by Eq. (13).

Some numerical tests are used to check the convergence properties of these two formulations. The analytical solution to Blasius equation is not available, so we solve this equation numerically with a very refined mesh and a reliable solver (MATLAB function ‘ode113’ (MathWorks, 2006, Ashino, 2000)) and consider this result as the correct solution.

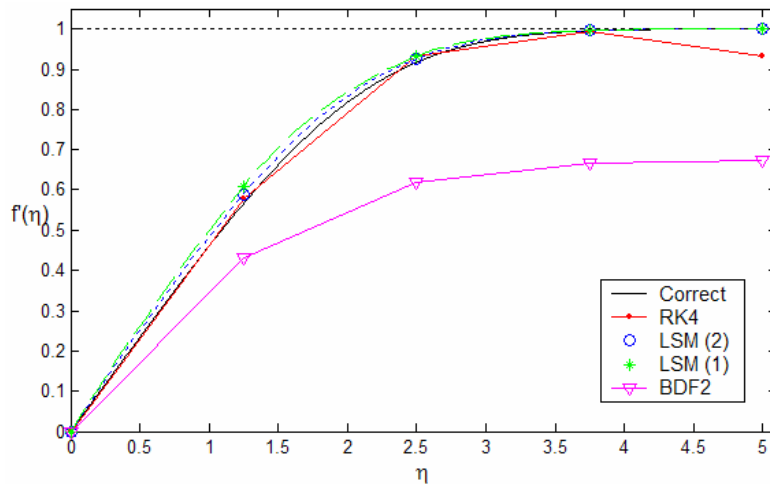


Figure 6 – Results obtained for a very coarse mesh

Figure (6) shows the result for the second derivatives of Blasius equation using the LSM, Runge Kutta and BDF methods (Heath, 2000). The Backward Differentiation Formulas (BDF) is an implicit, multi-step and very stable method in larger steps sizes. We used here Newton iteration to find the solution of 2<sup>nd</sup> order backward corrector and forward Euler approximation for the first step in BDF2 solution. LSM (1) represents the approximations using Eq. (15) and in LSM (2) we used Eq. (17). Several points should be noted:

- a) The LSM shows better accuracy than RK4 and BDF2 for this coarse mesh. The BDF2 suffers from the poor approximation of first step, resulting in a totally wrong approximation, and RK4 is not stable as  $\eta \rightarrow \infty$  ;
- b) As expected, the LSM using a line in the interval of domain gives a better approximation;
- c) The approximation of LSM is, in fact, a piecewise continuous function, so the accuracy between the points of the mesh is much better than RK4 and BDF2.
- d) It approaches the correct value of the function at  $\eta \rightarrow \infty$  . In fact, this behavior happens even when the mesh has only 2 points, as depicted in Fig. (7).

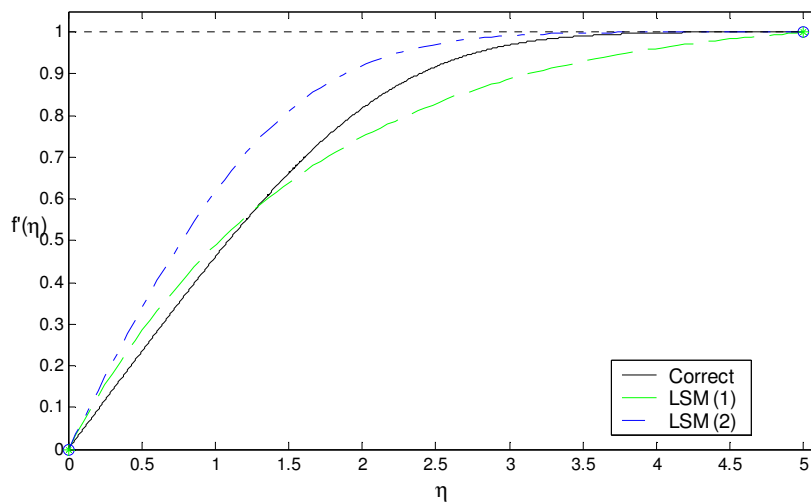


Figure 7 – Approximation using LSM with 1 step. The Runge Kutta does not give reasonable results.

As the number of points in the mesh increases, the Runge-Kutta method gives a better approximation than LSM in each of the points of the mesh (Figure 8). Nevertheless, between the reference points of the mesh the LSM presents a curve that closely follows the behavior of the solution and still gives a better approximation than BDF.



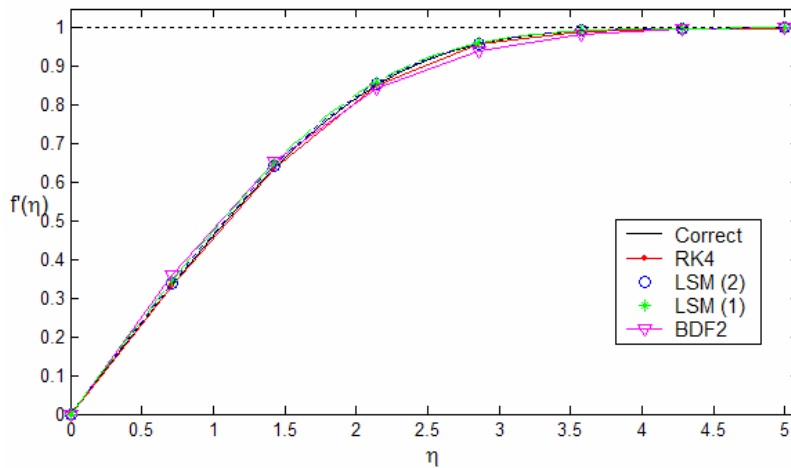


Figure 8 – Results obtained for a refined mesh

Figure (9) shows the relative error of LSM and RK4. It can be noted that the RK4 error presents a faster convergence. For coarser meshes, LSM shows better accuracy than RK4. Anyway, as the size of interval of solution gets smaller, the error of the two methods tends to the same value, showing that this two formulations have the same order of approximation as  $\Delta x \rightarrow 0$ .

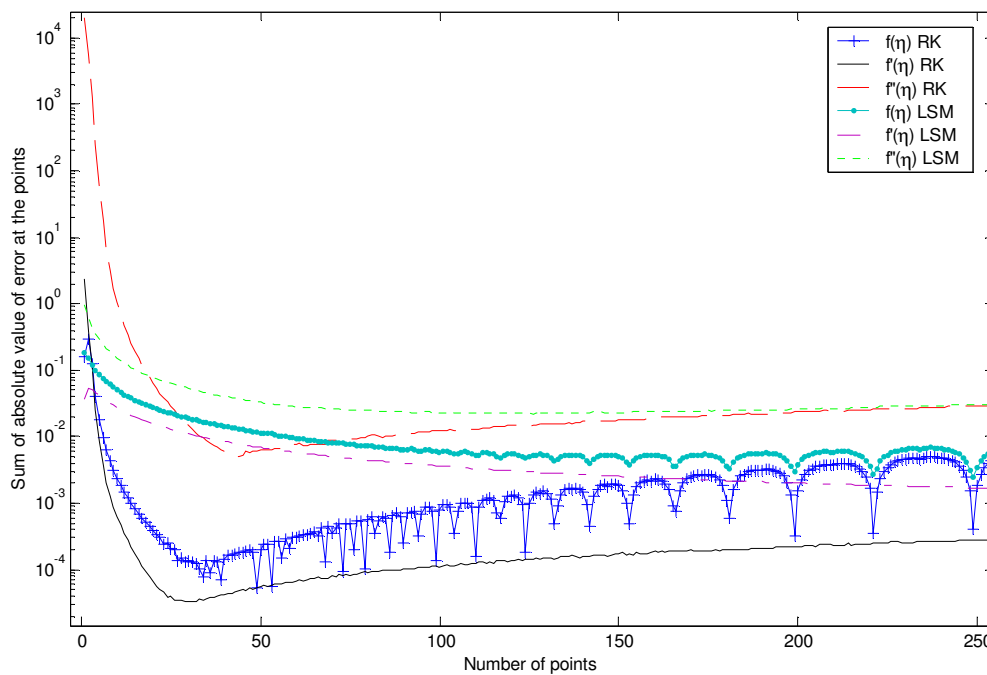


Figure 9 – Relative error of the methods

#### 4. Conclusions

In this work a new technique was developed to the solution of ordinary differential equations. The method is based in a local solution obtained from original differential equation by the assumption that, in a very small region, some terms of differential equation could be approximated by a constant or linearly varying variables. The new differential equation has an analytical solution that closely approximate the function as the size of region of interest tends to zero. This method could be used to a large class of differential equations and has the desirable properties of local smooth piecewise continuous solutions. The results presented here show very good agreement when compared to analytical and numerical solutions. We also show that this method, when applied to Blasius equation, have the same order of error of classical 4<sup>th</sup> order Runge-Kutta when the step size goes to zero and that the functions used to the approximate terms are also of great importance in accuracy of the method. The LSM also shows a very stable and good approximation when

larger step size is used. The application of technique to more complex differential equation, stiff and singular problems, as well as error bounding and a more accurate mathematical description of the method are under development and will be published elsewhere.

## **5. References**

- Ashino, R.; Nagase, M. and Vaillancourt, R., 2000, "Behind and beyond the MATLAB ODE suite", *Comput. Math. Appl.* 40, no. 4, pp. 491-512.
- Butcher, J. C., 1996, "A history of Runge-Kutta Methods", *Applied Numerical Mathematics*, 20, pp. 247-260.
- Butcher, J. C., 2000, "Numerical methods for ordinary differential equations in the 20th century", *Journal of Computational and Applied Mathematics*, 125, pp. 1-29.
- Heath, M. T., 2002, "Scientific Computing, An Introductory Survey", 2nd edition, McGraw-Hill, New York.
- Iavemaro, F. and Mazzia, F., 1998, "Solving ordinary differential equations by generalized Adams methods: properties and implementation techniques", *Applied Numerical Mathematics*, 28, pp. 107-126.
- MathWorks, 2006, Documentation for function ode113 is available online at MATLAB website: <http://www.mathworks.com/access/helpdesk/help/techdoc/ref/ode45.shtml>, last access: 06/01/2006.
- Press, W. H., Flannery, B. P., Teukolsky, S. A. and Vetterling, W. T., 1992, "Numerical Recipes in Fortran", Cambridge University Press.
- Ramos, J. I., 2005, "Linearization techniques for singular initial-value problems of ordinary differential equations", *Applied Mathematics and Computation*, 161, pp. 525-542.
- Ramos, J. I. and García-López, C. M., 1997, "Piecewise-linearized methods for initial-value problems", *Applied Mathematics and Computation*, 82, pp. 273-302.
- Shawagfeh, N. and Kaya, D., 2004, "Comparing Numerical Methods for the Solutions of Systems of Ordinary Differential Equations", *Applied Mathematics Letters*, 17, pp. 323-328.

## **6. Copyright Notice**

The authors are the only responsables for the printed material included in his paper.

## COMPUTATIONAL FLUID DYNAMIC APPLIED TO COMBUSTION CHAMBER FOR SMALL GAS TURBINES USING NATURAL GAS

### Harley Souza Alencar

Federal University of Itajubá, Benedito Pereira dos Santos Av., 1303, ZipCode 37500-903, Itajubá, MG, Brazil  
haarley@terra.com.br

### Marco Antonio Rosa do Nascimento

Federal University of Itajubá, Benedito Pereira dos Santos Av., 1303, ZipCode 37500-903, Itajubá, MG, Brazil  
marco.antonio@unifei.edu.br

### Helcio Francisco Villa Nova

Federal University of Itajubá, Benedito Pereira dos Santos Av., 1303, ZipCode 37500-903, Itajubá, MG, Brazil  
patricia.foroni@itelefonica.com.br

*Abstract. This work focuses the development about a practical scheme using commercial softwares to study the flame behavior (combustion and flow) in annular combustion chambers for small gas turbines to attend its adaptability and economic feasible to use different fuels. For it, it is used the methane gas to represent the natural gas, whose chemical composition is done mainly by 88(%) in volume by methane. Among the softwares: Gatecycle® is applied to calculate the thermodynamic parameters of combustion chamber from thermal cycle for small gas turbine; GASEQ® to calculate the chemical reactions and concentration of reactants and products to esteemed the emissions; and CFX® is used to simulate the dynamic fluid for flow, combustion, heat transfer and emissions. The combustion in CFD is simulated by Eddy Dissipation Model with two steps of chemical reaction for the methane gas. The turbulence is simulated by RNG K-ε Model. The heat transfer by radiation is made by P1 Method and the NOx emission is calculated by Fenimore and Zeldovich Methods. The expected numeric results are the fields of velocity, pressure and temperature, to feature the load losses and the volume of flame. From this way, it intends to make available information about: the parameters that affect the position of flame; the maximum temperature of flame; the temperature of hot gas in exhaustion; the affects from nozzles; the affects of flame from methane gas in design; and the preliminary efficiency value for combustion chamber. Besides, it is presented a typical validation process for CFD in combustion chamber with tri-dimensional domain.*

**Keywords.** CFD, Combustion Chamber, Small Gas Turbine, Natural Gas

### 1. Introduction

The Small Gas Turbines are becoming a promise way for distributed power generation and combined heat and power applications. Its conception is from 40's decade together others gas turbines for aeronautical and navy industry. Later, this system became alternative equipment to generate power and heat to attend the population located in isolated regions out to the interlinked electrical system, because of the Oil Crisis in 1973.

Part of its success is due to the advances in electronics, which allow unattended operation and interfacing with the commercial power grid. Electronic power switching technology eliminates the need for the generator used by gas turbine to be synchronized with the power grid.

These equipments can generate power until 350 (kW) and have the following advantages in relation to piston engine generators, Gurgel et al. (2002): (a) higher power density (with respect to footprint and weight); (b) low emissions; (c) few, or just one, moving parts; (d) its design uses foil bearings and air-cooling operate without oil, coolants or other hazardous materials; (e) its operation is simply; (f) its installation and maintenance are easy; and (g) its operation cost is low. However, piston engine generators are quicker to attend the changes in output power requirement.

Besides, the emission control of pollutant agents, such as NO<sub>x</sub>, CO and SO<sub>x</sub>, is adjusted by U.S. Environment Protection Agency (1993), where it is set that the maximum NO<sub>x</sub> emission for small gas turbines is 15 ppmv (15% Oxygen) for the flame temperature equal to 1900 [K], Lefebvre (1995).

Currently, the main manufactures of small gas turbines are: Turbo-Genset, Capstone Turbine Co., Solar Turbines, Allison Engine, Bowman Power, Turbec, Honeywell, Iliott MagneTek Power System, General Electric, Turbomeca, ABB, Harvester Company and Rolls Royce Nimbus.

The small gas turbines are based on Ideal Brayton-Joule's Cycles, which is a thermodynamic cycle where air is compressed isentropically in compressor, the combustion occurs at constant pressure in combustion chamber, and the expansion for hot gas over the turbine occurs isentropically until the back to the starting pressure, Lefebvre (1983).

Among the parts from small gas turbine, the combustion chamber is focused in earlier researches, mainly about the potential improvements in combustion efficiency, which has two basic factors that can limit its operation range: (a) friction losses; and (b) thermal limitations for blades in turbine.

The friction losses occur in bearings, in the blades and the metallic walls in compressor, turbine and nozzles in combustion chamber. Due to this friction, the small gas turbines need a compressor to supply more pressure. In case for annular chambers, the compressor is radial, while the compressor is axial for tubular chambers, Lefebvre (1983).

On the other hand, it is possible to compensate these losses using more fuel flow, although it can provoke higher temperatures than thermal limitations for blades in turbine, as well as the creep.

The main improvements that are done for small gas turbines are: (a) higher compression ratios; (b) higher temperature gradients in combustion chamber; (c) more efficient combustion; (d) better cooling of engine parts; and (e) reduced emissions. For it, two ways are applied: (a) the analysis of geometry; and (b) the development of fuels more reactive with low emissions.

The first way has been explored in relation to change of geometry, where can be sign the Dry Low NO<sub>x</sub> - DLN chambers, which are the cylindrical combustion chambers with big quantity of holes in Liners and dilution holes in downstream of combustion chamber, Allen (1998), Nickolaus et al. (2002) and Wakabayashi et al. (2002).

The second way has been studied in relation to the use of fuels without sulfur with high reactivity, like as, the natural gas, according to Lefebvre [16], Melick et al (1999), Vandebroek et al. (2003) and Williams (1985).

Allen (1998) explored the DLN chambers using other kind of fuel with liquid phase and generated by emulsion process. Vandebroek et al. (2003) explored the affects of chemical kinetic in combustion of methane with 4 steps to esteem the temperature for auto ignition. Lefebvre [15] and Melick et al. (1998) show other way to control the NO<sub>x</sub> emission based on the catalysts with NH<sub>3</sub>, as well as its possible effects in environment.

In general, all of results can be obtained by experimental and numerical tests. While the first, it is necessary to raise enough money with expensive equipments and benches, the numerical tests have been a promise alternative with the advantage use of more quick computers with high capacity to save data, Kuo (1986) and Keating (1993).

In relation to numerical tests, different numerical methods have been tested to simulate the flame for combustion chambers. Among these methods, the Computational Fluid Dynamic (CFD) has more applications

The use of CFD for combustion chambers is not early and it has presented satisfactory results in some models, such as, tubular combustion chambers, Gosselin *et al.* (2000), Lee *et al.* (1990) and Nicklaus *et al.* (2002), which executed the first simulations using CFD in cylindrical combustion chambers in bi-dimensional and three-dimensional geometry models. The turbulence models tested by them are K- $\epsilon$ , RNG K- $\epsilon$  and Reynolds Stresses Model (RSM), and the combustion models Fast Chemistry with PDF Model and Flamelet Laminar Model (FLM).

Besides, the works from Fuller and Smith (1994), Hamer and Roby (1997) introduced others combustion models, such as, Eddy Dissipation Model (EDM), with models of NO<sub>x</sub> emission of Fenimore and Zeldovich, in three-dimensional geometry models. In particular, Hamer and Roby (1997) archived satisfactory results with simulations until 4 steps for chemical reactions using methane gas.

Based on the satisfactory results for CFD from studies about tubular combustion chambers, new works has been developed too for the annular combustion chambers with the function to solve its particular problems, such as, low efficiency, high emission of pollution, and to check its economic feasibility.

Hence, considering the importance of CFD application in thermal-aerodynamics problems involving the combustion and its complex phenomena for annular combustion chambers, this work presents a methodology to guide researchers and engineers in the analysis of the flame behavior in annular combustion chamber for small gas turbines using natural gas by CFD. The tested geometric model uses an alternative fuel, the methane gas, once that its participation in chemical composition for natural gas is bigger than 88 (%) in volume. The adopted annular combustion chamber operates with conventional fuels, such as, diesel and kerosene oil, whose power can be until 50 (KW) at ISO conditions, according to the manufacturer of small gas turbine called Solar Turbines. The adopted chemical reactions use interactive steps based on the Minimization Principle of Free Gibbs Energy or Lewis – NASA Method, Svehla and McBride (1973).

The knowledge about the flame behavior is achieved by the establishment of the fields of velocity, pressure and temperature, to describe the load losses and the volume of flame using the mathematical approximations, such as: of turbulent flow; of combustion; of heat transfer by radiation; and NO<sub>x</sub> and CO emissions.

## 2. The Combustion Chamber Tested

The tested combustion chamber is part of small gas turbine of kind Titan T-62T-32 Turbo shaft, 1-stage centrifugal compressor with annular profile, 1-stage turbine designed to work with diesel, kerosene and JP-4 oil according to the manufacturer Solar Turbines, whose main operation conditions are, Alencar et al (2005):

- (a) Fuel pressure in inlet of nozzle: 275790 (Pa);
- (b) Environment temperature and pressure in ISO condition: 25 (°C) and 1 (atm);
- (c) Mass flow of fuel for 6 nozzles: 29 (kg/s);
- (d) Fuel consumption 0.85 (kg / MWh);
- (e) Pressure, temperature and mass flow of air in outlet from compressor, respectively: 206843 (Pa), 52 (°C) and 0.75 (kg/s);
- (f) Maximal temperature of gas in outlet of combustion chamber: 637.9 (°C); and
- (g) Maximal temperature of gas in exhaustion after turbine: 493.2 (°C).

In combustion chamber, the primary air is mixed with fuel in nozzle with high pressure and velocity. Depend on its capillarity and viscosity, the fuel can be drafted by air, due to the decrease of relative pressure inside of nozzle, which is classified like as Venturi Nozzle. Figure 1 shows the equipment that is modeled.

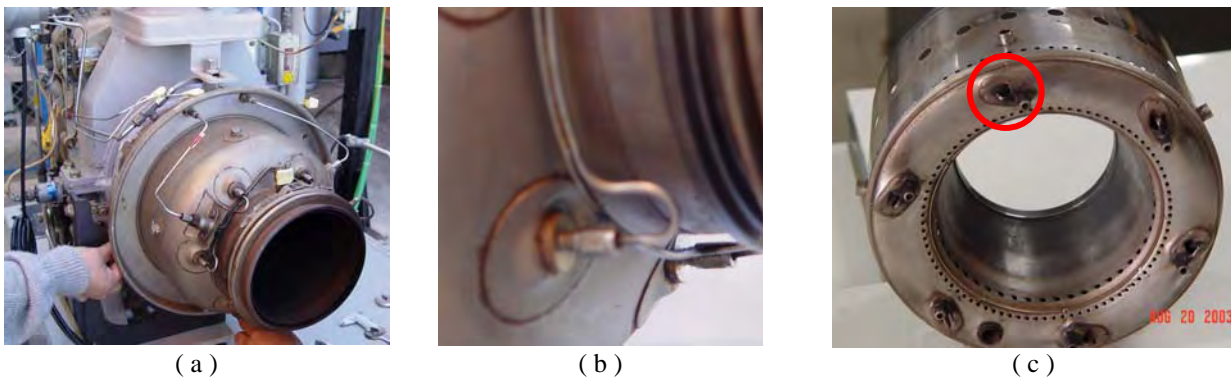


Figure 1. (a) Small gas turbine; (b) External detail of nozzles; and (c) Combustion chamber with 6 nozzles (one of them is shown by red cycle), Alencar et al (2005)

Due to the inclination of the nozzles in relation to the axial axle of combustion chamber, it is generated a secondary flow with rotation. From this way, the time residence tends to be big in annular combustion chamber, which permits to improve the combustion efficiency.

This annular combustion chamber has 6 nozzles with inclination equal to 60 grades in relation to the axial axle of combustion chamber. This model has 86 primary holes and 46 secondary holes to the inlet of air that is used to the combustion and to the dilution and cooling of hot gas. The flow in nozzles obeys the basic principle of jets, like as it is shown in Fig. 2.

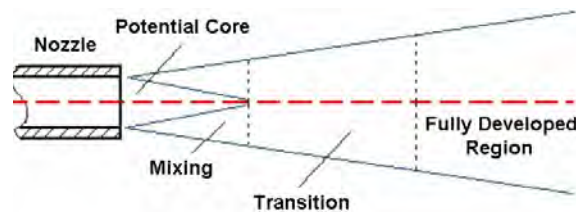


Figure 2 – Main regions of jets, Chigier (1983)

From Fig. 2, it is identified from jet: (a) Potential Core Region, where the air / fuel mixture have a jet that is not affected by environment; (b) Mixing Region, where the jet have load loss and air / fuel mixture presents a diffuse flow; (c) Transition Region, where the air / fuel mixture archives typical velocities for turbulence and it is affected by secondary flow in environment; (d) Fully Developed Region, where jet losses pressure and the flow is mixed with the secondary flow.

Considering the flow from lateral holes in Liner, a big recirculation is created near to the nozzles. From this, the flame can be anchored near to nozzles. Depend on the air / fuel reason, the flame can be classified as Diffusive Flame or Premixed flame. It is recommended to test different combustion models to identify the kind of flame, according to Louis et al. (2001).

The conventional fuels diesel and kerosene oil used by the small gas turbine have Lower Caloric Value equal to 10900 (Kcal/kg) and 10232 (Kcal/kg), respectively. The density for diesel and kerosene oil are 849.2 (kg/m<sup>3</sup>) and 819.2 (kg/m<sup>3</sup>), respectively. These fuels can have auto ignition in function of temperature, according to Fig.3, Lefebvre (1983).

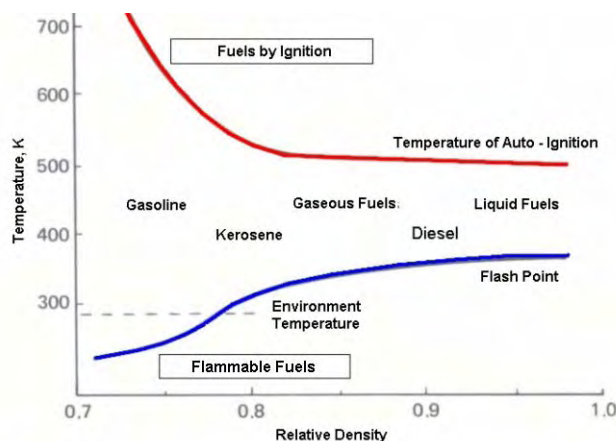


Figure 3 – Typical auto ignition for fuels in relation to relative density and the temperature, Lefebvre (1983)

From Fig. 3, it is possible to identify that in environment temperature, the auto ignition for fuels is not occurs. The temperature in that the fuels are flammables defines the Flash point. In this point, the fuel only can continue the combustion with the action of flash. This point is near to the environment temperature. From this, diesel can have auto ignition in temperature 500 (K), but it is flammable in temperature 350 (K), while the kerosene oil has auto ignition in temperature 630 (K), but it is flammable in environment temperature.

On the other hand, emission is the last step from combustion process and can be characterized by different types of substances, which cannot have more reactivity enough to interact with others, constituting the non-combustion species. Depending on the concentration of these species, the same can provoke damages to the environment. Table 1 shows the typical agents of pollution from emission at small gas turbines and its affects, using diesel and kerosene oil.

Table 1 – Typical emissions and its affects for environment and man, Lefebvre (1995)

Pollutant	Effect
Monoxide of Carbon – CO	Toxic.
Unburned Hydrocarbon - UHC	It contributes for the greenhouse effect in the urban region.
Particles of Carbon – C	It is visible and can provoke respiratory problems.
Oxides of Nitrogen - NOX	Toxic. It is precursory of chemical smoke and the ozone depletion in the atmosphere.
Oxides of Sulfur – SOX	Toxic and Corrosive.
Dioxides of Carbon – CO <sub>2</sub>	It contributes the greenhouse effect in the atmosphere.

### 3. Thermodynamic Calculation

For the development of the thermodynamic calculations of the tested combustion chamber, it is used the program GateCycle® from GE Electronics, which permits to calculate the parameters from thermal cycle of small gas turbine considering the following conditions for Thermal System Laboratory at NEST: a) mechanic power 51.01 (+/-1) (kW); b) Heat Rate 37617.49 (kJ/kWh); and c) Temperature of exhaustion 749 (K). The reference work fuel is natural gas in environment temperature 22 (°C) and pressure 913 (mmHg). Figure 4 presents the characteristics of each element that it composes the small gas turbine.

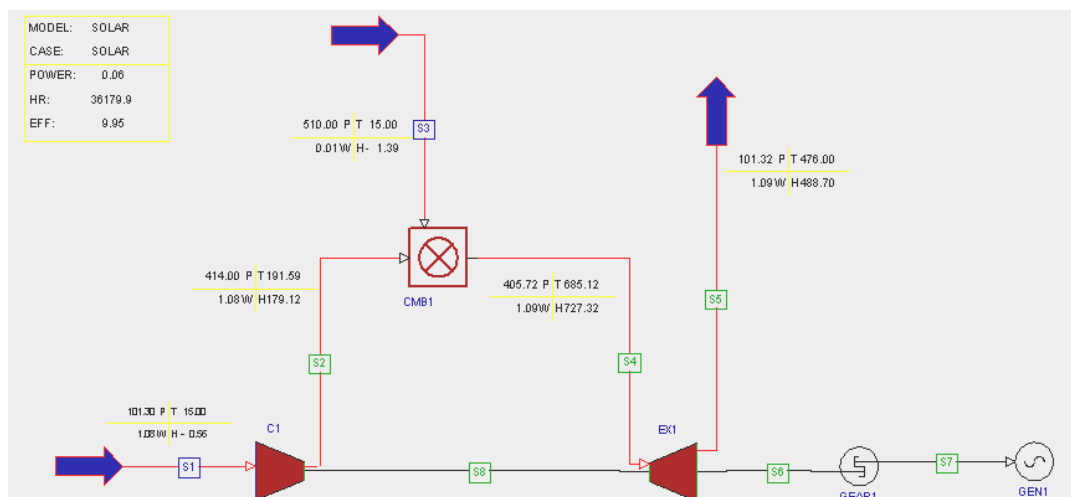


Figure 4 – Typical scheme from GateCycle® for small gas turbine tested

In Fig. 4, the temperature and the pressure are calculated for all point, based on thermal balance in an interactive way. Others parameters, such as: heat loss and pressure loss are obtained by references Borbely et al. (2001) and Lefebvre (1983).

Tab. 2 shows the final values calculated by GateCycle® for control points from S1 to S5 (see Fig. 4) in relation to Compressor, Combustion Chamber and Turbine.

Table 2 – Main values from GateCycle® for Small Gas Turbine

Generator Terminal Power	Stream	Temperature (°C)	Pressure (kPa)	Flow (kg/s)	Enthalpy (kJ/kg)
51.01 kW	S1	22	91.63	0.95609	6.54
	S2	199	368.14	0.95609	188.32
	S3	22	510.00	0.01144	14.29
	S4	685.08	360.78	0.96754	736.22
	S5	478.50	91.63	0.96754	497.52

In Tab. 2, the values calculated for control points from S2 to S4 can be used to define the Boundary Conditions for CFD.

#### 4. Methodology for Solution

The methodology consists to solve the physical problem relative to the behavior of flame in a model of annular combustion chamber for a small gas turbine, using CFD, which represents a part of science for Numerical Methods based on Eulerian Method known as Finite Volume Method to solve the Conservative Equations of Continuous, Momentum and Energy in differential way.

For this method, the physical domain is discretized by volume elements of type tetrahedral or quadrilateral, whose geometries can be structured or not, i.e., the physical greatness is distributed in center of these elements, which permits to realize detailed analysis in geometrical model. The computational tool is the CFX® v 5.7 from ANSYS, which is used to define the three basic steps for the modeling: pre-processing, processing and post-processing.

In Pre-processing, the geometry model is generated from annular combustion chamber, which corresponds to 1 / 6 from original volume of chamber, because there are 6 nozzles and it is preserved the symmetry in relation to the axle of chamber. According to Gosselin *et al.* (2000), this modeling can get satisfactory results because of the symmetry, not considering the affects from heat transfer by convection.

Figure 5 shows the geometry generated by Mechanical Desktop ® v.6 for structural and CFD conceptions. Figure 6 presents details about this geometry for CFX ® v 5.7.

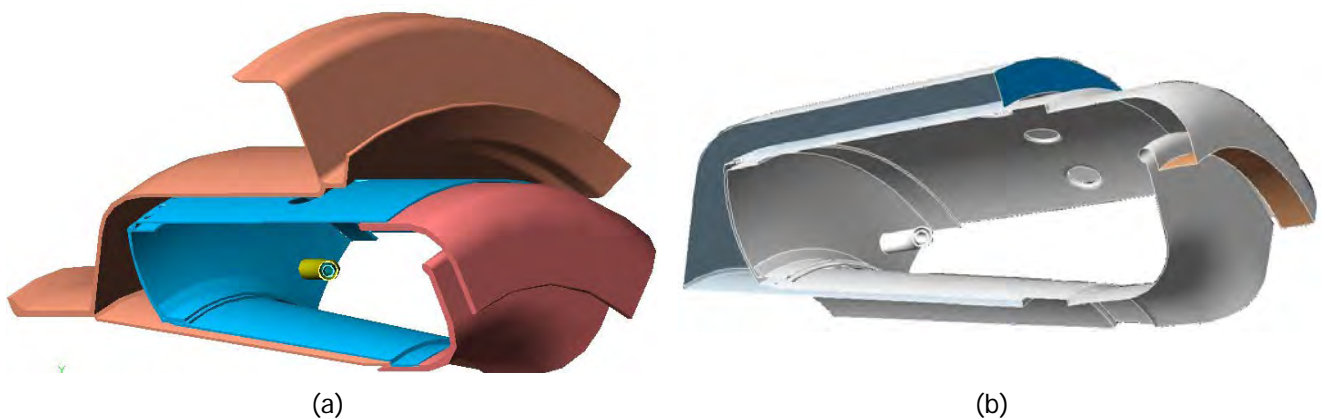


Figure 5 – Geometric model tested: (a) Structural Conception; (b) CFD conception



Figure 6 – Geometry Details

The mesh generated is unstructured with 1915508 tetrahedral and prismatic elements and 4666141 nodes. Figure 7 shows a part of this mesh in a longitudinal surface that is paralleled to the axial axle of combustion chamber in a region near to the outlet of nozzle. Figure 8 presents the details of this mesh.

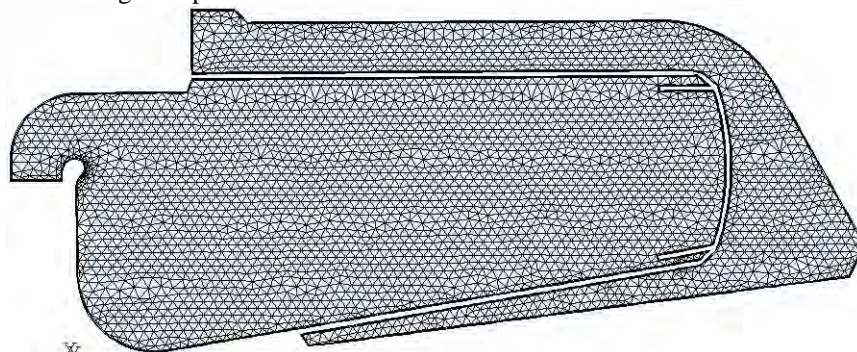


Figure 7 – Lateral view of mesh

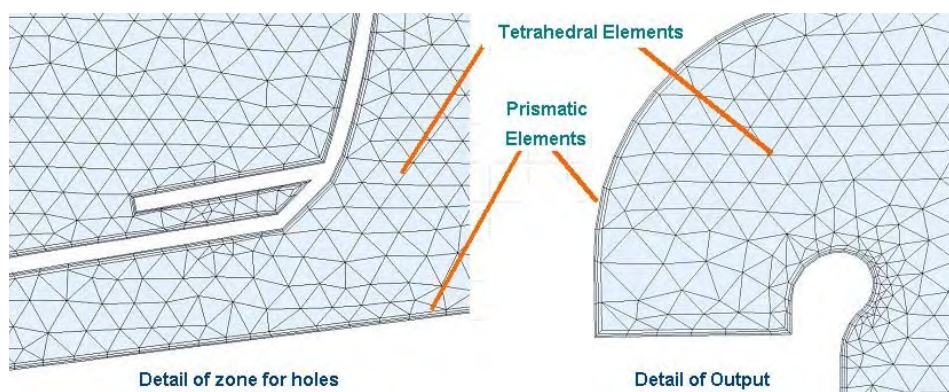


Figure 8 – Details of mesh: (a) in region near to the nozzle; and (b) in outlet of hot gas

The setting of this work fluid in CFD needs the knowledge about the thermal chemical balance, which is simulated by software GASEQ® v 7.2, which applies the Minimization Principle of Free Gibbs Energy or Lewis – NASA Method, Svehla and McBride (1973).

This method is an interactive calculation that permits to simulate the thermal chemical process for some chemical reactions and by mass balance among reactants and products. It has a library has more than 120 chemical species. In general, this software permits to esteem the concentrations for products and, mainly, the emissions.

The adopted work fluid is the methane gas, because in chemical composition of natural gas, this specie corresponds to 88 (%) in volume and it is a reference fuel for every works in combustion. Table 3 shows the chemical composition for natural gas, according to Abreu and Martinez (1999). Table 4 presents the chemical composition for the combustion of methane gas by GASEQ® v 7.2.

Table 3 – Chemical composition of alternative fuels for the model of small gas turbine

Species	Composition in volume (%)
	Natural Gas
N <sub>2</sub>	1.2
CO <sub>2</sub>	0.65
CO	0
CH <sub>4</sub>	88.56
C <sub>2</sub> H <sub>6</sub>	9.17
C <sub>3</sub> H <sub>8</sub>	0.42
H <sub>2</sub> S	0
H <sub>2</sub>	0
H <sub>2</sub> O	0
<b>Low Caloric Value [kcal / kg]</b>	<b>11430</b>

Table 4 – Composition of reactants and products in chemical reaction from GASEQ® v 7.2 after 5 iterations.

Local	Species	(Mol)	Mass (kg)
Reactants	CH <sub>4</sub>	0.105	1.68449
	O <sub>2</sub>	0,21	6.71975
	N <sub>2</sub>	0.76	22.1326
Products	CO <sub>2</sub>	0.095	4.19314
	H <sub>2</sub> O	0.204	3.6801
	N <sub>2</sub>	0.788	22.1026
	O <sub>2</sub>	0.005	0.15865
	CO	0.009	0.27232
	OH <sup>-</sup>	0.003	0.05375
	H <sup>+</sup>	0.0004	0.000418
	O <sup>-2</sup>	0.0002	0.00361
	H <sub>2</sub>	0.004	0.00794
	NO	0.002	0.06436

Some proprieties from natural gas are preserved, according to Abreu and Martinez (1999) and Lefebvre (1995):

- a) it generates low emissions about CO<sub>2</sub> and NO<sub>x</sub>;
- b) It has high combustion efficiency;
- c) its combustion can be considered clean, because it does not have sulfur oxide emission, that combined with water molecules can form acid;
- e) it has quick dispersion in air, due to its low molecular mass in relation to air; and
- f) It presents agility in the transport and good level of safety.



Figure 9 shows the reference surfaces defined by boundary conditions for CFX ® v 5.7.

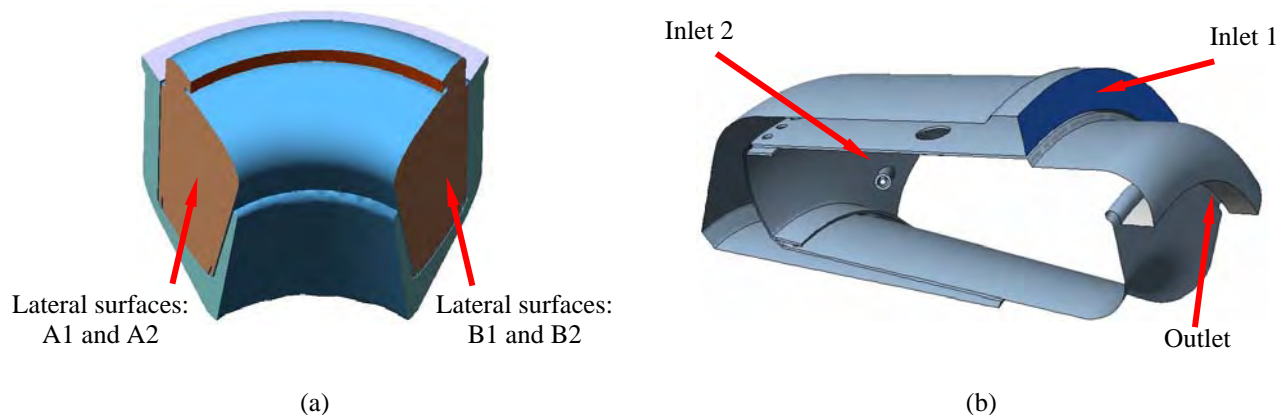


Figure 9 – Surfaces defined by boundary conditions: (a) Frontal view; and (b) Lateral view.

Table 5 presents the values adopted for the boundary conditions for Fig. 9.

Table 5 – Boundary conditions used in CFD

Greatness	Local	Unit	Values
Mass flow of air	Inlet 1	(kg/s)	0.1593
Temperature of air		(K)	500
Mass flow of fuel	Inlet 2	(kg/s)	0.0341
Temperature of fuel		(K)	295
Relative pressure of hot gas	Outlet	(Pa)	0.0
Temperature of hot gas		(K)	918
Rotational symmetry in axle	Lateral surfaces A1 – pre-chamber	---	Periodic Interface
Rotational symmetry in axle	Lateral surfaces A2 - chamber	---	Periodic Interface
Rotational symmetry in axle	Lateral surfaces B1 – pre-chamber	---	Periodic Interface
Rotational symmetry in axle	Lateral surfacesB2 - chamber	---	Periodic Interface

In Processing, the domain for mixtures is controlled by the Conservative Equations of Continuous, Momentum (Navier Stokes) and Energy in differential way in relation to the time and the space, respectively:

$$\frac{\partial(\rho \cdot \tilde{Y}_I)}{\partial t} + \frac{\partial(\rho \cdot u_j \cdot \tilde{Y}_I)}{\partial x_j} = \frac{\partial}{\partial x_j} \left( \Gamma_{ref} \cdot \frac{\partial \tilde{Y}_I}{\partial x_j} \right) + S_I \quad (1)$$

$$\frac{\partial(\rho \cdot U)}{\partial t} + \nabla \cdot \left[ \rho \cdot U \otimes U - \mu \cdot (\nabla U + \nabla U^T) \right] = -\frac{\partial p}{\partial x_j} + S_M \quad (2)$$

$$\frac{\partial(\rho \cdot H)}{\partial t} - \frac{\partial P}{\partial t} + \frac{\partial(\rho \cdot U_j \cdot H)}{\partial x_j} = \frac{\partial}{\partial x_j} \left( \lambda \cdot \frac{\partial T}{\partial x_j} + \sum_{N_i}^{N_j} \Gamma_i \cdot h_i \cdot \frac{\partial Y_i}{\partial x_j} + \frac{\mu_t}{Pr_t} \cdot \frac{\partial h}{\partial x_j} \right) + S_E \quad (3)$$

Where:  $\rho$  is the relative density of mixture, in (kg/m<sup>3</sup>);  $U$  is the velocity, in (m/s);  $T$  is the temperature, in (K);  $t$  is the time, in (s);  $p$  is the relative pressure, in (Pa);  $\Gamma_i$  is the molecular diffusion coefficient of specie  $i$ ;  $S_{ct}$  is the Schmidt's Number for high turbulence flows;  $\mu$  is the average dynamic viscosity for fluid, in (m<sup>2</sup>/s);  $\mu_t$  is the dynamic viscosity for turbulence in mixture, in (m<sup>2</sup>/s);  $S_I$  is the source term due to the chemical reaction rate for specie  $i$ , which is given by  $S_I = \omega_i / \rho$ , function to reaction rate  $\omega_i$  (**Kuramoto-Sivashinski' Equation**), Ariolli e Koch (2003), which is the key term to **Combustion Models**;  $S_M$  is the body force due to the gravity for each specie  $i$  in mixture,  $i$  and  $j$  is the Indicinal Notation of Einstein, in relation to direction  $x$  and  $y$ , respectively, to discretization of space;  $H$  is the total enthalpy for mixture;  $\lambda$  is the Thermal Conductivity Coefficient of domain;  $\Gamma_i$  is the molecular diffusion coefficient;  $S_E$  is the molecular energy for species  $i$  in mixture;  $Pr_t$  is the Prandtl's Number for turbulent flows; and  $\rho \cdot U \otimes U$  is the force due to Reynolds's Stress, in (N), which is the key term to **Turbulence Models**.

The equations (1), (2) and (3) are obtained considering the following hypotheses: transient, compressible and turbulent flow; specific heat for species in mixture is constant; small convection effect; and uniform combustion.

Besides, the heat transfer for radiation can be analyzed by the Transportation Equation for Spectral Radiation (RTE), according to Beer *et al.* (1971) e Chung (2002):

$$\frac{d I_v (r, s)}{d s} = -(K_{av} + K_{sv}) \cdot I_v (r, s) + K_a \cdot I_b (v, T) + \frac{k_{sv}}{4 \cdot \pi} \cdot \int_{4 \cdot \pi} d I_v (r, s') \cdot \Phi(s \cdot s') d \Omega + S \quad (4)$$

Where:  $\nu$  is the emission frequency;  $\mathbf{r}$  is the position vector;  $\mathbf{s}$  is the direction vector;  $s'$  is the distance traveled by radiation;  $\mathbf{K}_{av}$  is the absorption coefficient;  $\mathbf{K}_{sv}$  is the reflexion coefficient;  $\mathbf{I}_b$  is the intensity of emission in black body;  $\mathbf{I}_n$  is the intensity of spectral radiation that depend on the position  $\mathbf{r}$  and the direction  $\mathbf{s}$ ;  $\Omega$  is the solid angle;  $\Phi$  is function to phase of reflexion due to a immersed solid in domain; and  $\mathbf{S}$  is the source term for the radiation (combustion).

From this way, the modeling of flow is done by application of different turbulence models to solve the term represented by Reynolds Stress Tensor,  $\mathbf{U}_i \cdot \mathbf{U}_j$ , in Eq. (2).

Among the turbulence models applied by others authors, such as K- $\epsilon$ , RNG K- $\epsilon$  and Reynolds Stress Model (RSM), the RNG K- $\epsilon$  Model can be adopted because permits to describe flows in curved surfaces, where there is rotary flows, and have the capacity to capture the smallest vorticity generated by small holes in combustion chamber model. For this model, the Reynolds Stress Tensor  $\mathbf{U}_i \cdot \mathbf{U}_j$  can be calculated by expression, Yadigaroglu (1998):

$$-\rho \cdot (U_i \cdot U_j) = 2 \cdot \mu_t \cdot S_{ij} - \frac{2}{3} \cdot \rho \cdot k \cdot \delta_{ij} \quad (5)$$

Where  $\delta_{ij}$  is the Dirac Function; and  $S_{ij}$  is the Average Shear Tensor calculated by expression:

$$S_{ij} = \frac{1}{2} \cdot \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \quad (6)$$

Where the turbulent viscosity  $\mu_t$  is given by:

$$\mu_t = C_\mu \cdot \rho \cdot \frac{k^2}{\epsilon} \quad (7)$$

Where:

$$k = \frac{1}{2} \cdot (U_i \cdot U_j) \quad \epsilon = \nu \cdot \left( \frac{\partial U_i}{\partial x_j} \cdot \frac{\partial U_j}{\partial x_i} \right) \quad (8)$$

Thus, the main equations used to simulate the turbulence by RNG K- $\epsilon$  Model are:

$$\frac{\partial (\rho \cdot U_j \cdot k)}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \cdot \frac{\partial k}{\partial x_j} \right] + 2 \cdot \mu_t \cdot S_{ij} \cdot S_{ij} - \rho \cdot \epsilon \quad (9)$$

$$\frac{\partial (\rho \cdot U_j \cdot \epsilon)}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_\epsilon} \right) \cdot \frac{\partial \epsilon}{\partial x_j} \right] + 2 \cdot C_{\epsilon 1} \cdot \frac{\epsilon}{k} \cdot \mu_t \cdot S_{ij} \cdot S_{ij} - C_{\epsilon 2} \cdot \rho \cdot \frac{\epsilon^2}{k} \quad (10)$$

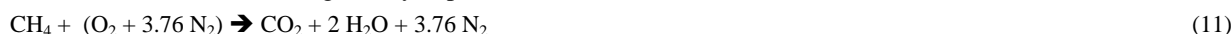
Where there are five important constants,  $C_\mu$ ,  $C_{\epsilon 1}$ ,  $C_{\epsilon 2}$ ,  $\sigma_k$  e  $\sigma_\epsilon$ , whose typical values based on experimental tests for isotropic flows are presented by Tab. 6, where are shown others three auxiliary constants  $\eta$ ,  $\eta_0$ , and  $\beta$ :

Table 6 – Typical values for constants in turbulence model RNG K- $\epsilon$

$C_\mu$	$C_{\epsilon 1}$	$C_{\epsilon 2}$	$\sigma_k$	$\sigma_\epsilon$	$\eta$	$\eta_0$	$\beta$
0.076 to 0.094	$\eta \cdot \left( \frac{1 - \eta}{\eta_0} \right)$ 1.42 - $\frac{1}{1 + \beta \cdot \eta^3}$	1.51 to 1.85	0.6103 to 0.8256	0.6461 to 0.7897	$\left( \frac{k}{\epsilon} \right) \cdot (2 \cdot S_{ij} \cdot S_{ij})^{1/2}$	3.94 to 4.82	0.012 to 0.017

Besides, among the combustion models such as, Eddy Dissipation Model (EDM), Finite Rate Chemistry (FRM), and Laminar Flamelet Model (LFM), the EDM Model or Eddy Break Up (EBU) can be adopted because it can describe combustion reactions with pre-mixture, where the effects of viscosity can be considered, between methane and air in nozzles, where the mechanism of reaction can be described by the following equations of combustion, Gurgel et al (2002):

Phase 1: the main reaction is given by expression:



Phase 2: the reaction is described using stoichiometric air in two steps:



According to Poinso e Veynante (2005), the reaction rate can be affected by turbulence. From this way, the reaction rates for reactants and products are, respectively (kinetic chemical):

$$R_K = A \cdot \frac{\varepsilon}{k} \cdot \min \left( \frac{[I]}{v_{KI}^*} \right) \quad (14)$$

$$R_K = A \cdot B \cdot \frac{\varepsilon}{k} \cdot \left( \frac{\sum_P [I] \cdot W_I}{\sum_P v_{KI}^{**} \cdot W_I} \right) \quad (15)$$

Where  $[I]$  is the molar concentration of each species between the reagents;  $A$  is a proportionality constant, that depends on the physical properties of the reactants and if there is or not pre mixture and the conditions of reactions with or not free radicals;  $P$  tracks all the products generated in reaction  $K$ ;  $W_I$  is the molecular weight of each component of the product;  $B$  is a numerical parameter that indicates if the simple reaction of or multiple stage. If  $B$  is negative, the formation of the products for determined reaction  $K$  is not performed;  $\varepsilon$  is the dissipation energy due the turbulence  $k$  is the turbulence in flow; and  $v$  is the stoichiometric coefficient for reactants and products.

On the other hand, among the models for Heat Transfer by Radiation, such as, Rosseland (to opaque domains), P1, Monte Carlo, Discrete Transfer (the three to semi-transparency domains) and Spectral (to full transparency domains), Hottel and Sarofim (1967), the P1 Model can be adopted because permits to describe the radiation in domains with unitary emissive and unitary absorption in walls, which can be considered not catalytic (not affect the reactions).

The P1 Model or Radiation Differential Approximation, Raithby (1991), is a simplification of equation (4), considering that the radiation intensity is independent of direction and it is in an isotropic domain. In this model, the radiation heat flux in diffusion limit for domain is given by equation:

$$q_{rv} = - \frac{1}{3 \cdot (K_{av} - K_{sv}) - A \cdot K_{sv}} \cdot \nabla G_v \quad (16)$$

In equation (4), this term permits to get the following identity:

$$-\nabla \cdot \left[ \frac{1}{3 \cdot (K_{av} - K_{sv}) - A \cdot K_{sv}} \cdot \nabla G_v \right] = K_{av} \cdot (E_{bv} - G_v) \quad (17)$$

Where  $A$  is the Linear Coefficient of Anisotropy.

From this, considering that the radiation emitted by walls in domain is independent of direction, the boundary condition can be calculated by equation:

$$n \cdot q_{rv} = - \frac{1}{3 \cdot (K_{av} - K_{sv}) - A \cdot K_{sv}} \cdot \frac{\partial G_v}{\partial n^+} = \frac{\varepsilon_v}{2 \cdot (2 - \varepsilon_v)} \cdot [E_{bv} - G_v]_w \quad (18)$$

Where  $n$  is the normal vector in relation to the wall surface;  $n^+$  is the distance in the same direction of  $n$ ; and  $w$  is the magnitude of  $n^+$  in the walls.

More details about the radiation model P1 are given by Raithby (1991).

On the other hand, the heat transfer mechanism in tested combustion chamber can be featured by the heating of domain and walls using radiation, while the cooling of walls and the dilution of heat gas is done by convection.

Thus, among the energy variation models used by CFD, such as, Isothermal Model; Thermal Energy Model; and Total Energy Model, it is adopted the Total Energy Model because the effects of the thermal energy and the kinetic energy are super posted in calculation of thermal balance by the first law of thermodynamic.

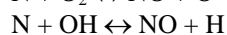
Besides, in general way, there are three different mechanisms for NOx emission. Table 7 shows these three mechanisms and its description, Lefebvre (1983).

Table 7 – Mechanisms for NOx generation, Lefebvre (1983)

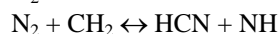
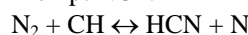
Mechanism	Description
Thermal NOx	Zeldovich's Theory – It is produced NOx by the oxidation of atmospheric nitrogen in the gases after the flame.
Prompt NOx	Fenimore's Theory – it is generated NOx by the reactions of high speed in the front of the flame.
Fuel NOx	It is formed NOx by the oxidation of existing nitrogen in the fuel and can generates free radicals with low molecular weight, such as, NH <sub>3</sub> , NH <sub>2</sub> , NH, CN and others.

Among these models to simulate the NOx emission, the adopted models are Thermal Model of Zeldovich and Prompt Model of Fenimore, because of emission from methane combustion depends on oxidation of atmosphere nitrogen in front of flame and the high velocity reactions, respectively, given by the following basic reactions, Vandebroek *et al* (2003):

Thermal NOx:



Prompt NOx:



### 5. Validation of CFD Calculation

In general, all CFD calculations shall be performed in comparison to the experimental results for simple geometries, in function to guarantee the right result with quality to represent the behavior of physical phenomena studied for more complex geometries.

A good example for it, which can be applied for combustion chambers, is the CFD validation for radiation calculation in a geometry proposed by Stuttaford (1997), which is used to validate the numerical method developed by Shah (1979).

This geometry domain consists of a little cylinder with length 5 (m) and diameter 2 (m). The walls have temperature of 227 (°C) and an emissivity of 0.8. The gas contains a hot region at a temperature of 1427 (°C) and has an absorption coefficient of 0.6 (m<sup>-1</sup>). The external surface in cylinder is remain with temperature equal to 827 (°C) and has an absorption coefficient of 0.05 (m<sup>-1</sup>), as it is shown by Fig. 10.

The calculation uses mesh with 89782 tetrahedral elements. The convergence condition has 50 iterations with goal error equal to 10<sup>-4</sup> for heat transfer. Besides, the radiation model is Heat Transfer Discrete Model, where the radiation source can be discretized by 8, 64 and 128 rays with enough dimensions for that the average refraction is smaller than emission, because of work fluid is a hot gas without particles (photons). The adopted model to discrete the spectrum is Gray Model, because the spectrum is uniform. The Scattering Model is considered null, because that the term in transport equation for energy due the particles (photons) is not necessary and the domain is isotropic.

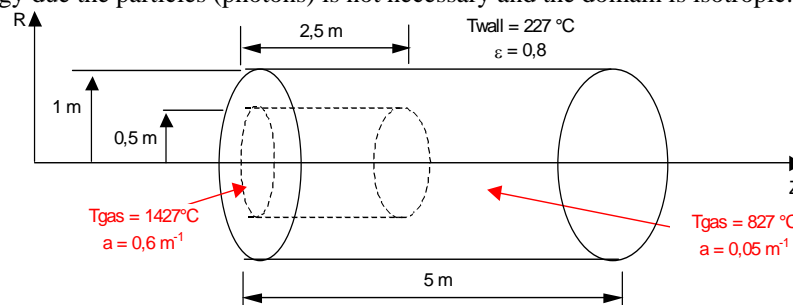


Figure 10 – Geometry Domain and boundary conditions for radiation calculation, Stuttaford (1997)

Table 8 presents the intensity of radiation in non-dimensional form obtained by experimental test and by CFD calculation, where **I<sub>o</sub>** is the maximum radiation intensity and equal to 80000 (W / m<sup>2</sup>) and **L** is the length of cylinder and equal to 5 (m).

Table 8 – Results from Experimental Tests and CFD calculation

Distance	Z / L	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Experimental Radiation Intensity <b>I / I<sub>o</sub></b>		0.78	0.93	1.00	0.99	0.90	0.60	0.34	0.24	0.18	0.14	0.13
Numerical Radiation Intensity <b>I / I<sub>o</sub></b>		0.74	0.90	0.98	0.97	0.86	0.56	0.31	0.21	0.15	0.11	0.09

From Tab. 8, it is possible to done a graphical comparison in relation to Non-dimensional Net Radiation Heat Flux for Axial Distance. In figure 8, this comparison is done, considering the maximum radiation intensity **I<sub>o</sub>** and the length of cylinder **L**.

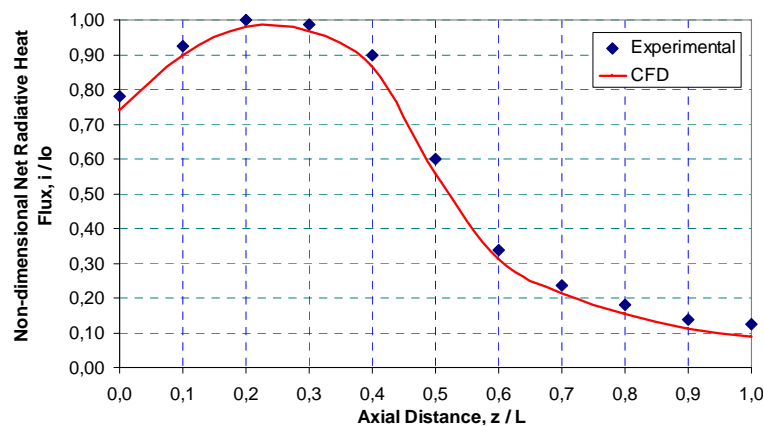


Figure 11 – Non-dimensional Radiation Heat Flux for Axial Distance

It is observed from Fig. 11 that the behavior of radiation intensity is due the characteristics of hot gas zone with temperature 1427 (°C) and the other hot gas zone with temperature 827 (°C).

The values of experimental and numerical tests are near and the CFD calculation permits to get satisfactory results in a simple geometry for radiation calculation. The maximum difference between the results is 4 (%) in non-dimensional axial distance equal to 0.5. From this, CFD can be used to study more complex geometries, where the heat transfer is done by radiation too, like as, in the combustion cases.

## 6. Analysis of Results

In general, the duration for the calculations is approximately equal to 14 hours using a computer DELL with double processor of 3.4 GHz and 1GByte of RAM, considering a goal error equal to  $10^{-5}$  for mass flow, velocity, turbulence and turbulent dissipation energy in CFX ® v 5.7. Figure 12 presents the distributions of velocity and pressure.

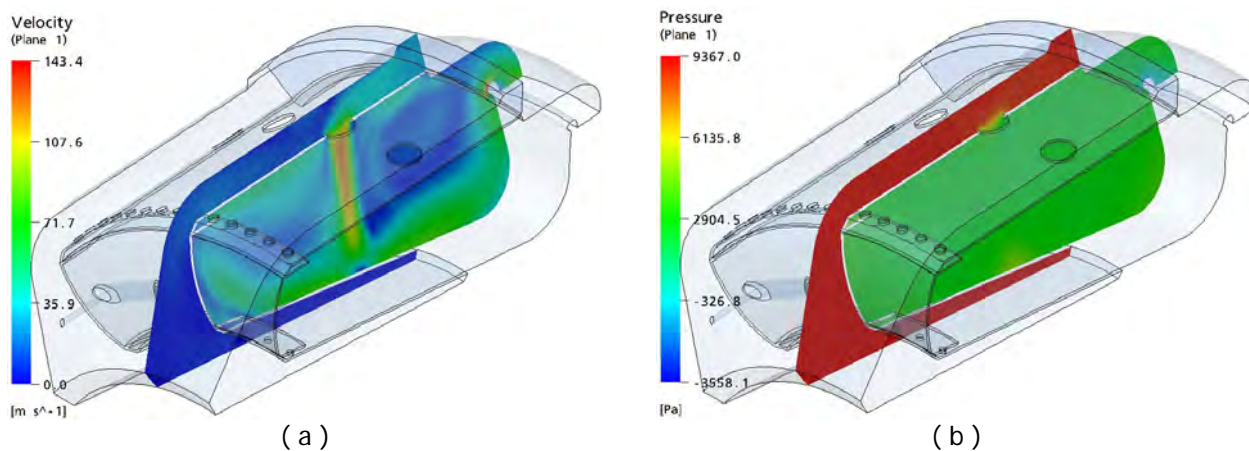


Figure 12 – (a) Velocity contour; (b) Pressure contour

Figure 13 shows the distribution of temperature and the format of flame from iso-surfaces of temperature, with gradients equal to 200 (°C), approximately, where is shown that the flame is formed from a specific distance in relation to the nozzle with low dispersion. It is a characteristic for this kind of annular combustion chamber.

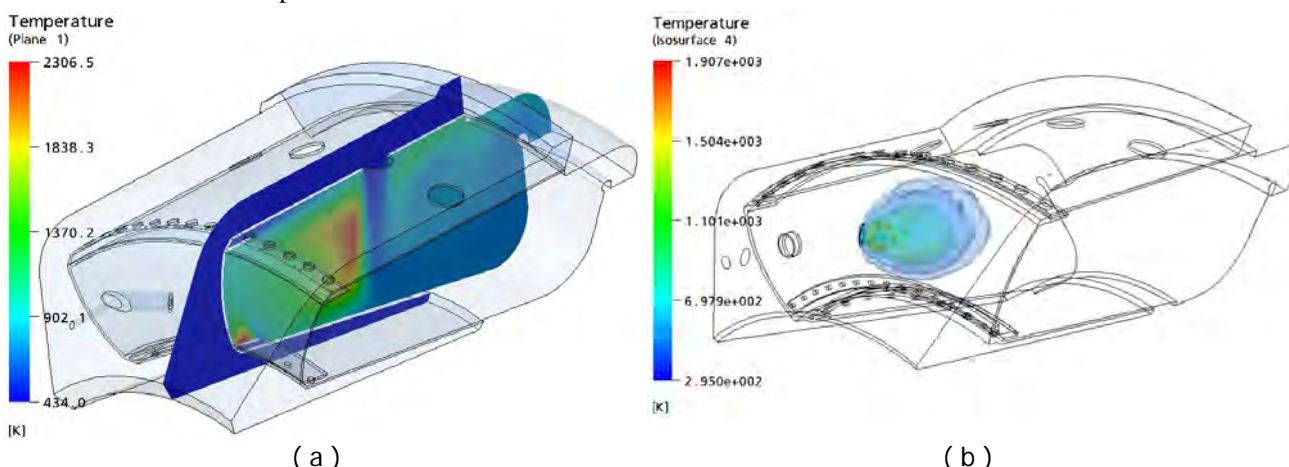


Figure 13 – (a) Temperature contour; (b) Iso-surfaces of temperature

The distribution of iso-surfaces of temperature shows that the flame has dispersion in annular combustion chamber, as well as, the temperature is affected by rotary flow. This flow changes the position where the flame has its maximum temperature after the ignition. Maybe, it is necessary to change the dimensions of lateral holes of dilution air in Liner.

Besides, in longitudinal surface that is shown in Fig. 12 and 13, it is possible to determine the variation for velocity, pressure and temperature along the local axle from nozzles, which is parallel to main axle of combustion chamber, according to Fig.14. Table 9 presents the values for these greatnesses for this local axle obtained after the convergence.

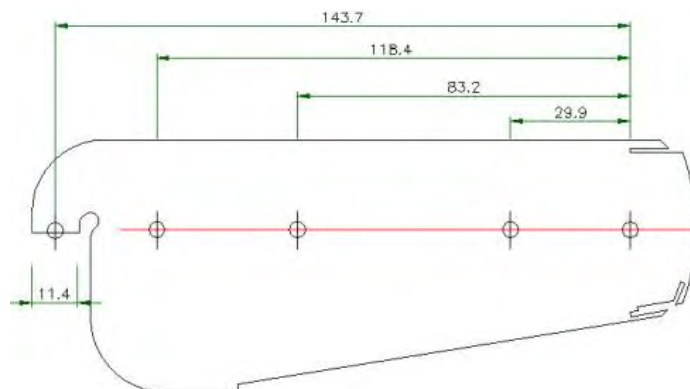


Figure 14 – Position of Z axle from nozzle

Table 9 – Values obtained after the convergence

Greatness	Symbol	Unit	Values				
Position in relation to nozzle	Z	(mm)	0	30	83	118	144
Average velocity of flow	$V_{GAS}$	(m/s)	52	45	32	27	62
Average pressure	$P_{GAS}$	(Pa)	2350	3200	3800	4100	0
Gas temperature	$T_{GAS}$	(°C)	907	2033	1012	810	630
Gas density	$\rho_{GAS}$	(kg/m <sup>3</sup> )	0.35	0.18	0.36	0.42	0.52
Reynolds' number	$RE_{GAS}$	(--)	4.71E+07	4.07E+07	2.90E+07	2.44E+07	5.61E+07

Besides, others important parameters that can be determined are the maximum temperature of flame and the temperature of gas in exhaustion, whose values are 2033 (°C) and 630 (°C), respectively, which are coherent with data from manufacturer Solar Turbine, whose experimental values for temperature of gas in exhaustion is 637.9 (°C). Besides, this maximum temperature computed by Combustion Model EDM is near to the maximum temperature of flame esteemed by GASEQ @ V 7.2, approximately 2220 (°C) for methane gas.

From Tab. 9, it is possible to determine the characteristic curves of operation for the model, as it is shown from Fig. 15 to Fig. 17, whose reference is the Z-axis in longitudinal surface (see Fig. 14).

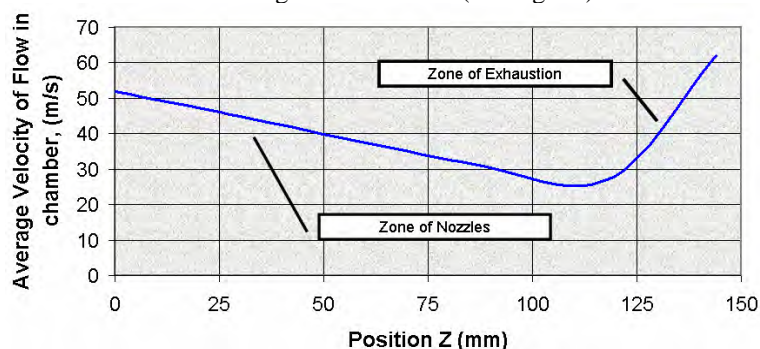


Figure 15 – Characteristic velocity curve

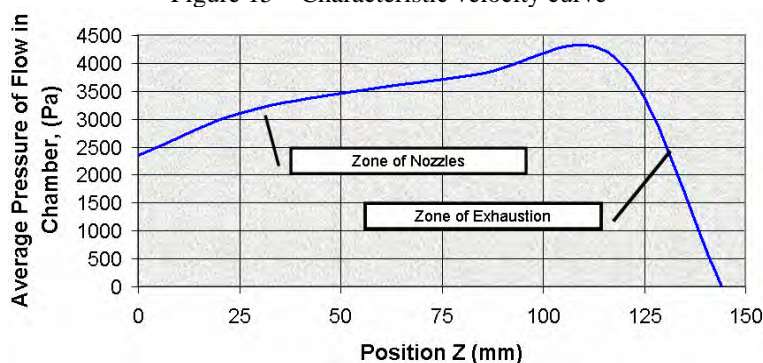


Figure 16 – Characteristic pressure curve

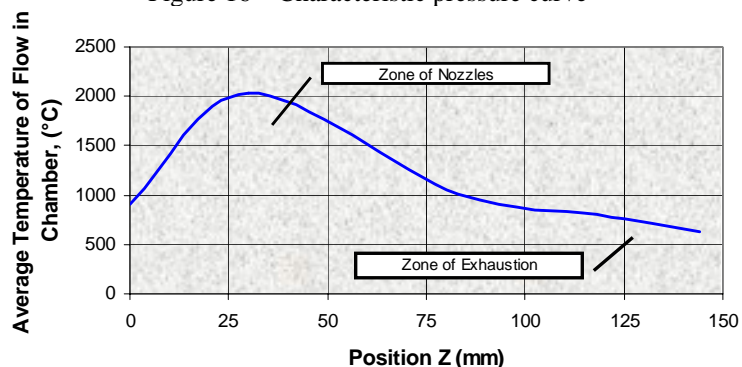


Figure 17 – Characteristic temperature curve

From Fig. 17, it is possible to identify the maximum temperature of flame is near to the nozzles, but its position is affected by the secondary flow, which can be represented by high flow of air from dilution holes in lateral surface (for distances between 60 and 90 (mm) from nozzles) and by the rotary flow due to the inclination of nozzles in relation to the main axle on combustion chamber, respectively, Figs 12(a) and 13(a). Because of it, it is possible to avoid the approximation of hot gases in relation to the metallic walls, too.

The velocity and pressure values obtained in Figs. 15 and 16, respectively, are affected by the turbulence zone near to the nozzles and exhaustion zones; see Fig. 13(a).

Other characteristic that can be identified is the rotary flow in zone of nozzles, which can benefit the combustion, because it contributes to increase the residence time.

Other important information is the efficiency of combustion chamber that can be estimated using the following equation, Odgers and Carrie (1973):

$$\eta_{CC} = 100 - \left( 0.95 + \frac{37.5}{\% \text{ CO}_2} \right) \cdot \frac{(T_{GAS} - T_o)}{100} \quad (19)$$

Where  $T_o$  is the reference temperature in environment, in [°C].

In tested model, the reference temperature is 22 (°C). The percentage of CO<sub>2</sub> in outlet surface is, approximately, 17.5 (%) and the Temperature of gas or flame is 2033 (°C). So, the combustion efficiency is, approximately, 37.8 (%) for this combustion chamber in these operation condition.

According to manufacturer Solar Turbines, the combustion efficiency for this equipment is between 35 (%) and 40 (%) for the efficiency of cycle between 25 (%) and 30 (%) in relation to the power 50 (kW).

## 7. Conclusions

In this work, the important contribution is the development about a practical scheme using commercial softwares, like as Gatecycle®, GASEQ® and CFX ® to study the behavior of combustion and the flow in annular combustion chambers for small gas turbines to attend its adaptability and economic feasible to use different fuels.

Considering the methane gas to represent the natural gas, whose chemical composition is done mainly by 88(%) in volume of methane, as well as different models for flow, combustion, heat transfer and emission in CFD, whose boundary condition are set by parameters from thermal cycle of small gas turbine and chemical concentrations of reactants and products for combustion, the combustion chamber can be features by the following notes:

(a) From pressure, velocity and temperature distributions, the high flow of air from dilution holes in lateral surface in Liner affects the position where it is the maximum temperature of flame in relation to nozzles, whose value is 2033 (°C) for distances between 60 and 90 (mm) from nozzles;

(b) The inclination of nozzles can induce a rotary flow, which affects the position and stability of flame and the residence time for combustion;

(c) The methane gas signs that the flame can be dispersed, which obligates in new design of combustion chamber. This new design is to avoid the approximation of hot gases in relation to the metallic walls;

(d) The temperature of gas in exhaustion obtained by CFD is equal to 630 (°C), whose value is near to the temperature from manufacturer Solar Turbine, which is equal to 637.9 (°C); and

(e) The CFD simulation permits to esteem the combustion efficiency around 38 (%) for this tested model. This is a satisfactory precision, once that the manufacturer esteems this efficiency between 35 (%) and 40 (%) for power 50 (kW).

## 8. Acknowledgments

For all team from NEST at UNIFEI by concession of equipment about the small gas turbine used in simulations.

## 9. References

- Abreu, P. L. and Martinez J. A., 1999, "Natural Gas: Fuel for the New Millennium", Plural Communication, Porto Alegre, RS, Brazil.
- Alencar H. S., Villanova H. F., and Antonio M. R. N., 2005, "Analysis of Flame Behavior in Small Combustion Chambers Using CFD", Proceedings of COBEM 2005, 18th International Congress of Mechanical Engineering by ABCM, November 6-11, Ouro Preto, MG, Brazil.
- Allen J. W., 1998, "Low NOx Burner Designs", Proceedings of the American Power Conference, Vol. 60 – II, pp 869 – 874.
- Arioli G. e Koch H., 2003, "The Bifurcation Graph of Kuramoto-Sivashinski Equation", Third International Workshop on Taylor Methods, Proceedings, Miami, USA. P. 15.
- Beer J. M., Foster P. J. and Siddall R. G., 1971, "Calculation Methods of Radiative Heat Transfer", HTFS Design Report No. 22 AEA Technology (Commercial).
- Borbely, Ann-Marie, and Kreider, Jan F., 2001, "Distributed Generation", Washington D.C.: CRC Press.
- Chigier N. A., and Berr J. M., 1983, "Combustion Aerodynamics", Robert E. Krieger Publishing Company, Malabar, Florida, USA.
- Chung T. J., 2002, "Computational Fluid Dynamics", Cambridge University Press, UK, ISBN 0-521-59416-2, p.1012.
- Fuller E. J., and Smith C. E., 1994, "CFD Analysis of a Research Gas Turbine Combustor Primary Zone", 30th AIAA / ASME / SAE / ASEE Joint Propulsion Conference, June 27 – 29, Indianapolis, IN, USA.
- Gosselin P., DeChamplain S. K., and Kretschmer D., 2000, "Three Dimensional CFD Analysis of a Gas Turbine Combustor", 36th AIAA / ASME / SAE / ASEE Joint Propulsion Conference and Exhibit, pp 11, Huntsville, Alabama, USA.

- Gurgel C. A., Alves F. S., Olavo M. M. C., 2002, “Projeto e Estudo de Desempenho da Câmara de Combustão de uma Microturbina”, IX Congresso Brasileiro de Engenharia e Ciências Térmicas -- 9th Brazilian Congress of Thermal Engineering and Sciences, Paper CIT02-0391, Itajubá, MG.
- Hamer A. J., and Roby R. J., 1997, “CFD Modeling of a Gas Turbine Combustor Using Reduced Chemical Kinetic Mechanisms”, AIAA 1997 – 3242, 33rd AIAA ASME / SAE / ASEE Joint Propulsion Conference and Exhibit, July 6 – 9, Seattle, WA, USA.
- Hottel, H.C. and Sarofim, A.F., 1967, “Radiative Transfer”, McGraw-Hill, New York, USA.
- Keating E. L., 1993, “Applied Combustion”, Marcel Dekker Inc., New York, USA.
- Kuo K.K., 1986, “Principles of Combustion”, John Wiley & Sons Inc., 1<sup>st</sup> Edition, New York, EUA.
- Lee D., Yeh C., Tsuei Y., Jiag W., and Chung Y., 1990, “Numerical Simulation of Gas Turbine Combustor Flows”, 26th AIAA / ASME / SAE / ASEE Joint Propulsion Conference, July 16 – 18, Orlando, FL, USA.
- Lefebvre A. H., 1995, “The Role of Fuel Preparation in Low Emission Combustion”, ASME Journal of Engineering for Gas Turbines and Power, Vol. 117, pp. 617-654.
- \_\_\_\_\_, 1983, “Gas Turbine Combustion”, Taylor and Francis Co., 1<sup>st</sup> Edition, ISBN 0-89116-896-6, New York, USA.
- Louis J. J., Kok J. B. W. E, Klein S. A., 2001, “Modeling and Measurements of a 16-Kw Turbulent No-adiabatic Syngas Diffusion Flame in a Cooled Cylindrical Combustion Chamber”, Combustion and Flame, Volume 125, Issues 1-2, pp. 1012-1031.
- Melick T. A. et al., 1999, “Burner Modifications for Cost Effective NOx Control-Part 1”, Proceedings of the American Power Conference, Vol. 61 – I, pp 478 – 482.
- \_\_\_\_\_, 1998, “Burner Modifications for Cost Effective NOx Control-Part 2”, Proceedings of the American Power Conference, Vol. 60 – II, pp 855 – 860.
- Nickolaus D. A., Croker D. S., and Smith C. E., 2002, “Development of a Lean Direct Fuel Injector for Low Emission Aero Gas Turbine”, ASME.
- Odgers J., Carrier C., 1973, “Modeling Gas Turbine Combustors; Considerations of Combustion Efficiency and Stability”, Journal of Engineering for Power, pp. 105-113.
- Poinsot T. and Veynante D., 2005, “Theoretical and Numerical Combustion”, 2<sup>a</sup> Edition, Edwards Inc, ISBN 1-930217-10-2, Philadelphia, USA, p. 522.
- Raithby, G.D., 1991, “Equations of Motion For Reacting, Particle-Laden Flows”, Progress Report, Thermal Science Ltda, EMR.
- Shah, N. G., 1979, “New Method of Computation of Radiant Heat Transfer in Combustion Chambers”, PhD Thesis, University of London, UK.
- Stuttaford P. J., 1997, “Preliminary Gas Turbine Combustion Design Using a Network Approach”, PhD Thesis, Cranfield University, USA.
- Svehla R. A., McBride B. J., 1973, “Fortran IV Computer Program for Calculation of Thermodynamic and Transport Properties of Complex Chemical Systems”, NASA, TN D-7056.
- Vandebroek L., Winter H., and Berghmaus J, 2003, “Numerical Study of the Auto Ignition Process in Gas Mixtures Using Chemical Kinetics”, Heat Mass Transfer Journal.
- Wakabayashi T. et al., 2002, “Performance of a Dry Low NOx Gas Turbine Combustor Designed with a New Fuel Supply Concept”, Engineering for Gas Turbines and Power Journal, ASME, Vol. 124, pp. 771-775.
- Williams F., 1985, “Combustion Theory”, Addison – Wesley Publishing Company Inc., 2<sup>nd</sup> Edition, London, UK.
- Yadigaroglu G. et al., 1998, “Numerical and Experimental Study of Swirling Flow in a Model Combustor”, Heat Mass Transfer Journal, Vol. 41, No. 11, pp. 1485-1497.
- U.S. Environment Protection Agency, 1993, “Alternative Control – Techniques Document – NOx Emissions from Stationary Gas Turbines”, North Carolina, USA.

## **10. Copyright Notice**

The author is the only responsible for the printed material included in his paper.



## IDENTIFICAÇÃO DE UM SISTEMA TÉRMICO USANDO REDE NEURAL DE B-SPLINES

### Leandro dos Santos Coelho

Laboratório de Automação e Sistemas, LAS  
Programa de Graduação em Engenharia de Produção e Sistemas, PPGEPS  
Pontifícia Universidade Católica do Paraná – PUCPR  
Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil  
leandro.coelho@pucpr.br

### Viviana Cocco Mariani

Programa de Pós-Graduação em Engenharia Mecânica  
Pontifícia Universidade Católica do Paraná – PUCPR  
Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil  
viviana.mariani@pucpr.br

*Resumo. As redes neurais de B-splines empregam funções de base para aproximar uma função contínua, possivelmente multidimensional. As redes de B-splines podem ser comparadas com as redes neurais de função de base radial, que utilizam funções Gaussianas como funções elementares de base. As redes B-splines são geralmente treinadas por algoritmos baseados em informação do gradiente, o que pode levar a um ponto de mínimo local durante o procedimento de treinamento. Para superar os problemas encontrados pelos métodos convencionais de treinamento, o método de otimização com evolução diferencial utiliza uma busca estocástica para otimização global das redes B-splines. Uma abordagem de evolução diferencial para ajustar os pontos de controle (knots) de uma rede neural de B-splines é proposto neste artigo. Os resultados de simulação para identificação um passo à frente de um sistema térmico SISO (Single-Input Single Output) demonstram a eficiência da abordagem de rede neural de B-splines proposta.*

*Palavras chave: sistema térmico, rede neural de B-splines, identificação não-linear, evolução diferencial*

### 1. Introdução

As redes neurais são metodologias da inteligência computacional inspiradas em estudos do cérebro e do sistema nervoso humano. As redes neurais podem ser consideradas como sistemas dinâmicos não lineares consistindo de elementos de processamento (neurônios), que são interconectados entre si, com ponderações ajustáveis (Haykin, 1996). Entre as características relevantes das redes neurais têm-se: processamento paralelo, aprendizado, memória associativa e distribuída. Estas características são inspiradas nas redes neurais biológicas. As redes neurais providenciam, usualmente, conhecimento quantitativo não-paramétrico sendo adequadas para identificação de sistemas, aprendizado e adaptação em sistemas com comportamento dinâmico complexo (Coelho e Villa, 2005).

Em particular, as redes neurais de B-spline (RN-BS) são uma alternativa de modelo não-linear para problemas de regressão e classificação de padrões. Parte do crescente interesse pelas RNs-BS é proveniente do fato delas possuírem fundamentação estatística e serem consideradas aproximadores universais de funções com dada precisão, da mesma forma que as redes neurais função de base radial (Guerra e Coelho, 2005), realizando um mapeamento das entradas de rede de dimensão vetorial  $m$ , em uma saída de dimensão vetorial  $n$  de qualquer função vetorial contínua. Neste contexto, a literatura sobre as RNs-BS tem apresentado vários estudos em identificação e controle de processos não-lineares (An *et al.*, 1994; Wang *et al.*, 1995; Lightbody *et al.*, 1997; Yiu *et al.*, 2001; Chen *et al.*, 2004).

As RNs-BS apresentam potencialidades de generalização local, onde a interferência no aprendizado é minimizada e o aprendizado é relativamente rápido, devido ao número mínimo de pesos ajustados a cada vetor de treinamento apresentado à rede neural. Na literatura, várias abordagens de treinamento de RN-BS têm sido apresentadas, entre as quais usando algoritmo ANOVA (An *et al.*, 1994; de Vries *et al.*, 1998), algoritmos genéticos (Leu, 2004), entre outros (Wang *et al.*, 1995).

A contribuição deste trabalho é avaliar as potencialidades de usar uma abordagem de treinamento da RN-BS baseada em otimização por evolução diferencial (ED). A ED é um paradigma da computação evolutiva (ou evolucionária) desenvolvido por Rainer Storn e Kenneth Price (Storn e Price, 1995; Storn, 1997) para problemas de otimização não-linear contínua. Basicamente, a ED realiza mutações nos vetores pela adição ponderada de diferenças aleatórias entre eles. A ED, neste contexto, determina os nós de cada uma das funções de base B-spline da RN-BS em uma aplicação de identificação não-linear de um sistema térmico.

O artigo é organizado da seguinte forma. A descrição da RN-BS e o procedimento de treinamento usando ED são detalhados na seção 2. Uma breve descrição do sistema térmico e a análise dos resultados obtidos na identificação são detalhadas na seção 3. Finalizando, a conclusão e comentários sobre futura pesquisa são apresentados na seção 4.

## 2. Rede neural de B-spline

Uma RN-BS consiste de uma rede neural da classe de redes *feedforward*, sendo que a RN-BS é composta de funções B-splines. Na RN-BS, cada função de base é composta de  $q$  segmentos polinomiais. Aqui existe uma relação simples e recursiva estável para avaliação da pertinência de uma função de base B-spline de ordem  $q$ , tal que

$$N_q^j(x) = \left( \frac{x - \lambda_{j-q}}{\lambda_{j-1} - \lambda_{j-q}} \right) N_{q-1}^{j-1}(x) + \left( \frac{\lambda_j - x}{\lambda_j - \lambda_{j-q+1}} \right) N_{q-1}^j(x) \quad (1)$$

$$N_1^j(x) = \begin{cases} 1 & \text{se } x \in I_j \\ 0 & \text{outros casos} \end{cases} \quad (2)$$

onde  $N_q^j(\cdot)$  é definido como a  $j$ -ésima função de base univariada de ordem  $q$ ;  $\lambda_j$  é o  $j$ -ésimo nó (*knot*) e  $I_j$  é o  $j$ -ésimo intervalo. A saída da RN-BS é dada por

$$\hat{y}_k = f(x_k) = \sum_{j=1}^p w_j N_q^j(x_k) \quad (3)$$

sendo  $p$  o número de *knots*,  $x_k$  e  $\hat{y}_k$  são as entradas e saída da rede, respectivamente;  $w_j$  é a ponderação atribuída para a  $j$ -ésima função de base e  $N_q^j(\cdot)$  é dada pela equação recursiva (2). O índice  $j$  é associado com a região de suporte local  $\lambda_{(j-q)} \leq x \leq \lambda_{(j)}$ , enquanto o índice  $q$  indica a ordem das funções de base (Harris *et al.*, 1993).

A qualidade da aproximação depende da alocação de nós das funções B-spline. O objetivo da otimização da RN-BS usando ED é a determinação dos nós de cada uma das funções de base B-spline. Entretanto, neste trabalho o número de funções de base é escolha do projetista.

### 2.1. Otimização da RN-BS usando evolução diferencial (ED)

Os algoritmos evolucionários (ou evolutivos) são ferramentas promissoras para busca, otimização, aprendizado de máquina e para resolução de problemas de projeto. Estes algoritmos utilizam a evolução simulada para busca de soluções em problemas complexos (Bäck *et al.*, 1997; Whitley, 2001). Os algoritmos evolucionários são baseados em uma população de indivíduos, onde cada um representa um ponto de busca no espaço de soluções potenciais de um dado problema. Os algoritmos evolucionários possuem alguns procedimentos de seleção baseados na aptidão (*fitness*) dos indivíduos mais aptos, operadores de cruzamento e mutação.

Os algoritmos evolucionários incluem algoritmos genéticos, programação evolucionária, estratégias evolutivas, programação genética, entre outras variantes. Segundo Bäck *et al.* (1997), existem diversas variantes de implementações de algoritmos evolucionários, no entanto, as diferenças principais repousam na representação dos indivíduos, no projeto dos operadores de cruzamento e/ou mutação e nos mecanismos de seleção.

O operador de seleção emprega o princípio de sobrevivência dos indivíduos mais aptos, através de uma metáfora aos procedimentos de reprodução assexuada e seleção natural, de acordo com o grau de adaptação do indivíduo ao ambiente. O objetivo básico do operador de seleção é enfatizar as melhores soluções que constituem uma população. O operador não cria nenhuma nova solução. Este operador seleciona as soluções relativamente aptas de uma população e remove as soluções remanescentes.

O operador de cruzamento é responsável pela troca de material genético entre os indivíduos, com maior probabilidade de reproduzirem os indivíduos mais aptos em relação ao ambiente. O operador de mutação modifica o valor dos genes do indivíduo e visa restaurar o material genético perdido ou não explorado em uma população. Este operador, quando projetado de forma apropriada, pode prevenir a convergência prematura de um algoritmo evolutivo para soluções sub-ótimas e manter a diversidade da população.

Dentre estes algoritmos evolucionários destaca-se o algoritmo de evolução diferencial. A ED foi desenvolvida por Price e Storn (Storn e Price, 1995; Storn, 1997) visando a busca por melhores resultados com uma abordagem um

pouco diferente da utilizada nos algoritmos genéticos e em estratégias evolutivas. Algumas das potencialidades da evolução diferencial têm-se a rapidez de convergência da otimização, a facilidade de implementação e validação.

Basicamente, a ED realiza mutações nos vetores pela adição ponderada de diferenças aleatórias entre eles. A escolha da ED para problemas de otimização é baseada nas características de que a ED:

- é um algoritmo de busca estocástica que é motivado pelos mecanismos de seleção natural;
- é menos susceptível a mínimos (ou máximos) locais, pois busca a solução ótima global pela manipulação de uma população de soluções candidatas, ou seja, busca um número de diferentes áreas simultaneamente no espaço de busca;
- não requer informação de derivadas para o cálculo da função de aptidão;
- manipula diretamente os números de ponto flutuante (fenótipo), diferente dos algoritmos genéticos canônicos ou binários que manipulam cadeias de bits (genótipo), o que diminui o custo computacional necessário para transformação de uma representação genotípica em fenotípica;
- geralmente não necessita de populações grandes para funcionar eficientemente.

Na ED clássica, cada variável (indivíduo) é representada por um valor real (ponto flutuante). A variante implementada neste trabalho foi a ED/rand/1/bin, que é regida pelas seguintes etapas:

Etapa 1: *Iniciar os parâmetros de controle da evolução diferencial*: O projetista deve escolher os parâmetros de controle da ED, tais como tamanho da população ( $M$ ), limites (máximos e mínimos) das variáveis de otimização, taxa de mutação ( $f_m(t)$ ), taxa de cruzamento ( $CR$ ) e o critério de parada do procedimento de otimização.

Etapa 2: *Iniciar o contador de gerações*: Atribuir geração inicial,  $t=1$ .

Etapa 3: *Iniciar a população inicial de indivíduos (soluções)*: Gerar uma população inicial aleatória, com distribuição uniforme, de soluções factíveis à resolução do problema em questão, onde as regras de “reparo” garantem que os valores atribuídos às variáveis estão internas às fronteiras delimitadas pelo projetista.

Etapa 4: *Avaliar os indivíduos da população*: Avaliar a função objetivo (custo) de cada um dos indivíduos da população.

Etapa 5: *Aplicar a operação de mutação (ou operação diferença)*: A mutação é uma operação que adiciona um vetor diferencial para o vetor dos indivíduos da população, de acordo com a equação:

$$z_i(t+1) = x_{i,r_1}(t) + f_m(t)[x_{i,r_2}(t) - x_{i,r_3}(t)] \quad (4)$$

sendo  $i=1,2,\dots,M$  o índice do indivíduo da população;  $j=1,2,\dots,n$  é a posição do indivíduo em um espaço  $n$ -dimensional;  $t$  é a geração (tempo);  $x_i(t) = [x_{i_1}(t), x_{i_2}(t), \dots, x_{i_n}(t)]^T$  consiste da posição do  $i$ -ésimo indivíduo de uma população de  $M$  vetores  $n$ -dimensionais;  $z_i(t) = [z_{i_1}(t), z_{i_2}(t), \dots, z_{i_n}(t)]^T$  é responsável pela posição do  $i$ -ésimo indivíduo de um vetor que sofrerá mutação;  $r_1, r_2$  e  $r_3$  são valores inteiros mutuamente diferentes, selecionados aleatoriamente com distribuição uniforme do conjunto  $\{1, 2, \dots, i-1, i+1, \dots, N\}$ ;  $f_m(t) > 0$  é um parâmetro real denominado de taxa de mutação, que controla a amplificação da diferença entre os dois indivíduos de índices  $r_2$  e  $r_3$  para evitar a estagnação da busca e é usualmente projetado com valores no intervalo  $[0,4; 1,0]$ .

Etapa 6: *Aplicar a operação de cruzamento*: Após a operação de mutação, o cruzamento (ou recombinação) é aplicado a população. O cruzamento é empregado para gerar um novo vetor tentativa ou vetor doador (*trial vector*) pela substituição de certos parâmetros do vetor destino (*target vector*) pelos seus parâmetros correspondentes ao vetor doador, estes gerados aleatoriamente.

Nesta operação, para cada vetor,  $z_i(t+1)$ , um índice  $rnbr(i) \in \{1, 2, \dots, n\}$  é escolhido aleatoriamente usando uma função densidade de probabilidade uniforme, e um vetor denominado de vetor tentativa,  $u_i(t+1) = [u_{i_1}(t+1), u_{i_2}(t+1), \dots, u_{i_n}(t+1)]^T$ . Neste caso é gerado um novo vetor tal que,

$$u_{i_j}(t+1) = \begin{cases} z_{i_j}(t+1), & \text{se } (randb(j) \leq CR) \text{ ou } (j = rnbr(i)), \\ x_{i_j}(t), & \text{se } (randb(j) > CR) \text{ ou } (j \neq rnbr(i)) \end{cases} \quad (5)$$

sendo  $randb(j)$  a  $j$ -ésima avaliação da geração de um número aleatório com distribuição uniforme no intervalo  $[0, 1]$ ; e  $CR$  é a taxa de cruzamento (ou recombinação) no intervalo  $[0, 1]$ . Geralmente, o desempenho do algoritmo de ED depende do projeto de três variáveis: o tamanho da população,  $M$ , a taxa de mutação,  $f_m(t)$ , e a taxa de cruzamento,  $CR$ .

Etapa 7: *Aplicar a operação de seleção*: A seleção é um procedimento em que os “melhores” descendentes (indivíduos filhos) são produzidos. Para decidir se o vetor  $u_i(t+1)$  será (ou não) um membro da população na próxima geração, ele é comparado com o vetor  $x_i(t)$ . Assim considerando que  $F$  denota a função objetivo sob minimização, então:

$$x_i(t+1) = \begin{cases} u_i(t+1), & \text{se } F(u_i(t+1)) < F(x_i(t)), \\ x_i(t), & \text{outros} \end{cases} \quad (6)$$

Neste caso, o custo de cada vetor tentativa  $u_i(t+1)$  é comparado com seu vetor destino  $x_i(t)$ . Se o custo,  $F$ , do vetor destino  $x_i(t)$  possui valor menor que o custo do vetor tentativa, é permitido ao vetor destino continuar na próxima geração. Caso contrário, o vetor destino é substituído pelo vetor tentativa na próxima geração.

Etapa 8: *Verificar se o critério de parada foi atendido*: Atribuir à geração  $t = t + 1$ . Retornar para a Etapa 4 até que o critério de parada seja atendido, usualmente o número de gerações,  $t_{max}$ .

Na figura 1 é apresentado um resumo do fluxo de dados da ED adotada neste trabalho.

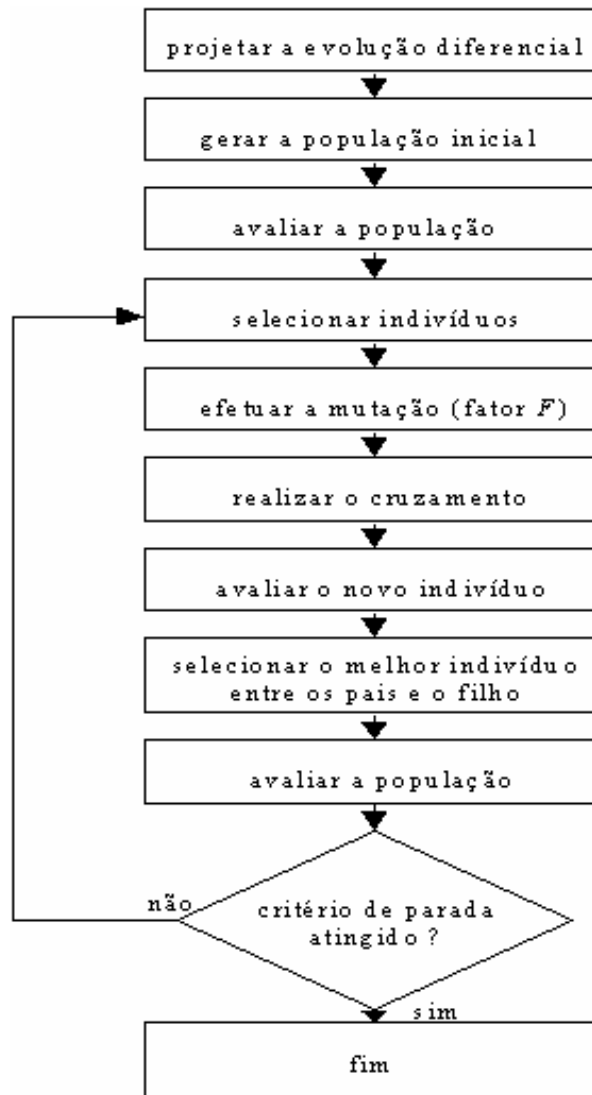


Figura 1 - Fluxograma da ED para otimização de projeto.

### 3. Identificação de um sistema térmico

A identificação de sistemas é uma área do conhecimento que envolve o estudo de formas de modelar e analisar matematicamente fenômenos a partir de dados.

A idéia da identificação de sistemas é permitir a elaboração do modelo matemático de um sistema dinâmico baseado em medidas coletadas pelo ajuste de parâmetros e/ou do modelo matemático, até que a saída do sistema aproxime, de forma adequada, os valores de saída desejada (Johansson, 1993). O procedimento para a identificação de um sistema consiste de quatro etapas básicas, conforme mostra a figura 2.

Um estudo de caso de identificação não-linear de um sistema térmico foi abordado neste trabalho, este disponível na base de dados *Database for the Identification of Systems* (Daisy, 2006).

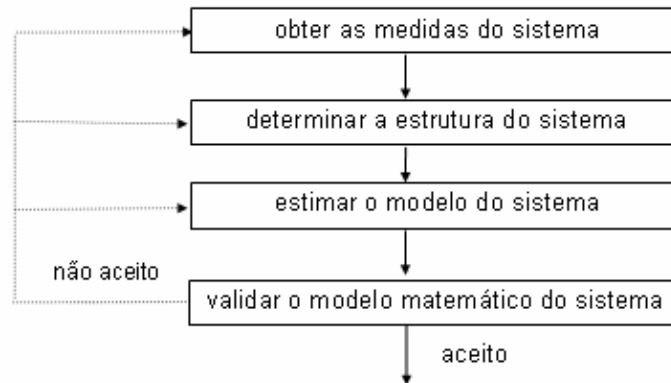


Figura 2 - Procedimento utilizado na identificação de um sistema.

O sistema térmico analisado consiste de uma lâmpada de halógena de 330 Watts suspensa sobre uma placa fina de aço. A saída do sistema consiste da medida de um termopar localizado abaixo da placa (Daisy, 2006). O período de amostragem dos dados é de 2 segundos e o procedimento de identificação, adotado neste trabalho, utiliza como entrada a tensão aplicada à lâmpada (em Volts) e como variável de saída é adotada a temperatura da placa, esta medida em graus Celsius (Dullerud & Smith, 1996).

O experimento foi configurado para utilizar as amostras 1 a 650 (2 a 1300 segundos) para a fase de estimação (otimização da RN-BS) e as amostras 651 a 800 (1302 a 1600 segundos), para a fase de validação (teste) da RN-BS. Os dados de entrada e a saída do sistema térmico avaliado são apresentados na figura 3.

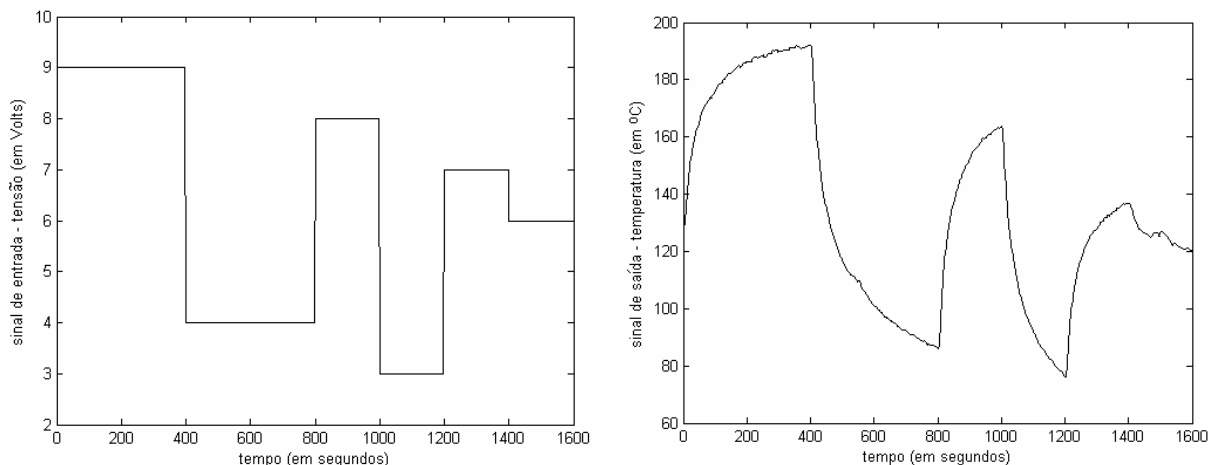


Figura 3 - Dados de entrada e saída do sistema térmico.

O procedimento de análise da identificação através de RN-BS usando ED é dividido nas seguintes etapas: (i) escolha da estrutura da RN-BS para representar o processo, (ii) determinação da otimização da RN-BS através da ED; (iii) estimação dos parâmetros do modelo matemático (*fase de estimação*), e (iv) a validação do modelo matemático (*fase de validação*).

Para identificação foi escolhido um modelo matemático para a representação do processo. A estrutura de modelo testada é o modelo multivariável *NARX* (*Nonlinear AutoRegressive with eXogenous inputs*) em concepção série-paralela com 3 entradas e 1 saída, isto é, as entradas  $[u_1(t-1); y(t-2); y(t-1)]$  e a saída  $\hat{y}(t)$ .

A identificação do sistema térmico é apropriada se um critério de erro definido previamente pelo projetista está entre valores admissíveis às necessidades do projeto. O critério escolhido neste estudo (função custo a ser maximizada) foi o coeficiente de correlação múltipla regido pela equação

$$R^2 = 1 - \frac{\sum_{t=1}^{Na} (y(t) - \hat{y}(t))^2}{\sum_{t=1}^{Na} (y(t) - \bar{y})^2}, \quad (7)$$

onde  $Na$  é o número de amostras avaliado (1300 amostras na fase de estimação e 300 amostras na fase de validação),  $y(t)$  é a saída real do processo,  $\hat{y}(t)$  é a saída estimada pela RN-BS,  $\bar{y}$  é a média das medidas do sistema térmico.

Quando o valor de  $R^2$  é igual a 1,0 indica uma aproximação exata do modelo aos dados medidos do processo. O valor de  $R^2$  entre 0,9 e 1,0 é considerado suficiente para aplicações práticas, principalmente em projetos de identificação e sistemas de controle baseados em modelo.

Adota-se, neste artigo, uma ED com tamanho de população de 10 indivíduos (vetores solução) com critério de parada de 100 gerações. Foram testadas de 3 a 5 B-splines na camada intermediária para cada entrada da RN-BS. Entretanto, os resultados de identificação usando RN-BS com ED foram mais precisos com 3 funções B-splines na RN-BS onde o  $R^2$  obtido foi 0,9998801 (na fase de estimação) e 0,9938608 (na fase de validação), ou seja, estes resultados significam um erro médio quadrático de 0,1768661 (estimação) e 0,1589295 (validação). Nas figuras 4 e 5 são apresentados, respectivamente, os sinais da saída real e estimada do processo e também o e o sinal de erro de predição obtidos usando a RN-BS com 3 funções B-splines.

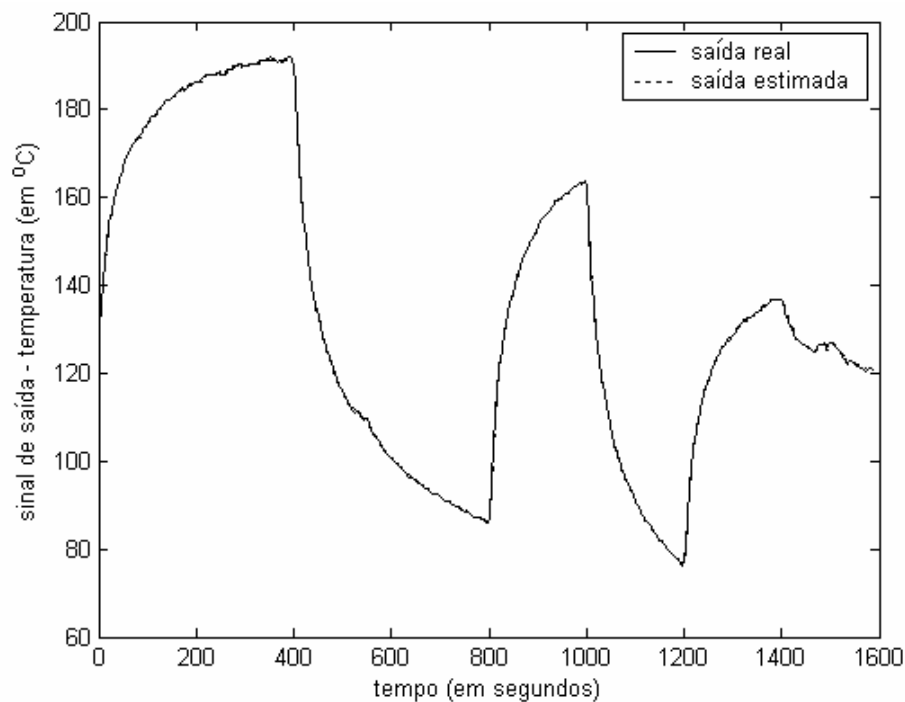


Figura 4 – Saída real e estimada usando 3 funções B-splines para cada entrada na RN-BS.

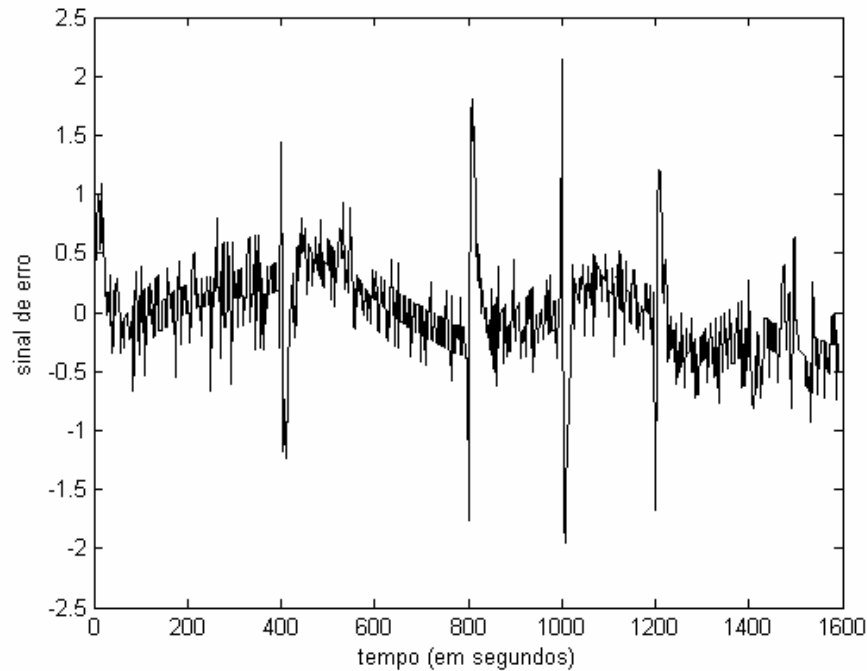


Figura 5 - Erro de predição (dos sinais apresentados na figura 4) usando 3 funções B-splines na RN-BS.

#### 4. Conclusão e futura pesquisa

Neste artigo foi apresentada uma metodologia de otimização baseada em ED para ajustar os pontos de controle (*knots*) de uma RN-BS.

Os resultados de simulação para identificação um passo à frente, de um sistema térmico, demonstraram a eficiência da abordagem de otimização para uma RN-BS. Entretanto, a análise de um melhor compromisso entre precisão, complexidade computacional e ordens dos vetores de entrada deve ser abordada em futura pesquisa.

Em pesquisa futura, deseja-se utilizar procedimentos de otimização com múltiplos objetivos junto ao projeto do modelo de RN-BS. Neste contexto deseja-se determinar um conjunto de soluções de identificação dentro do qual se verifica o compromisso entre os diversos objetivos. O conjunto destas soluções se dá o nome de Pareto-ótimo. Neste contexto, deseja-se projetar uma RN-BS aliado a um procedimento de otimização visando a incorporação de informação da precisão do modelo (usando  $R^2$ ), variância presente no residuo decorrente da diferença entre a saída do sistema e a saída estimada, e o número de parâmetros estimados (graus de liberdade do sistema ou complexidade do modelo).

#### 5. Referências

- An, P. E., Brown, M., Harris, C. J., Lawrence, A. J., Moore, C. G., 1994, "Associative Memory Neural Networks: Adaptive Modeling Theory, Software Implementations and Graphical User Interface", Engineering Applications of Artificial Intelligence, Vol. 7, No. 1, p. 1-21.
- Bäck, T., Fogel, D.B., Michalewicz, Z. (eds.), 1997, "Handbook of Evolutionary Computation", Bristol, Philadelphia: Institute of Physics Publishing, NY, USA, Oxford: Oxford University Press.
- Chen, Y. Q., Moore, K. L., Bahl, V., 2004, "Learning Feedforward Control using a Dilated B-spline Network: Frequency Domain Analysis and Design", IEEE Transactions on Neural Networks, Vol. 15, No. 2, pp. 355-366.
- Coelho, L. S., Villa, L. F., 2005, "Identificação de Processo Não-Linear Usando Rede Neural com Método de Treinamento Híbrido baseado em Filtro de Kalman e Máxima Descida", Anais do 7º Congresso Brasileiro de Redes Neurais, Natal, RN.
- Daisy, Database for the identification of systems, <http://www.esat.kuleuven.ac.be/sista/daisy/> [Acesso 16/02/2006], 2006.
- De Vries, T. J. A., Idema, L. J., Velthuis, W. J. R., 1998, "Parsimonious Learning Feed-Forward Control", Proceedings of European Symposium on Artificial Neural Networks, Bruges, Belgium, pp. 85-90.
- Dullerud, G., Smith, R., 1996, "Sampled Data Model Validation: an Algorithm and Experimental Application", International Journal of Robust and Nonlinear Control, Vol. 6, No. 9/10, pp. 1065-1078.

- Harris, C. J., Moore, C. G., Brown, M., 1993, "Intelligent Control: Aspects of Fuzzy Logic and Neural Nets", Word Scientific.
- Haykin, S., 1996, "Neural Networks", 2nd edition, Prentice-Hall, Upper Saddle River, NJ, USA.
- Johansson, R., 1993, "System Modeling and Identification", Prentice-Hall, Englewood Cliffs, USA.
- Leu, Y. -G., 2004, "Nonlinear System Modeling Using GA-based B-spline Membership Fuzzy-Neural Networks", Proceedings of 2nd International Conference on Autonomous Robots and Agents, Palmerston North, New Zeland, pp. 229-233.
- Lightbody, G., O'Reilly, P., Irwin, G. W., Elly, K., McCormick, J., 1997, "Neural Modeling of Chemical Plant Using MLP and B-Spline Networks", Control Engineering Practice, Vol. 5, No. 11, pp. 1501-1515.
- Storn, R., Price, K., 1995, "Differential Evolution: a Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces", Technical Report TR-95-012, International Computer Science Institute, Berkeley, USA.
- Storn, R., 1997, "Differential Evolution - a Simple and Efficient Heuristic for Global Optimization over Continuous Spaces", Journal of Global Optimization, Vol. 11, No. 4, pp. 341-359.
- Wang, C. H., Wang, W. Y., Lee, T. T., Tseng, P. S., 1995, "Fuzzy B-Spline Membership Function (BMF) and its Applications in Fuzzy-Neural Control", IEEE Transactions on Systems, Man, and Cybernetics, Vol. 25, No. 2, pp. 841-851.
- Whitley, D., 2001, "An Overview of Evolutionary Algorithms: Practical Issues and Common Pitfalls", Information and Software Technology, Vol. 43, pp. 817-831.
- Yiu, K. F. C., Wang, S., Teo, K. L., Tsoi, A. C., 2001, "Nonlinear System Modeling via Knot-Optimizing B-splines Networks", IEEE Transactions on Neural Networks, Vol. 12, No. 5, pp. 1013-1022.

## **HEATING SYSTEM IDENTIFICATION USING B-SPLINES NEURAL NETWORK**

Leandro dos Santos Coelho  
Laboratório de Automação e Sistemas, LAS  
Programa de Graduação em Engenharia de Produção e Sistemas, PPGEPS  
Pontifícia Universidade Católica do Paraná – PUCPR  
Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil  
leandro.coelho@pucpr.br

Viviana Cocco Mariani  
Programa de Pós-Graduação em Engenharia Mecânica  
Pontifícia Universidade Católica do Paraná – PUCPR  
Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil  
viviana.mariani@pucpr.br

### **Abstract**

B-spline neural networks employ basis functions to approximate a continuous, possibly multidimensional, function. B-spline networks can be compared to, e.g., radial basis function networks, which use Gaussian functions as their elemental basis function. B-spline networks are usually trained by gradient-based methods, which may fall into local minimum during the learning procedure. To overcome the problems encountered by the conventional learning methods, the differential evolution provides a stochastic search for global optimization of B-spline networks. A differential evolution approach to adjust the control points (knots) of a B-spline neural network is proposed in this paper. Simulation results for one step ahead identification of a heating system in SISO (Single-Input Single-Output) conception demonstrate the effectiveness of the proposed B-spline neural network.

Keywords: heating system, B-spline neural network, nonlinear identification, differential evolution



## IDENTIFICAÇÃO NÃO-LINEAR DE UM TROCADOR DE CALOR BASEADA EM UM MODELO NEBULOSO INTERPOLATIVO

### Leandro dos Santos Coelho

Laboratório de Automação e Sistemas, LAS  
Programa de Graduação em Engenharia de Produção e Sistemas, PPGEPS  
Pontifícia Universidade Católica do Paraná – PUCPR  
Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil  
leandro.coelho@pucpr.br

### Luciano Alves

Ford, Av. do Taboão, 899, CEP 0955-900, São Bernardo do Campo, SP, Brasil  
la306@uol.com.br

*Resumo. Os sistemas nebulosos são baseados em conceitos de codificação nebulosa da informação e operação com conjuntos nebulosos em vez de números. Os modelos nebulosos, especialmente os modelos nebulosos de Takagi-Sugeno (TS), têm recebido atenção particular na área de identificação de sistemas não-lineares devido a suas capacidades de aproximar comportamento não-linear. Neste artigo uma metodologia para identificação não-linear baseada em modelo nebuloso de TS com método de agrupamento de Gath-Geva é apresentada e validada. A eficiência do método de identificação proposto usando modelo de TS aliado a agrupamento de dado é ilustrada pela identificação de um sistema não-linear. Os resultados de identificação experimental de um trocador de calor confirmam a eficiência e característica de generalização do método de identificação baseado em modelo nebuloso de TS.*

*Palavras chave: trocador de calor, modelo nebuloso, identificação não-linear, agrupamento de dados.*

### 1. Introdução

A identificação de sistemas, seja ela realizada de forma *on-line* ou *off-line*, é uma parte essencial no projeto de sistemas de controle. Em geral, a identificação de sistemas lineares é uma área bem estabelecida e através dos anos, muitas abordagens desenvolvidas já estão consolidadas na literatura (Foss *et al.*, 1998). Entretanto, em muitas situações práticas, a hipótese de linearidade não pode ser mantida devido à existência de não-linearidades ou elementos variantes no tempo. Em tais casos, é difícil aplicar os métodos quantitativos convencionais que descrevam adequadamente as características não-lineares e variantes no tempo de processos industriais.

A identificação de sistemas não-lineares, isto é, sistemas que mostram relações não-lineares entre a(s) entrada(s) e/ou saída(s), é geralmente uma tarefa difícil. Para este propósito, os sistemas nebulosos são uma alternativa para identificação de sistemas. Os sistemas nebulosos, também conhecidos como sistemas de inferência nebulosa, sistemas nebulosos baseados em regras, ou modelos nebulosos, representam a mais importante ferramenta de modelagem baseada na teoria dos conjuntos nebulosos. Um sistema de inferência nebulosa é um mapeamento ou função de um espaço de alternativas de entrada para um espaço de saída. A estrutura básica de um sistema nebuloso possui três componentes conceituais: uma base de regras que contém o conjunto de regras nebulosas, uma base de dados que define as funções de pertinência usadas nas regras nebulosas e um mecanismo de raciocínio que realiza um procedimento de inferência (raciocínio nebuloso) para obter a saída ou conclusão, baseado nas regras e fatos conhecidos (Almeida, 2005).

Neste contexto, destacam-se os sistemas nebulosos de Takagi-Sugeno (TS) (Takagi e Sugeno, 1985; Sugeno e Kang, 1988). O sistema nebuloso de TS tem sido investigado, principalmente por ser aproximador universal de funções com uma dada precisão e por apresentar características atrativas de interpolação e extrapolação numérica.

Neste artigo é proposto um procedimento de otimização de um modelo nebuloso do tipo TS que desacopla a identificação dos antecedentes (funções de pertinência dos antecedentes ou parâmetros da premissa das regras) da estimação da parte conseqüente (parâmetros do conseqüente) das regras nebulosas. O procedimento proposto é decomposto em duas abordagens.

Uma abordagem de agrupamento (*clustering*) usando o método de Gath-Geva (Gath e Geva, 1989) para o ajuste da parte antecedente das regras. A escolha do método de Gath-Geva é devido a suas potencialidades de boa divisão de agrupamentos em relação a outros métodos clássicos, tais como o método *k*-médias e *c*-médias. A outra abordagem emprega o método dos mínimos quadrados (Johansson, 1993) e visa a otimização dos parâmetros do conseqüente das regras nebulosas. Este procedimento de otimização é avaliado na identificação de um trocador de calor.

As próximas seções do artigo são organizadas da seguinte forma. Na seção 2 são descritos os fundamentos do modelo nebuloso de *TS* e o seu procedimento de otimização. A descrição do trocador de calor e a análise dos resultados da aplicação do modelo nebuloso são apresentados nas seções 4 e 5, respectivamente. Na seção 6, a conclusão e alguns comentários sobre pesquisa futura são apresentados.

## 2. Modelos nebulosos

A utilização de conjuntos nebulosos e de um tipo de computação baseado na regra composicional de inferência com seus operadores associados, resulta em sistemas de decisão estruturados no formato de uma base de regras nebulosas, adequados para a implementação de processos dedutivos. Esses sistemas, conhecidos como sistemas de inferência nebulosa ou simplesmente sistemas nebulosos, dependem da especificação de uma série de elementos. Estes elementos incluem a quantidade e o tipo de regras nebulosas, os parâmetros das funções de pertinência, a semântica das regras que participam do raciocínio aproximado e os operadores do mecanismo de inferência utilizado para obter uma saída a partir dos dados de entrada. A utilização de sistemas nebulosos na forma de raciocínio aproximado pode expressar a ambigüidade e a subjetividade presentes no raciocínio humano (Gomide e Pedrycz, 1998).

O uso de sistemas nebulosos é indicado quando (Lin e Lee, 1996): (i) o modelo matemático do sistema não existe, ou quando existe, é de difícil codificação (não temos um modelo matemático *a priori* do processo de incineração); (ii) o modelo matemático é muito complexo para ser rapidamente avaliado em tempo real ou requer muita memória; (iii) no ambiente existe alto nível de ruído e/ou os sensores de medição poucos precisos no sistema; (iv) o processo envolve a interação com um operador humano, ou especialista, preparado para especificar os parâmetros do conjunto de regras a ser utilizado no sistema nebuloso.

Um modelo nebuloso é um conjunto de regras de produção SE-ENTÃO que mapeia entradas em saídas. A estrutura básica de um modelo nebuloso consiste de três componentes conceituais: a *base de regras*, que contém um conjunto de regras nebulosas, a *base de dados* (ou dicionário), que define as funções de pertinência (e seus fatores de escala) vinculadas às regras nebulosas, e o *mecanismo de inferência*, que executa o procedimento de inferência sobre as regras e uma dada condição para derivar uma saída razoável ou conclusão.

### 2.1 Modelo nebuloso de Takagi-Sugeno

A concepção de um modelo nebuloso é baseada em um mecanismo de “dividir para conquistar”, onde os antecedentes das regras nebulosas dividem o espaço de entrada em uma determinada quantidade de regiões nebulosas locais, enquanto os conseqüentes descrevem o comportamento de uma dada região através de vários parâmetros constituintes.

Existem vários modelos de sistemas nebulosos, onde na maioria dos casos, o antecedente é formado por proposições lingüísticas e a distinção entre os modelos se dá no conseqüente das regras nebulosas. Entre os modelos mais conhecidos pode-se destacar: modelo de Mamdani e o modelo *TS*.

O sistema de inferência nebulosa de Mamdani emergiu como uma primeira tentativa de se controlar a caldeira de uma máquina a vapor, através de um conjunto de regras de controle baseadas em variáveis lingüísticas, regras estas obtidas de operadores humanos especializados (Mamdani e Assilian, 1975). A característica básica do modelo de Mamdani é o fato de utilizar conjuntos nebulosos nos conseqüentes das regras nebulosas.

O modelo *TS* foi proposto como resultado de um esforço para se desenvolver de forma sistemática, uma abordagem para a geração de regras nebulosas a partir de dados de entrada-saída (Takagi e Sugeno, 1985). O modelo nebuloso *TS* é composto de uma base de regras nebulosas que particiona o espaço de entrada, denominado de universo de discurso, em regiões nebulosas descritas pelos antecedentes da regra nas quais as funções do conseqüente são válidas (Almeida, 2005).

Diferentes constituintes dos conseqüentes resultam em diferentes tipos de sistemas nebulosos, mas seus antecedentes são sempre os mesmos. Neste artigo, o modelo nebuloso de *TS* é adotado com funções lineares como conseqüentes. As regras do modelo de *TS* apresentam boas características de interpolação e extrapolação com um número reduzido de regras de produção.

O modelo nebuloso de *TS* trata-se de um equivalente funcional da rede neural de base radial. Neste caso, quando a rede neural apresenta o mesmo número de funções de ativação na camada oculta que o número de regras do modelo nebuloso *TS* de ordem zero (Wu e Tam, 1999). Um aspecto relevante do sistema de *TS* é o seu poder de representação, especialmente para a descrição de sistemas complexos. Este modelo nebuloso permite a decomposição de um sistema complexo em subsistemas simples.

Um modelo nebuloso consiste de múltiplas regras, cada regra contendo uma parte premissa (antecedente) e uma parte conseqüente. A parte da premissa é responsável pela especificação de um certo subespaço de entrada pela conjunção de cláusulas nebulosas que contém as variáveis de entrada. A parte conseqüente é um modelo de regressão linear. Os modelos nebulosos de *TS* consistem de regras de produção — *SE* <condição> *ENTÃO* <ação> — que podem ser representadas na forma geral:

$$R_i : SE \ x_1 \acute{E} \ A_{i1} \ E \dots \ E \ x_n \acute{E} \ A_{in} \ ENT\tilde{A}O \ \hat{y}_i = a_i x + b_i, \quad i = 1, 2, \dots, K \quad (1)$$

onde o antecedente *SE* define a parte antecedente (premissa) enquanto as funões da regra *ENT* $\tilde{A}$ *O* constituem-se na parte conseqüente do modelo nebuloso;  $R_i$  é a  $i$ -ésima regra,  $\mathbf{x} = [x_1, \dots, x_n]^T \in \mathcal{X}$  é o vetor das variáveis de entrada das regras (antecedentes),  $A_{i1}, \dots, A_{in}$  são conjunto nebulosos definidas no espaço dos antecedentes, e  $y_i$  é a saída da regra. A variável  $K$  denota o número de regras na base de regras, e a saída agregada do modelo  $\hat{y} \in \mathfrak{S}$  é calculada pela média ponderada dos conseqüentes da regra

$$\hat{y} = \frac{\sum_{i=1}^K \beta_i(\mathbf{x}) \hat{y}_i}{\sum_{i=1}^K \beta_i(\mathbf{x})} \quad (2)$$

onde  $\beta_i(\mathbf{x})$  é o grau de ativação da  $i$ -ésima regra

$$\beta_i(\mathbf{x}) = \prod_{j=1}^n \mu_{A_{ij}}(x_j), \quad i = 1, 2, \dots, K \quad (3)$$

onde  $\mu_{A_{ij}}(x_j) : \mathfrak{R} \rightarrow [0, 1]$  é a função de pertinência do conjunto nebuloso  $A_{ij}$  no antecedente de  $R_i$ . A cada conjunto nebuloso do antecedente  $A_{ij}$  é associada uma função de pertinência  $\mu_{A_{ij}}(x_i)$  descrita por

$$\mu_{A_{ij}}(x_i) = \exp \left[ -\frac{1}{2} \frac{(x_i - m_{ij})^2}{\sigma_{ij}^2} \right] \quad (4)$$

onde  $m_{ij}$  e  $\sigma_{ij}$  são o centro e a largura função de pertinência do tipo Gaussiana, respectivamente. A união de todos estes parâmetros define o conjunto de parâmetros da premissa.

A construção de um modelo nebuloso de *TS* a partir dos dados medidos do processo (ou sistema) a ser identificado envolve duas tarefas, a identificação da estrutura e a identificação de parâmetros. As duas etapas importantes da identificação da estrutura são: (i) a determinação do número de regras SE-ENT $\tilde{A}$ O e (ii) a partição do espaço de entrada para um dado conjunto de conjuntos nebulosos com funções de pertinência. A identificação de parâmetros envolve a identificação dos parâmetros das funções de pertinência e os parâmetros dos conseqüentes funcionais (coeficientes de equações lineares).

Neste artigo, a análise de *cluster* (grupo) é a metodologia empregada na procura de *clusters* correspondentes às classes presentes em dados; um método de agrupamento divide todos as amostras em pequenos sub-grupos e os classifica de acordo com a similaridade entre eles. Uma análise de *cluster* reconstrói a probabilidade de densidade de dados das amostras e então extrai a informação encapsulada nos dados pela extrapolação da pertinência às classes.

Os algoritmos de agrupamento nebulosos permitem um grau de associação para cada elemento em cada grupo. Um elemento pertence a diferentes grupos, com diferentes graus de associação, fornecendo informações mais detalhadas sobre a estrutura de dados. As maiores dificuldades encontradas durante o agrupamento nebuloso de dados são (Gath e Geva, 1989a): (i) o número de grupos nem sempre pode ser definido *a priori*; (ii) as características e as localizações dos centros dos grupos também não são necessariamente conhecidas *a priori*; e (iii) grande variabilidade de formas, densidade, e número de dados em cada grupo. Existem diversas técnicas de agrupamento entre os métodos não-supervisionados, tais como agrupamento nebuloso adaptativo,  $k$ -médias, decomposição de mistura Gaussiana e  $c$ -médias (Yen e Langari, 1999).

Uma alternativa a estes métodos é o algoritmo de Gath-Geva (Gath e Geva, 1989a, 1989b), utilizado em sistemas nebulosos (Abonyi *et al.*, 2005, Abonyi *et al.*, 2002), que emprega uma distância da norma baseado em estimativas nebulosas de probabilidade máxima (FMLE – *Fuzzy Maximum Likelihood Estimates*), proposta por Bezdek e Dunn (1975).

Este algoritmo é derivado pela combinação entre o algoritmo  $k$ -means nebuloso e o algoritmo FMLE. Este método também pode ser denominado de UFP-ONC (*Unsupervised Fuzzy Partition-Optimal Number of Classes*); ele foi desenvolvido com o propósito de classificar os conjuntos nebulosos sem supor um número *a priori* de grupos, podendo detectar vários formatos e agrupamento, densidades e números de pontos em cada grupo (Gath e Geva, 1989a).

Este algoritmo interpreta os dados como variáveis aleatórias, supõe uma distribuição Gaussiana com uma probabilidade *a priori* pertencente a cada grupo. Os métodos para determinar o número de grupos e a partição mais adequada, são utilizados quando o algoritmo não considera um critério de validação específico (Almeida, 2005). Estes

métodos utilizam funções de validação como indicadores da qualidade do resultado da partição, pois fornecem índices de validação (Silva, 2005). O algoritmo Gath-Geva, nos fornece o índice de hipervolume nebuloso e o índice de partição por densidade média.

Neste algoritmo, o cálculo da distância como medida de similaridade entre o dado,  $x$ , e o centro do grupo,  $v$ , tal que:

$$D_{ik}(x_k, v_i) = \frac{\sqrt{\det(F_{wi})}}{\alpha_i} \exp\left(\frac{1}{2}(x_k - v_i^{(l)})^T F_{wi}^{-1}(x_k - v_i^{(l)})\right) \quad (5)$$

Esta distância da norma envolve um termo exponencial e diminui mais rapidamente do que a norma do produto interno.  $F_{wi}$  é a matriz de covariância nebulosa do  $i$ -ésimo grupo é dada por:

$$F_{wi} = \frac{\sum_{k=1}^N (\mu_{ik})^w (x_k - v_i)(x_k - v_i)^T}{\sum_{k=1}^N (\mu_{ik})^w}, \quad 1 \leq i \leq c \quad (6)$$

onde  $w=1$  no algoritmo original FMLE, mas neste caso utilizá-se  $w = 2$ , para que o agrupamento compense o termo exponencial da distância da norma. A probabilidade de seleção de cada grupo é dada por:

$$\alpha_i = \frac{1}{N} \sum_{k=1}^N \mu_{ik} \quad (7)$$

Os graus do conjunto são interpretados como as probabilidades posteriores para selecionar o  $i$ -ésimo grupo dos dados, obtido o  $x_k$  do conjunto de dados de entrada do modelo nebuloso.

Neste contexto, o algoritmo Gath-Geva pode ser resumido pelas seguintes etapas:

Dado um conjunto de dados  $X$ , escolher um número de grupos  $1 < c < N$ , o expoente  $w > 1$  e determinar a tolerância  $\varepsilon > 0$ .

- (i) Iniciar aleatoriamente a matriz de pertinência,  $U$ , composta dos elementos  $u_{ik}$ ;
- (ii) Calcular os centros dos grupos,  $v$ , com a equação:

$$v_i^{(l)} = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^w x_k}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^w}, \quad 1 \leq i \leq c$$

- (iii) Calcular a matriz de covariância:

$$F_i^{(l)} = \frac{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^w (x_k - v_i^{(l)})(x_k - v_i^{(l)})^T}{\sum_{k=1}^N (\mu_{ik}^{(l-1)})^w}, \quad 1 \leq i \leq c$$

- (iv) Calcular a distância como medida de similaridade entre o dado e o centro do grupo:

$$D_{ik}^2(x_k, v_i) = \frac{(2\pi)^{n/2} \sqrt{\det(F_i)}}{\alpha_i} \exp\left(\frac{1}{2}(x_k - v_i^{(l)})^T F_i^{-1}(x_k - v_i^{(l)})\right)$$

com a probabilidade *a priori* dos dados pertencentes ao grupo  $i$ ,

$$\alpha_i = \frac{1}{N} \sum_{k=1}^N \mu_{ik}$$

- (v) Atualizar a matriz de pertinência:

$$\mu_{ik}^{(l)} = \frac{1}{\sum_{j=1}^c (D_{ikA_i}(x_i, v_i) / D_{jk}(x_k, v_j))^{2/(w-1)}}, \quad 1 \leq i \leq c, \quad 1 \leq k \leq N$$

- (vi) Repetir até  $\|U^{(l+1)} - U^{(l)}\| \leq \varepsilon$ , senão voltar para a etapa (ii).

Neste trabalho, o método de ajuste linear para os parâmetros de saída do modelo nebuloso de *TS* é a pseudo-inversa.

### 3. Descrição do trocador de calor

O trocador de calor ou permutador de calor é um equipamento onde dois fluidos com temperaturas diferentes trocam calor através de uma interface metálica. Portanto, um fluido cede calor, sob forma de calor sensível ou calor latente, a outro, mais frio. As aplicações dos trocadores de calor vão desde os aparelhos domésticos até os equipamentos de grande porte nas indústrias químicas e siderúrgicas (Macintyre, 1997).

Os trocadores de calor nem sempre precisam de controle especial da temperatura. Uma vez que os objetivos dos trocadores num processo é possibilitar a recuperação de calor máxima, na existe qualquer razão para restringir seus desempenhos com o uso de controladores. Normalmente basta usar válvulas manuais que devem permanecer bem abertas. Os controles principais são realizados normalmente para o resfriador e para o aquecedor que são adjacentes ao trocador (Kern, 1980).

O trocador de calor analisado neste trabalho consiste de um trocador de calor de fluxo de vapor líquido saturado, onde a água é aquecida por vapor de fluxo saturado pressurizado em um tubo de cobre. A variável de saída é a temperatura do líquido de saída. As variáveis de entrada são: a taxa de transferência de calor do líquido, a temperatura de vapor e a temperatura do líquido de entrada. Neste contexto, a temperatura do vapor e a temperatura do líquido de entrada são mantidas constantes em seus valores nominais (Daisy, 2006). O período de amostragem dos dados é de 1 segundo e o procedimento de identificação, adotado neste trabalho, utiliza como entrada a taxa de transferência de calor do líquido e como variável de saída é adotada a temperatura do líquido de saída. Uma representação do trocador de calor é apresentada na figura 3 (Bittanti e Piroddi, 1996).

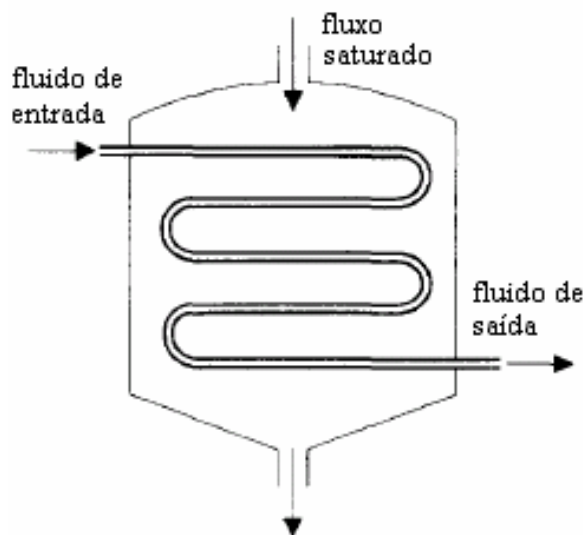


Figura 3 - Representação do trocador de calor.

O experimento foi configurado para utilizar as amostras 1 a 1200 ( $N=1200$ ) para a fase de estimação e as amostras 1201 a 1600, para a fase de validação do modelo nebuloso de *TS*.

Para identificação foi escolhido um modelo matemático para a representação do processo. A estrutura de modelo testada é o modelo multivariável *NARX* (*Nonlinear AutoRegressive with eXogenous inputs*) em concepção série-paralela com 3 entradas e 1 saída, isto é, as entradas  $[u(t-1); y(t-2); y(t-1)]$  e a saída é  $\hat{y}(t)$ , conforme apresentado na figura 4.

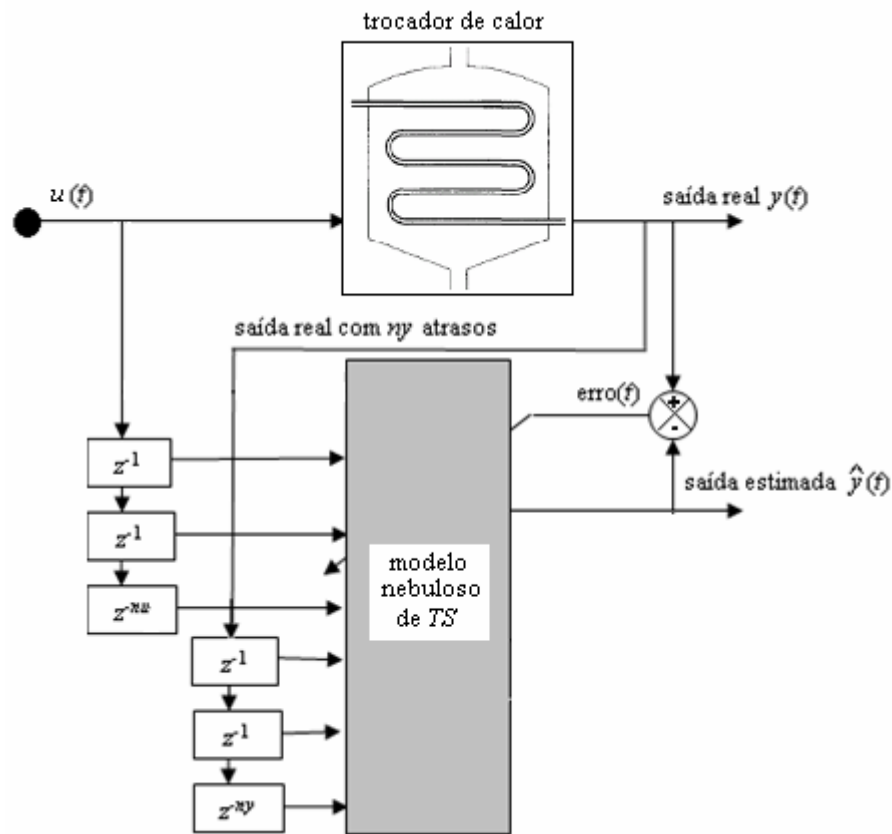


Figura 4 - Identificação usando um modelo nebuloso com estrutura série-paralela.

A identificação do sistema térmico é apropriada se um critério de erro definido previamente pelo projetista está entre valores admissíveis às necessidades do projeto. O critério escolhido neste estudo (função custo a ser maximizada) foi o coeficiente de correlação múltipla regido pela equação

$$R^2 = 1 - \frac{\sum_{t=1}^{Na} (y(t) - \hat{y}(t))^2}{\sum_{t=1}^{Na} (y(t) - \bar{y})^2}, \quad (8)$$

onde  $Na$  é o número de amostras avaliado (2500 amostras na fase de estimação e 1500 amostras na fase de validação),  $y(t)$  é saída real do processo,  $\hat{y}(t)$  é a saída estimada pelo modelo nebuloso de  $TS$ ,  $\bar{y}$  é a média das medidas do sistema térmico. Quando o valor de  $R^2$  é igual a 1,0 indica uma aproximação exata do modelo aos dados medidos do processo. O valor de  $R^2$  entre 0,9 e 1,0 é considerado suficiente para aplicações práticas, principalmente em projetos de identificação e sistemas de controle baseados em modelo (Schaible *et al.*, 1997).

#### 4. Análise dos resultados de identificação do trocador de calor

Após a identificação, um modelo do processo em análise é obtido e deve-se verificar se ele é válido ou não. O conhecimento da finalidade do modelo se faz necessário, para poder julgar se ele incorpora ou não as características requeridas. O modelo será considerado válido se ele incorporar aquelas características do sistema que são fundamentais para a aplicação em questão.

Na tabela 1 são apresentados de forma resumida os resultados das simulações de previsão um passo à frente para a saída do trocador de calor na fase de estimação e validação. Os resultados apresentados nesta tabela mostram os resultados usando o método de agrupamento de Gath-Geva, sendo que também são apresentados os resultados usando o método clássico de agrupamento  $k$ -médias. Detalhes relativos ao método  $k$ -médias são apresentados em Duda e Hart (1973) e Jain e Dubes (1988).

Tabela 1. Resultados das simulações do modelo nebuloso de *TS* na identificação do trocador de calor.

método de agrupamento	funções de pertinência Gaussianas para cada entrada do modelo nebuloso de <i>TS</i>	$R^2$ (estimação)	$R^2$ (validação)	$R^2$ (média harmônica)
Gath-Geva	2	0,8486	0,0205	0,0400
Gath-Geva	3	0,8900	0,7465	0,8120
Gath-Geva	4	0,9589	0,7510	0,8423
Gath-Geva	5	0,9632	0,8398	0,8973
Gath-Geva	6	0,9631	0,8287	0,8909
Gath-Geva	7	0,9630	<b>0,8424</b>	<b>0,8987</b>
Gath-Geva	8	<b>0,9645</b>	0,8369	0,8962
<i>k</i> -médias	2	0,8400	0,0000	0,0000
<i>k</i> -médias	3	0,9101	0,1630	0,2765
<i>k</i> -médias	4	0,9381	0,6454	0,7647
<i>k</i> -médias	5	0,9420	0,7216	0,8172
<i>k</i> -médias	6	0,9434	0,7345	0,8259
<i>k</i> -médias	7	0,9440	0,7396	0,8294
<i>k</i> -médias	8	<b>0,9445</b>	<b>0,7471</b>	<b>0,8343</b>

Nota-se pelos resultados da tabela 1, que os resultados obtidos (analisando-se a medida harmônica de  $R^2$ ) com modelo nebuloso de *TS* aliado ao método de agrupamento de Gath-Geva são superiores aos resultados usando o método clássico de agrupamento *k*-médias.

Nas figuras 5 e 6 são apresentados os melhores resultados (conforme a tabela 1) de identificação do trocador de calor. Neste contexto, o melhor resultado usando o método de agrupamento de Gath-Geva foi com 7 funções de pertinência Gaussianas para entrada do modelo nebuloso com  $R^2$  (média harmônica) = 0,8997. E o melhor resultado do modelo nebuloso de *TS* usando *k*-médias foi  $R^2$  (média harmônica) = 0,8343 com 8 funções de pertinência.

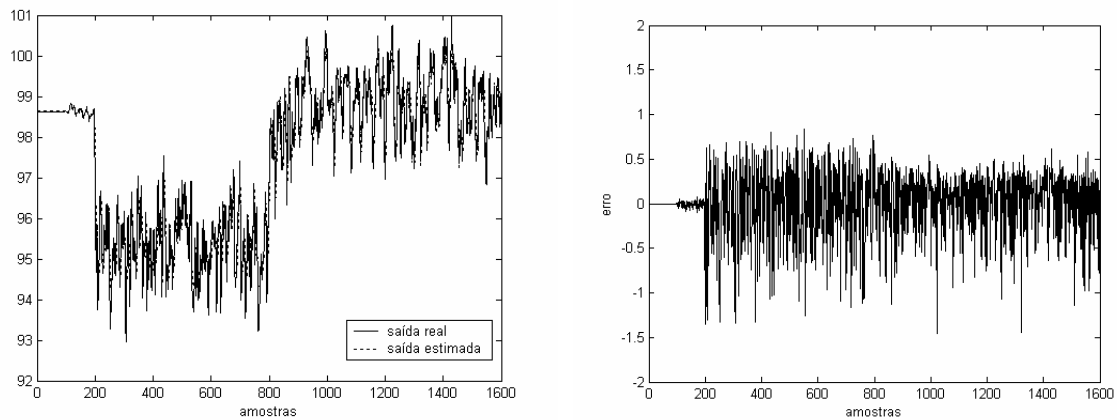


Figura 5 - Saída real e estimada usando modelo nebuloso de *TS* e método de agrupamento de Gath-Geva.

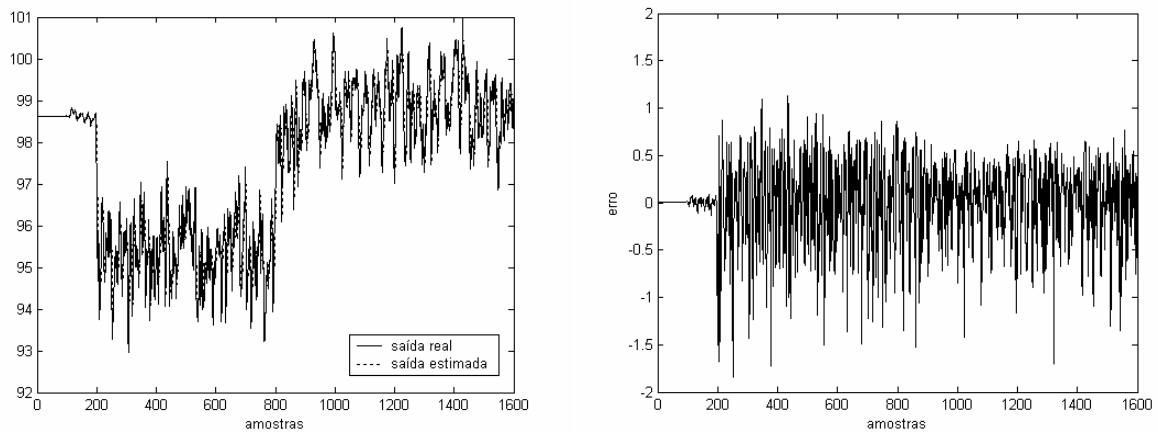


Figura 6 - Saída real e estimada usando modelo nebuloso de *TS* e método de agrupamento *k*-médias.

Pode-se verificar pelas figuras 5 e 6 que o modelo nebuloso de *TS* obtido, representa o comportamento dinâmico da saída do trocador de calor de forma eficiente. Contudo vale salientar que quando trata-se de eficiência, é conveniente levar em conta outros algoritmos que não foram utilizados e também outros aspectos como tempo de processamento do algoritmo e a aplicabilidade do modelo obtido.

## 5. Conclusão e futura pesquisa

Neste artigo é proposto um procedimento de otimização de um modelo nebuloso do tipo *TS* usando método de agrupamento de dados de Gath-Geva para identificação dos antecedentes (premissa das regras do modelo nebuloso) aliado ao método dos mínimos quadrados para o ajuste dos consequentes (problema linear nos parâmetros) das regras nebulosas.

Os resultados obtidos foram satisfatórios. Neste contexto, porém, alguns trabalhos serão realizados para complementar esta pesquisa. Como por exemplo, a utilização de outros algoritmos de otimização e também algoritmos para otimizar os centros dos grupos encontrados pelo algoritmo de agrupamento de Gath-Geva. Uma outra extensão desta pesquisa será, não só a identificação um passo à frente, mas também infinitos passos à frente.

## 6. Referências

- Abonyi, J., Babuska, R., Szeifert, F., 2002, "Modified Gath-Geva Fuzzy Clustering for Identification of Takagi-Sugeno Fuzzy Models", *IEEE Transactions on Systems, Man, and Cybernetics – Part B*, Vol. 32, No. 5, pp. 612-621.
- Abonyi, J., Feil, B., Nemeth, S., Arva, P., 2005, "Modified Gath-Geva Clustering for Fuzzy Segmentation of Multivariable Time-Series", *Fuzzy Sets and Systems*, Vol. 149, No. 1, pp. 39-56.
- Almeida, F. M., 2005, "Identificação Multivariável de um Processo de Incineração de Resíduos Líquidos Utilizando Modelos Nebulosos Takagi-Sugeno", Dissertação de mestrado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, SP.
- Bezdek, J. C., Dunn, J. C., 1975, "Optimal Fuzzy Partitions: a Heuristic for Estimating the Parameters in a Mixture of Normal Distributions", *IEEE Transactions on Computers*, pp. 835-838.
- Bittanti, S., Piroddi, L., 1996, "Nonlinear Identification and Control of a Heat Exchanger: a Neural Network Approach", *Journal of the Franklin Institute*, Vol. 334B, No. 1, pp. 135-153.
- Daisy, Database for the Identification of Systems, <http://www.esat.kuleuven.ac.be/sista/daisy/> [Acesso 16/02/2006], 2006.
- Duda, R. O., Hart, P. E., 1973, "Pattern Classification and Scene Analysis", John Wiley, & Sons, New York, NY.
- Foss, B. A., Lohmann, B., Marquardt, W. A., 1998, "A Field Study of the Industrial Modeling Process", *Journal of Process Control*, Vol. 8, Nos. 5-6, pp. 325-338.
- Gath, I., Geva, A. B., 1989a, "Unsupervised Optimal Fuzzy Clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 7, pp. 773-780.
- Gath, I., Geva, A. B., 1989b, "Fuzzy Clustering for the Estimation of the Parameters of the Components of Mixtures of Normal Distributions", *Pattern Recognition Letters*, Vol. 9, No. 2, pp. 77-86.
- Gomide, F., Pedrycz, W., 1998, "An Introduction to Fuzzy Sets, Analysis and Design", The MIT Press, Cambridge.
- Jain, A.K., Dubes, R.C., 1988, "Algorithms for Clustering Data", Prentice Hall, Englewood Cliffs, USA.
- Johansson, R., 1993, "System Modeling and Identification", Prentice-Hall, Englewood Cliffs, USA.
- Kern, D. Q., 1980, "Processos de Transmissão de Calor", Editora Guanabara Koogan, Rio de Janeiro, RJ.
- Lin, C. T., Lee, C. S. G., 1996, "Neural Fuzzy Systems", Prentice Hall, Englewood Cliffs, USA.
- Macintyre, A. J., 1997, "Equipamentos Industriais e de Processo", Editora LTC, Rio de Janeiro, RJ.
- Mamdani, E.H., Assilian, S., 1975, "An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller", *International Journal of Man-Machine Studies*, Vol. 7, No. 1, pp. 1-13.
- Schaible, B., Xie, H., Le, Y. C. Lee, 1997, "Fuzzy Logic Models for Ranking Process Effects", *IEEE Transactions on Fuzzy Systems*, Vol. 5, No. 4, pp. 545-556.
- Silva, L. R. S., 2005, "Aprendizagem Participativa em Agrupamento Nebuloso de Dados", Dissertação de mestrado, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, SP.
- Sugeno, M., Kang, G. T., 1988, "Structure Identification of Fuzzy Model", *Fuzzy Sets and Systems*, Vol. 28, p. 15-33.
- Takagi, T., Sugeno, M., 1985, "Fuzzy Identification of Systems and its Applications to Modeling and Control", *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 15, No. 1, pp. 116-132.
- Wu, A., Tam, P. K. S., 1999, "A Simplified Model of Fuzzy Inference System Constructed by Using RBF Neurons", *Proceedings of IEEE International on Fuzzy Systems Conference*, Vol. 1, Seoul, Korea, pp. 50-54.



## **NONLINEAR IDENTIFICATION OF A HEAT EXCHANGER BASED ON AN INTERPOLATIVE FUZZY MODEL**

Leandro dos Santos Coelho  
Laboratório de Automação e Sistemas, LAS  
Programa de Graduação em Engenharia de Produção e Sistemas, PPGEPS  
Pontifícia Universidade Católica do Paraná – PUCPR  
Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil  
leandro.coelho@pucpr.br

Luciano Alves  
Ford, Av. do Taboão, 899, CEP 0955-900, São Bernardo do Campo, SP, Brasil  
la306@uol.com.br

### **Abstract**

Fuzzy systems are based on the concept of fuzzy coding of information and operating with fuzzy sets instead of numbers. Fuzzy models, especially Takagi-Sugeno (TS) fuzzy models, have received particular attention in the area of nonlinear identification due to their capability to approximate any nonlinear behavior. In this paper a methodology for nonlinear identification based on TS fuzzy model with Gath-Geva clustering is presented and validated. The efficiency of the proposed identification method using TS fuzzy model with Gath-Geva clustering is illustrated by identification of a nonlinear system. The experimental identification results of a heat exchanger system have confirmed good efficiency and generalization characteristic of the proposed identification method based on TS fuzzy model.

Keywords: heat exchanger, fuzzy model, nonlinear identification, clustering.

# MODELAGEM DE COMPORTAMENTO DINÂMICO DE UM SISTEMA TÉRMICO BASEADA EM REDE NEURAL COM TREINAMENTO USANDO FILTRO DE KALMAN EVOLUTIVO

**Leandro dos Santos Coelho**

Laboratório de Automação e Sistemas, LAS

Programa de Graduação em Engenharia de Produção e Sistemas, PPGEPS

Pontifícia Universidade Católica do Paraná – PUCPR

Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil

leandro.coelho@pucpr.br

*Resumo. As redes neurais função de base radial (RNs-RBF) são modelos de aproximação de funções que podem ser treinados por exemplos para implementar um mapeamento (interpolação) de entrada e saída desejado. O desempenho das RNs-RBF depende do número e centros das funções radiais de base, suas formas, e o método utilizado para aprender o mapeamento de entrada-saída. Este artigo apresenta uma abordagem de RN-RBF usando funções Gaussianas e treinamento com filtro de Kalman baseado em otimização através de evolução diferencial. Os resultados de simulação indicam as potencialidades desta abordagem de aprendizado de RN-RBF para identificação de curto prazo (um passo à frente) da densidade de fluxo de calor (saída do sistema térmico) através de duas camadas de parede (tijolo e camada de isolamento). As entradas do sistema são a temperatura interna e a temperatura externa da parede.*

*Palavras chave: sistema térmico, identificação de sistemas, rede neural, filtro de Kalman, evolução diferencial.*

## 1. Introdução

A identificação de sistemas é o estudo de procedimentos que permitem desenvolver modelos matemáticos a partir de dados e sinais observados (Aguirre, 2000). A área de identificação de sistemas está consolidada para modelos lineares. Os modelos lineares desempenham importante papel no desenvolvimento das técnicas de identificação. Devido à sua simplicidade e à facilidade de obtenção, os modelos lineares são utilizados para aproximar o comportamento de sistemas em determinadas faixas de operação. Os modelos lineares desempenham um importante papel na modelagem de sistemas dinâmicos, quando é possível limitar a operação a regiões de linearidade. Na realidade, todos os sistemas físicos são não-lineares em alguma região de operação. Assim, a presença de comportamento não-linear conduziu ao desenvolvimento de ferramentas que consigam modelar e analisar este comportamento.

Neste contexto, a descrição matemática de sistemas dinâmicos nem sempre é uma tarefa simples na qual princípios básicos podem ser aplicados. Para sistemas complexos, modelagens baseadas em leis elementares para determinar o seu comportamento dinâmico nem sempre são possíveis. Uma alternativa interessante para solucionar este problema é a abordagem de identificação de sistemas. Por esta abordagem, é possível determinar um modelo matemático baseado nas entradas e saídas do sistema. A identificação de sistema é um procedimento que determina um modelo de um sistema dinâmico, com a intenção de prever e/ou compreender seu comportamento dinâmico. Quando existe pouco conhecimento *a priori* do sistema, é comum utilizar-se do modelo do tipo caixa-preta. Este modelo, que é uma estrutura flexível, podendo ser utilizado para aproximar diferentes sistemas não-lineares (Guerra, 2006).

As redes neurais artificiais provaram ser eficientes estruturas não-lineares do tipo caixa-preta em muitas aplicações (Haykin, 1996), e podem ser treinadas usando os dados de entrada e saída observados do próprio sistema. A principal potencialidade na estrutura das redes neurais consiste na sua habilidade de adaptação e aprendizagem. A habilidade de adaptação e aprendizagem pelo ambiente significa que modelos de redes neurais podem lidar com dados imprecisos e situações complexas. Uma rede neural treinada adequadamente tem a habilidade de generalizar quando são apresentadas as entradas que não estão presentes em dados já conhecidos por ela.

Um tipo de rede neural que pode ser aplicado à identificação de sistemas é a rede neural de base radial (RN-RBF, *Radial basis function neural network*). Esta rede compõe uma classe de redes neurais cuja camada oculta é definida por um conjunto de funções de base radial, das quais esta deriva seu nome. As unidades ocultas fornecem um conjunto de funções não-lineares que constituem uma base arbitrária para os padrões de entrada, quando eles são expandidos sobre o espaço oculto. Essas funções são chamadas de funções de base radial. Na estrutura da rede neural de base radial, aplicam-se os algoritmos de agrupamento para encontrar os centros de cada função de base radial, com a intenção de descobrir padrões no espaço de entrada.

Grande parte do interesse pelas RNs-RBF é proveniente do fato delas possuírem fundamentação estatística e são consideradas aproximadores universais de funções com dada precisão (Chen *et al.*, 1991), realizando um mapeamento das entradas de rede de dimensão vetorial  $m$ , em uma saída de dimensão vetorial  $n$  de qualquer função vetorial contínua.

As RNs-RBF apresentam potencialidades de generalização local, em que a interferência no aprendizado é minimizada e o aprendizado é relativamente rápido, devido ao número mínimo de pesos ajustados a cada vetor de treinamento apresentado à rede neural. Na literatura, diferentes abordagens de treinamento de redes neurais usando filtro de Kalman têm sido propostas, tais como Puskorius e Feldkamp (1994), Birgmeier (1995), Simon (2002), Xiong e Zhou (2003), Bai e Guo (2004) e Zhang e Luh (2005).

Na área de ciências térmicas, a literatura da aplicação de métodos de identificação em sistemas térmicos possui alguns trabalhos relevantes, a citar Cunningham (2001) e Haghghat *et al.* (1998). Entretanto, as potencialidades da utilização de redes neurais é ainda pouco explorada e poucos trabalhos têm sido apresentados, como por exemplo os trabalhos de Chen e Chen (2000), Leung *et al.* (2001), Conesa *et al.* (2004), Torrecilla *et al.* (2004), Turias *et al.* (2005), Aththariyakul e Leephakpreeda (2005), Abassi e Bahar (2005), Kumar *et al.* (2006), Aquino e Brigham (2006) e Parlak *et al.* (2006).

A contribuição deste trabalho é avaliar as potencialidades de usar uma abordagem de treinamento de RNs-RBF baseada em filtro de Kalman estendido (EKF, *Extended Kalman Filter*) com otimização usando evolução diferencial (Storn e Price, 1995; Storn, 1997) para identificação do comportamento dinâmico de um sistema térmico. Neste contexto, o estudo de caso testado tem por objetivo a estimativa da densidade de fluxo de calor através de uma parede de duas camadas (camada de tijolos e isolamento).

O artigo é organizado da seguinte forma. A descrição e a abordagem de treinamento da RN-RBF usando evolução diferencial é detalhada na seção 2. Uma breve descrição do sistema térmico e a análise dos resultados obtidos são detalhadas na seção 3. Finalizando, a conclusão e comentários em relação aos resultados obtidos são apresentados na seção 4.

## 2. Rede neural de base radial

Em relação ao cérebro humano, tem-se em torno de  $10^{11}$  neurônios, cada um podendo estar conectado a milhares de outros neurônios, o que facilita o estabelecimento de processamentos paralelos de informações e a formação de redes muito complexas. Assim, as redes neurais artificiais tentam reproduzir as funções das redes biológicas, buscando implementar seu comportamento básico e sua dinâmica (Braga *et al.*, 1998).

As redes neurais artificiais consistem em elementos de processamento altamente interconectados denominados neurônios (artificiais), cada qual possuindo um número de entradas e uma saída. A saída de cada neurônio é determinada como uma função não-linear de uma soma ponderada das entradas, embora operações matemáticas complexas possam ser incluídas. Os neurônios se interconectam através de pesos, os quais são ajustados durante o período de treinamento.

As redes neurais são especificadas pela topologia, características dos neurônios e regras de treinamento. Estas regras de treinamento especificam como o conjunto de pesos iniciais deve ser adaptado para a rede aprender o comportamento de um sistema. Os algoritmos de treinamento (ou aprendizado) de uma rede neurais ajustam iterativamente os pesos das conexões entre os neurônios até que os pares desejados de informações de entrada(s) e saída(s) sejam obtidos e as relações de causa e efeito possam ser estabelecidas. Se as condições mudarem de tal modo que o desempenho do modelo não seja mais adequado, pode-se submeter à rede neural a mais um treinamento sob estas novas condições de entrada(s) e saída(s) para corrigir seu desempenho. Assim sendo, podem-se apresentar dados à rede neural a uma atualização periódica, resultando num modelo auto-ajustável em linha.

De uma forma geral, as principais características das redes neurais artificiais são: (i) aprendizado através de exemplos, (ii) adaptabilidade, (iii) capacidade de generalização, (iv) tolerância a falhas e (v) rápida implementação.

A RN-RBF pode ser considerada como um projeto de rede neural para um problema de ajuste de curva (aproximação) em um espaço de alta dimensionalidade. Desta maneira, aprender é equivalente a encontrar uma superfície, em um espaço multidimensional, que forneça o melhor ajuste para os dados de treinamento, com o critério de “melhor ajuste” sendo medido em um sentido estatístico. Então o processo de generalização equivale a utilizar esta superfície multidimensional para interpolar outros pontos que não pertençam aos dados de treinamento, mas estejam em sua vizinhança.

Para uma RN-RBF, as unidades ocultas fornecem um conjunto de “funções” que consistem em uma “base” arbitrária para os padrões (vetores) de entrada, quando eles são expandidos sobre o espaço oculto: estas funções são denominadas de funções de base radial, daí quais a rede deriva seu nome. Cada uma dessas funções é centrada em uma particular coordenada do espaço multidimensional dos pontos que compõem o espaço de dados de entrada. Cada uma destas coordenadas particulares caracteriza-se por definir o centro de uma (entre várias possíveis) região de maior aglomeração de pontos ou grupos (*clusters*), do espaço de dados de entrada (Haykin, 1996).

A arquitetura da RN-RBF, em sua estrutura mais básica, é constituída de três camadas, com propósitos totalmente diferentes:

- *Camada de entrada*: é composta por nós de fonte (unidades sensoriais) que conectam a rede ao seu ambiente;
- *Camada oculta*: é única, aplica uma transformação não-linear do espaço de entrada para o espaço oculto (na maioria das aplicações, o espaço oculto é de alta dimensionalidade);

- *Camada de saída:* é linear, fornecendo a resposta da rede ao padrão (sinal) de ativação aplicada à camada de entrada.

O processamento da entrada de uma rede neural se realiza de acordo com sua topologia e o tipo da ativação dos neurônios. Em uma RN-RBF clássica, este aspecto consiste na alimentação adiante (*feedforward network*) de uma entrada de dimensão  $m$  em uma camada oculta, que possui  $c$  neurônios ocultos que implementam uma função de ativação baseada em uma medida de similaridade entre os vetores de entrada e os protótipos. Cada neurônio da camada oculta possui um protótipo. As saídas da camada oculta são interligadas às unidades da camada de saída por ramos com pesos sinápticos que ponderam cada saída dos neurônios presentes na camada oculta. Cada unidade da camada de saída implementa uma soma ponderada das saídas da camada oculta. A resposta das camadas de saída forma os elementos do vetor de resposta da RN-RBF.

Uma função de base radial,  $\phi$ , apresenta uma saída simétrica em torno de um centro associado,  $\mu_c$ , tal que  $\phi_c = \phi(\|x - \mu_c\|)$ , onde  $\|\cdot\|$  é a norma vetorial. Uma condição suficiente para uma função ser candidata à função de base radial é a de ser radialmente simétrica e possuir primeira derivada monotônica. A função Gaussiana é o tipo de função de base radial utilizada neste artigo. As funções Gaussianas são caracterizadas por um parâmetro de escala (ou comprimento),  $\sigma$ , e isto é verdade para muitas outras classes de funções de base radial. Tais classes podem ser representadas como:

$$\phi_c(x) = \phi(\|x - \mu_c\|; \sigma) \quad (1)$$

Um conjunto de funções de base radial pode servir como uma base para representação de uma diversificada classe de funções, que são expressas como combinações lineares de funções de base radial, tal que:

$$y(x) = \sum_{j=1}^M \omega_j \phi(\|x - \mu_c\|) \quad (2)$$

Para identificação de sistemas, as RN-RBFs modelam uma função implícita de um fenômeno físico. O modelo neural de identificação é paramétrico do tipo “caixa-preta”, pois a RN-RBF contém parâmetros livres sem significado físico a serem ajustados para que a rede consiga representar a função implícita do processo.

Os vetores de protótipos e pesos sinápticos são os parâmetros da estrutura da RN-RBF. Deve-se ressaltar que os valores desses parâmetros não acrescentam nada de conhecimento a respeito do sistema, porém devem cumprir o determinado: aproximar a resposta do sistema a uma determinada entrada. Para tanto a estimação dos parâmetros das redes neurais em geral, incluindo as RNs-RBF pode ser concebido como um problema de aprendizado com uma “possível” interpretação biológica e também com equivalência matemática (problema de otimização não-linear na estimação dos parâmetros do modelo segundo um critério de otimalidade) (Haykin, 1996).

Alguns pesquisadores têm treinado as RNs-RBF, escolhendo os centros das unidades ocultas aleatoriamente no conjunto de treinamento. Outros têm utilizado métodos não-supervisionados (por exemplo, agrupamento pelo método de  $k$ -médias e fuzzy  $c$ -médias), como também algoritmos supervisionados para a seleção dos centros (Simon, 2002). Os métodos de treinamento separam as tarefas de determinação dos protótipos e a otimização dos pesos. A vantagem dessa abordagem é a rapidez no treinamento da rede como resultado da separação das tarefas e aplicação de um método híbrido de treinamento em alternativa ao treinamento global e a otimização não-linear explícita da descida de encosta (incluindo o algoritmo de retropropagação do erro) (Chen *et al.*, 1991).

A separação das tarefas no treinamento da rede é possível devido a estrutura da RN-RBF e seu tipo de função de ativação que age localmente no espaço de entrada. Foi dada uma considerável atenção na literatura das RNs-RBF em como obter um método de treinamento baseado em técnicas lineares explorando características estruturas das RNs-RBF. Observa-se na figura 1 que se os protótipos são fixados e a função da camada escondida  $g(\cdot)$  conhecida, a rede é linear em seus pesos sinápticos  $w$ . Assim, a estrutura da RN-RBF possibilita a separação de tarefas no processo de aprendizado, com a vantagem que na estimação dos pesos  $w$  o treinamento da rede se torna uma tarefa rápida de ser resolvida através de métodos do tipo mínimos quadrados.

Segundo Chen *et al.* (1991), “o problema do mal-condicionamento nas estratégias híbridas pode, por exemplo, ocorrer devido aos centros estarem próximos, resultando em uma dependência linear entre ambos os centros”. Assim o mal-condicionamento numérico da rede pode ser contornado pelos métodos de agrupamento (*clustering*) que faz a escolha dos protótipos como sendo representantes de populações encontradas nos dados de entrada por aprendizado não supervisionado. Observa-se nos métodos que exploram a localidade da RN-RBF que somente as entradas similares aos protótipos serão capazes de contribuir de forma efetiva na resposta da rede e também não usam toda informação fornecida dos exemplos de entrada(s) e saída do conjunto de treinamento na fase de estimação dos parâmetros, acarretando em uma pobre generalização no aprendizado da rede. Neste contexto, o algoritmo do filtro de Kalman é descrito na subseção apresentada a seguir.

## 2.1. Filtro de Kalman estendido

Criado por Rudolph Emil Kalman em 1960, o filtro de Kalman foi desenvolvido inicialmente como uma solução recursiva para filtragem linear de dados discretos (Kalman, 1960). Para isto, utiliza equações matemáticas que implementam um estimador preditivo de estados, buscando corrigir iterativamente a resposta de um determinado sistema através de múltiplas variáveis relacionadas a ele. Suas áreas de aplicação são muito diversificadas, tais como: processamento de imagem, supervisores de eventos discretos, processamento de sinais, sistemas de inferência, entre outros (Côrrea, 2005).

O filtro de Kalman original (Kalman, 1960; Kalman e Bucy, 1961) foi concebido para ser aplicado em sistemas com dinâmica linear. Para um sistema dinâmico linear com ruído branco de processo de medição, o filtro de Kalman é um estimador ótimo dos estados do sistema. A fim de tornar a aplicação do filtro de Kalman em sistemas não-lineares, pesquisadores propuseram o filtro de Kalman estendido (Kuchner, 1967; Gordon *et al.*, 1993), o que se mostrou eficiente para muitos sistemas cuja não-linearidade é pouco significativa. Para sistemas não-lineares com ruído colorido, o filtro de Kalman pode ser estendido pela linearização da matriz de transição não-linear da descrição em variáveis de estado do sistema em torno dos parâmetros estimados correntes (Haykin, 2001). Neste contexto, o filtro de Kalman estendido atualiza os parâmetros de forma a ser consistente com todos os dados medidos do processo e geralmente converge em poucas interações (Simon, 2002; Welch e Bishop, 1995).

A formulação dos filtros de Kalman está fundamentada na representação em espaço de estados de um sistema dinâmico linear discreto de dimensão finita, ou seja, o filtro de Kalman é descrito em termos de equações a diferenças, porém se o sistema for não-linear, pode-se estender o uso do filtro de Kalman com um procedimento de linearização de um sistema da forma

$$\begin{aligned}\theta_{k+1} &= f(\theta_k) + \omega_k \\ y_k &= h(\theta_k) + v_k\end{aligned}\tag{3}$$

onde o vetor  $\theta_k$  é o estado do sistema no instante de tempo  $K$ ,  $\omega_k$  é o ruído provocado pela própria dinâmica do processo,  $y_k$  é o vetor de observação,  $v_k$  é o ruído de observação (ou de medida), e  $f(\cdot)$  e  $h(\cdot)$  são funções vetoriais não-lineares e representam esse sistema possivelmente variante no tempo.

A idéia básica do filtro de Kalman estendido consiste em linearizar a equação (3) a cada instante de tempo em torno do estado estimado recente. Se as não-linearidades são suficientemente suaves, pode-se fazê-lo aplicando a aproximação de Taylor de primeira ordem nas funções não-lineares  $v$  e  $h(\cdot)$  em torno do estado estimado  $\hat{\theta}$  para obter

$$f(\theta_k) = f(\hat{\theta}_k) + F_k(\theta_k - \hat{\theta}_k) + \text{termos de ordem elevada}\tag{4}$$

$$h(\theta_k) = h(\hat{\theta}_k) + H_k^T(\theta_k - \hat{\theta}_k) + \text{termos de ordem elevada}\tag{5}$$

sendo adotada a seguinte notação para as derivadas parciais:

$$F_k = \frac{\partial f(\theta)}{\partial \theta} \Big|_{\theta = \hat{\theta}_k}\tag{6}$$

$$H_k^T = \frac{\partial h(\theta)}{\partial \theta} \Big|_{\theta = \hat{\theta}_k}\tag{7}$$

Desconsiderando os termos de ordem mais elevada na equação (5), um sistema linear aproximado é obtido de (4), onde

$$\theta_{k+1} = F(\theta_k) + \phi_k\tag{8}$$

$$y_k = H_k^T(\theta_k) + \varphi_k\tag{9}$$

Neste caso,  $\phi_k$  e  $\varphi_k$  são definidos por

$$\phi_k = f(\hat{\theta}_k) - F_k(\hat{\theta}_k) \quad (10)$$

$$\varphi_k = h(\hat{\theta}_k) - H_k^T(\hat{\theta}_k) \quad (11)$$

A obtenção da aproximação linear do sistema representado pela equação (4) é o primeiro passo para utilização do filtro de Kalman. O problema do filtro de Kalman consiste em resolver simultaneamente e de maneira ótima as equações de processo e de medida para um estado desconhecido.

Pode-se formular o problema de filtragem de um sistema linear achando um valor ótimo de  $\hat{\theta}_k$ , tal que o sistema deve satisfazer as restrições impostas pelos teoremas do estimador médio condicional e o do princípio da ortogonalidade. Finalmente, pode-se mostrar que a estimação desejada  $\hat{\theta}_k$  pode ser obtida recursivamente pelas equações do filtro de Kalman:

$$\hat{\theta}_k = f(\hat{\theta}_{k-1}) + K_k[y_k - h(\hat{\theta}_{k-1})] \quad (12)$$

$$K_k = P_k H_k (R + H_k^T P_k H_k)^{-1} \quad (13)$$

$$P_{k+1} = F_k (P_k - K_k H_k^T P_k) F_k^T + Q \quad (14)$$

onde  $K_k$  é o ganho de Kalman e  $P_k$  a matriz de covariância do erro de estimação do estado  $\theta_k$ . As matrizes  $Q$  e  $R$  são as matrizes de covariância dos ruídos artificiais  $\omega_k$  e  $v_k$ , respectivamente. Como o sistema descrito nas equações (8) e (9) é uma aproximação de um sistema não-linear, o filtro não é ótimo e a estimação é somente uma aproximação da média condicional.

O problema de otimização da matriz de pesos sinápticos  $W$  e as coordenadas dos vetores protótipos  $v_j$  sob o ponto de vista da estimação ótima de parâmetros com filtro de Kalman de uma RN-RBF pode ser tratado como um problema de minimização do tipo mínimos quadrados ponderados. Neste caso, para aplicar o EKF, faz-se com que os elementos da matriz de pesos  $W$  e os elementos dos vetores de protótipos  $v_j$  constituam o estado do sistema discreto não-linear em que o filtro de Kalman é aplicado.

## 2.2. Evolução diferencial

A ED é um paradigma da computação evolutiva (ou evolucionária) desenvolvido por Rainer Storn e Kenneth Price (Storn e Price, 1995; Storn, 1997) para problemas de otimização não-linear contínua. Basicamente, a ED realiza mutações nos vetores pela adição ponderada de diferenças aleatórias entre eles. A escolha da ED para problemas de otimização é baseada nas características de que a ED:

- é um algoritmo de busca estocástica que é motivado pelos mecanismos de seleção natural;
- é menos susceptível a mínimos (ou máximos) locais, pois busca a solução ótima global pela manipulação de uma população de soluções candidatas, ou seja, busca um número de diferentes áreas simultaneamente no espaço de busca;
- não requer informação de derivadas para o cálculo da função de aptidão;
- manipula diretamente os números de ponto flutuante (fenótipo), diferente dos algoritmos genéticos canônicos ou binários que manipulam cadeias de bits (genótipo), o que diminui o custo computacional necessário para transformação de uma representação genotípica em fenotípica;
- geralmente não necessita de populações grandes para funcionar eficientemente.

Na ED clássica, cada variável (indivíduo) é representada por um valor real (ponto flutuante). A variante implementada neste trabalho foi a ED/rand/1/bin.

## 3. Descrição do sistema térmico e análise dos resultados de identificação usando RN-RBF

A seguir são apresentados os detalhes do sistema térmico a ser identificado e após são mostrados e comentados os resultados de identificação usando RN-RBF com filtro de Kalman e evolução diferencial.

### 3.1. Descrição do sistema térmico

A obtenção de uma estimativa apropriada para o comportamento dinâmico da densidade de fluxo de calor através de uma parede de duas camadas (camada de tijolos e isolamento) é o objetivo do procedimento de identificação baseado

em modelo neural deste trabalho. As entradas do sistema são a temperatura interna ( $u_1(t)$ ) e a temperatura externa ( $u_2(t)$ ) da parede. O sinal de saída  $y(t)$  consiste da densidade de fluxo de calor através de uma parede (Bloem, 1994). Neste caso, os dados (1680 amostras) usados no procedimento de identificação estão disponíveis na base de dados DaISy (De Moor, 2006). É importante mencionar que o sistema térmico, avaliado neste trabalho, é um sistema teste para métodos de identificação. Detalhes do sistema térmico podem ser encontrados em Bloem (1994) e De Moor (2006).

### 3.2. Análise dos resultados de identificação

O procedimento de análise da identificação através de RN-RBF é dividido nas seguintes etapas: (i) escolha da estrutura da RN-RBF para representar o sistema a ser identificado, (ii) otimização da RN-RBF usando agrupamento de dados e pseudo-inversa; (iii) estimação dos parâmetros do modelo matemático (*fase de estimação*), e (iv) a validação do modelo matemático (*fase de validação*).

Na procura de um modelo “ideal”, devem ser observadas as seguintes questões: se o modelo encontrado é flexível o suficiente e se o modelo é desnecessariamente complexo, ou em outras palavras, se o modelo é confiável para o propósito desejado. Existem diversas maneiras de se realizar a validação de um modelo. Neste trabalho, o índice de desempenho utilizado é o erro médio quadrático (*Mean Squared Error*, MSE).

Para identificação foi escolhido um modelo matemático para a representação do processo. A estrutura de modelo testada é o modelo *NARX (Nonlinear AutoRegressive with eXogenous inputs)* em concepção série-paralela com 2 entradas ( $[u_1(t-1); u_2(t-1)]$ ) e 1 saída estimada,  $\hat{y}(t)$ , para a densidade de fluxo de calor.

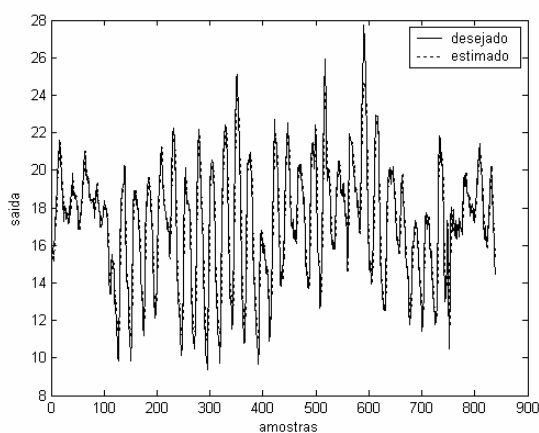
Na tabela 1 é apresentado o resultado para o projeto do filtro de Kalman (executado apenas em uma época de aprendizado) para o treinamento da RN-RBF. O espaço de busca usado na ED foi entre 0,1 e 2000 para as variáveis da diagonal principal de  $P$ ,  $Q$  e  $R$ . Neste caso a função de aptidão (*fitness*) usada na ED foi o *MSE*. Foram usadas as amostras de 1 a 840 na fase de estimação da RN-RBF, e as amostras de 841 a 1680 na fase de validação.

Os valores dos termos da diagonal principal das matrizes  $P$ ,  $Q$  e  $R$  foram obtidos usando uma abordagem de ED com uma população de 5 e critério de parada de 20 gerações. O melhor resultado obtido pela ED para o projeto do filtro de Kalman usado na RN-RBF foi  $P=238,2820$ ,  $Q=125,3487$  e  $R=15,2179$  usando uma RN-RBF com 3 funções Gaussianas (grupos) na camada oculta.

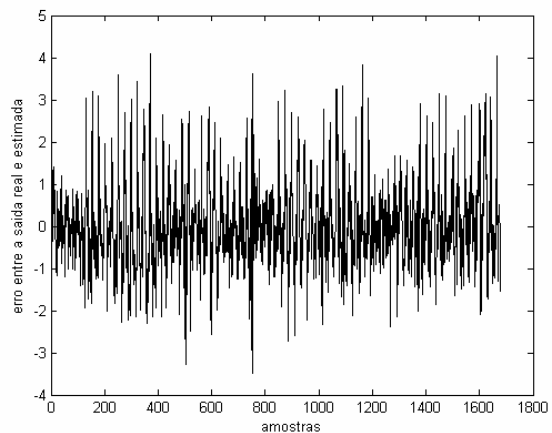
Tabela 1. Resultados das simulações para previsão densidade de fluxo de calor através de uma parede de duas camadas.

Gaussianas na camada oculta (nº de grupos)	MSE ( $\times 10^{-3}$ ) (fase de estimação)	MSE ( $\times 10^{-3}$ ) (fase de validação)
2	0,4139	0,0238
<b>3</b>	<b>0,3983</b>	<b>0,0154</b>
4	0,3993	0,0147
5	0,5110	0,0975

Na figura 1 é apresentado o sinal de erro e saída estimada para a densidade de fluxo de calor através de uma parede de duas camadas, respectivamente, para o melhor resultados apresentado na tabela 1 obtidos pela RN-RBF com 4 funções Gaussianas na camada oculta. Nota-se que a RN-RBF apresentou um índice de desempenho apropriado para modelar o comportamento dinâmico da densidade de fluxo de calor com MSE ( $\times 10^{-3}$ )= 0,3983 (fase de estimação) e MSE ( $\times 10^{-3}$ )=0,1054 (fase de validação).



(a) densidade de fluxo de calor



(b) sinal de erro de estimativa

Figura 1. Saída estimada para a densidade de fluxo de calor usando RN-RBF (4 funções Gaussianas na camada oculta).

#### 4. Conclusão e futura pesquisa

As RNs-RBF têm como uma das principais características agirem como se fossem aproximadores universais de funções, podendo ser utilizadas com êxito para identificar sistemas complexos. Este trabalho apresentou uma abordagem de RN-RBF baseada em treinamento por filtro de Kalman híbrido com evolução diferencial. Os resultados de simulação indicam as potencialidades desta abordagem de rede neural para identificação de curto prazo (um passo à frente) da densidade de fluxo de calor em uma parede.

O filtro de Kalman possibilitou lidar com informações ruidosas do sistema dinâmico, utilizando informações matemáticas para estimar o estado atual do ambiente no qual está inserido. Uma dificuldade no projeto do filtro de Kalman foi o projeto das matrizes  $P$ ,  $Q$  e  $R$ . Neste contexto, neste trabalho, utilizou-se a técnica de evolução diferencial para otimizar os termos da diagonal principal das matrizes de  $P$ ,  $Q$  e  $R$ . Os resultados de empregar filtro de Kalman com evolução diferencial no projeto da RN-RBF obtidos foram satisfatórios em termos de MSE.

Em pesquisa futura, deseja-se realizar uma análise estatística de diferentes abordagens de algoritmos de otimização para o projeto do filtro de Kalman em aplicações de previsão de séries temporais não-lineares de sistemas térmicos com características complexas de atraso de transporte e não-linearidades.

#### 5. Referências

- Abbassi, A., Bahar, L., 2005, "Application of Neural Network for the Modeling and Control of Evaporative Condenser Cooling Load", Applied Thermal Engineering, Vol. 25, No. 17-18, pp. 3176-3186.
- Aquino, W., Brigham, J. C., 2006, "Self-learning Finite Elements for Inverse Estimation of Thermal Constitutive Models", International Journal of Heat and Mass Transfer, Vol. 49, No. 15-16, pp. 2466-2478.
- Aguirre, L. A., 2000, "Introdução à Identificação de Sistemas – Técnicas Lineares e Não-lineares", Editora UFMG, elo Horizonte, MG.
- Atthajariyakul, S., Leephakpreeda, T., 2005, "Neural Computing Thermal Comfort Index for HVAC Systems", Energy Conversion and Management, Vol. 46, No. 15-16, pp. 2553-2565.
- Bai, L.; Guo, P., 2004, "A Novel Approach to Stellar Recognition by Combining EKF and RBF-net", Advances in Neural Networks, International Symposium on Neural Networks, Dalian, China, Lecture Notes in Computer Science, vol. 3173/2004, Springer, pp. 1002-1007, 2004.
- Birgmeier, M., 1995, "A Fully Kalman-trained Radial Basis Function Network for Nonlinear Speech Modeling", Proceedings of IEEE International Conference on Neural Networks, Vol. 1, pp. 259-264.
- Bloem, J. (ed.), 1994, "System Identification Competition, Benchmark Tests for Estimation Methods of Thermal Characteristics of Buildings and Building Components", Joint Research Centre, Ispra, Italy.
- Braga, A. P., Ludermir, T. B., Carvalho, A. C. P. L. F., 1998, Fundamentos de Rede Neurais Artificiais, 11ª Escola de Computação, SBC.
- Chen, S.; Cowan, C.; Grant, P., 1991, "Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks", IEEE Transactions on Neural Networks, Vol. 2, No. 2, pp. 302-309, 1991.
- Chen, Y., Chen, Z., 2000, "A Neural-Network-based Experimental Technique for Determining z-transfer Function Coefficients of a Building Envelope", Building and Environment, Vol. 35, pp. 181-189.
- Conesa, J. A., Caballero, J. A., Reyes-Labarta, J. A., 2004, "Artificial Neural Network for Modelling Thermal Decompositions", Journal of Analytical and Applied Pyrolysis, Vol. 71, No. 1, pp. 343-352.
- Côrrea, L. G. C., 2005, "Inferência da Qualidade de Produtos de Destilação utilizando Redes Neurais Artificiais e Filtro de Kalman Estendido", Dissertação de mestrado, Programa de Pós-Graduação em Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ.
- Cunningham, M. J., 2001, "Inferring Ventilation and Moisture Release Rates from Field Psychrometric Data only System Identification Techniques", Building and Environment, Vol. 36, pp. 129-138.
- De Moor B.L.R. (ed.), 2006, "DaISy: Database for the Identification of Systems", Department of Electrical Engineering, ESAT/SISTA, K.U.Leuven, Belgium, URL: <http://homes.esat.kuleuven.be/~smc/daisy/>, 10/06/2006. [Used dataset: Heat flow density through a two layer wall, section *Thermic Systems*, code 96-011]
- Gordon, N. J., Salmond, D. J., Smith, A. F. M., 1993, "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation", IEE Proceedings-F, vol. 140, pp. 107-113.
- Guerra, F. A., 2006, "Estudo e Análise de Métodos de Agrupamento para o Treinamento de Redes Neurais de Base Radial em Aplicações de Identificação de Sistemas Não-lineares", Dissertação de mestrado, PPGEPS, Pontifícia Universidade Católica do Paraná, Curitiba, PR.
- Haghighat, F., Fazio, P., Zmeureanu, R. A., 1998, "A Systematic Approach for Derivation of Transfer Function Coefficients of Buildings from Experimental Data", Energy and Buildings, Vol. 12, pp. 101-111.
- Haykin, S., 1996, Neural Networks, 2nd edition, Prentice-Hall, Upper Saddle River, New Jersey, 1996.
- Haykin, S., 2001, Kalman Filtering and Neural Networks, John Wiley & Sons, Inc, New York, 2001.
- Kalman, R. E., 1960, "A New Approach to Linear Filtering and Prediction Problems", Transactions of the ASME – Journal of Basic Engineering, Vol. 83, pp. 35-45.



- Krauss, G.K., Kindangen, J. I., Depecker, P., 1997, "Using Artificial Neural Networks to Predict Interior Velocity Coefficients", *Building and Environment*, Vol. 32, No. 4, pp. 295-303.
- Kumar, R., Kaushik, S. C., Garg, S. N., 2006, "Heating and Cooling Potential of an Earth-to-air Heat Exchanger Using Artificial Neural Network", *Renewable Energy*, Vol. 31, No. 8, pp. 1139-1155.
- Kuchner, H. J., 1967, "Approximations to Optimal Nonlinear Filters", *IEEE Transactions on Automatic Control*, Vol. 12, No. 5, pp. 546-556.
- Leung, A. W. T., Tam, C. M., Liu, D. K., 2001, "Comparative Study of Artificial Neural Networks and Multiple Regression Analysis for Predicting Hoisting Times of Tower Cranes", *Building and Environment*, Vol. 36, pp. 457-467.
- Parlak, A., Islamoglu, Y., Yasar, H., Egrisogut, A., 2006, "Application of Artificial Neural Network to Predict Specific Fuel Consumption and Exhaust Temperature for a Diesel Engine", *Applied Thermal Engineering*, Vol. 26, No. 8-9, pp. 824-828.
- Puskorius, G.V.; Feldkamp, L.A., 1994, "Neurocontrol of Nonlinear Dynamical Systems with Kalman Filter Trained Recurrent Networks", *IEEE Transactions on Neural Networks*, Vol. 5, No. 2, pp. 279-297, 1994.
- Rössler, O. E., 1976, "An Equation for Continuous Chaos", *Physical Letters*, Vol. 35A, pp. 397-398, 1976.
- Simon, D., 2002, "Training Radial Basis Neural Networks with the Extended Kalman Filter", *Neurocomputing*, Vol. 48, No. 1-4, pp. 455-475, 2002.
- Storn, R., 1997, "Differential Evolution — A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces", *Journal of Global Optimization*, Vol. 11, No. 4, pp. 341-359.
- Storn, R.; Price, K., 1995, "Differential Evolution: A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces", Technical Report TR-95-012, International Computer Science Institute, Berkeley.
- Torrecilla, J. S., Otero, L., Sanz P. D., 2004, "A Neural Network Approach for Thermal/pressure Food Processing", *Journal of Food Engineering*, Vol. 62, No. 1, pp. 89-95.
- Turias, I. J., Gutiérrez, J. M., Galindo, P. L., 2005, "Modelling the Effective Thermal Conductivity of an Unidirectional Composite by the Use of Artificial Neural Networks", *Composites Science and Technology*, Vol. 65, No. 3-4, pp. 609-619.
- Welch, G., Bishop, G. An Introduction to the Kalman Filter, Technical Report TR- 95-041, Department of Computer Science, University of North Carolina at Chapel Hill, NC, USA, 1995.
- Xiong, S.-S.; Zhou, Z.-Y., 2003, "Neural Filtering of Colored Noise based on Kalman Filter Structure", *IEEE Transactions on Instrumentation and Measurement*, Vol. 52, No. 3, pp. 742-747.
- Zhang, L.; Luh, P.B., 2005, "Neural Network-based Market Clearing Price Prediction and Confidence Interval Estimation with an Improved Extended Kalman Filter Method", *IEEE Transactions on Power Systems*, Vol. 20, No. 1, pp. 59-66.

## **DYNAMIC BEHAVIOR MODELING OF A THERMAL SYSTEM BASED ON NEURAL NETWORKS WITH TRAINING USING EVOLUTIONARY-KALMAN FILTER**

Leandro dos Santos Coelho  
Laboratório de Automação e Sistemas, LAS  
Programa de Graduação em Engenharia de Produção e Sistemas, PPGEPS  
Pontifícia Universidade Católica do Paraná – PUCPR  
Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil  
leandro.coelho@pucpr.br

### **Abstract**

Radial basis function neural networks (RBF-NNs) are function approximation models that can be trained by examples to implement a desired input-output mapping (interpolation). The performance of RBF-NNs depends on the number and centers of the radial basis functions, their shapes, and the method used for learning the input-output mapping. This paper presents a RBF-NN model using Gaussian functions and Kalman filter training based on differential evolution optimization. The simulation results indicate the potentialities of this RBF-NN learning approach to one step ahead identification of heat flow density (output of thermal system) through a two layer wall (brick and insulation layer). The system inputs are the internal and external temperature of the wall.

Keywords: thermal system, systems identification, neural network, Kalman filter, differential evolution

## THREE DIMENSIONAL SIMULATION OF HEAT TRANSFER WITHIN FURNACES AND REACTOR TUBES FOR THE THERMAL CRACKING OF EDC

**Roberto Janny Teixeira Junior**

Centro de Tecnologias Ambientais e Energéticas - Instituto de Pesquisas Tecnológicas de São Paulo - IPT. Avenida Prof. Almeida Prado, 532. Cid. Universitária, CEP 05508-901 São Paulo/SP - Brazil.  
rjanny@ipt.br

**Francisco Domingues Alves de Sousa**

Centro de Tecnologias Ambientais e Energéticas - Instituto de Pesquisas Tecnológicas de São Paulo - IPT.  
fdasousa@ipt.br

**Sandro da Silva Aguiar**

Braskem S.A. - Unidade Vinílicos - Pólo Cloroquímico de Alagoas, Marechal Deodoro/AL - Brazil.

**Abstract.** *This work is an application of the numerical code ERHESICS for thermal behavior simulation of a pyrolysis furnace based on Vinyl Unit of Braskem S/A. This furnace engenders thermal cracking of EDC (ethylene dichloride) into its monomer VCM (vinyl chlorine monomer). ERHESICS aims to predict the effect of operational parameters of wall burners on load's (tubular reactor skin) overall heat flux. To predict thermal performance of a furnace (where thermal radiation's the more significant way of heat transfer), accurate calculus of radiant transfer between combustion products, furnace enclosure and load must be achieved. For Simulation of Brakem's thermal cracking furnace, thermal radiation was approached using Hottel's integral zone method, assuming participant medium as a weighted sum of gray gases. For a known flue gas flow pattern, a coupled simulation of thermal radiation, convection and conduction was carried out. Basic blocks (direct exchange areas) for formulae of zone method were developed specially for rectangular furnaces, using special transformation of variables to reduce computation effort. Exchange areas were computed using a Monte Carlo-like technique. Simulation results were compared to a real case where experimental data's available to verify model predictions. Good accordance was obtained, with maximum relative deviance of 10% from experimental overall heat transfer to tubular reactors skin.*

**keywords:** *heat transfer, pyrolysis furnace, thermal radiation, zone method, erhesics*

### 1. Introduction

This work is a result of the development of a computational code (Janny and Sousa, 2005) to predict the effect of operational parameters of natural gas-fired wall burners installed in Braskem's cracking unit F-1401-A. Main parameters changing during furnace's operation would be air excess (i.e. air coefficient) and natural gas flowrate on each burner level – see Fig. (1).

Both effects of total power source provided through all burners and distribution of this source over these burners should be simulated: for a given total power source, which distribution of this source over all burner levels would lead to a increase on load's (tubular reactor skin) overall heat flux, hence greater efficiency. Numerical results would be then used as input to a process modeling software like Aspen Plus<sup>®</sup>.

The mathematical modeling argued here deals only with furnace region called "radiation chamber" or "radiation section" and, additionally, the following assumptions were made to those of Hottel's work (Hottel and Sarofim, 1967):

- i) Steady state condition;
- ii) Unidirectional flow with allowance for axial mixing, without secondary flow regions (i.e. recirculation);
- iii) Flue gas velocity profile as a sole function of mass flow, temperature and free stream's cross-sectional area;
- iv) Flue gas composition along furnace was calculated from imposed air coefficients (air coefficient  $\lambda_{air} = \frac{[\text{effectively used air}]}{[\text{stoichiometry air}]}$ );
- v) Complete fuel conversion (extent of reaction  $X = 1$ );
- vi) Combustion was admitted as punctual phenomena, as if each burner would be replaced by a punctual source of power related to its "injected" fuel flow rate (power =  $\dot{m}_{ng} \cdot HHV_{\forall}$ , where  $\dot{m}_{ng}$  and  $HHV_{\forall}$  are flow rate and higher heating value of natural gas, respectively, within control volume  $\forall$ ).

## 2. Description of the cracking unit

Radiation section has a central plane (symmetry plane) containing 22 reactor tubes, plus 6 reactor tubes on a cross-over plane located on the end of this section. Furnace can be approximated as a prism with rectangular base, whose dimensions (in meters) are: 6.5 height, 1.6 width and 16.7 length.

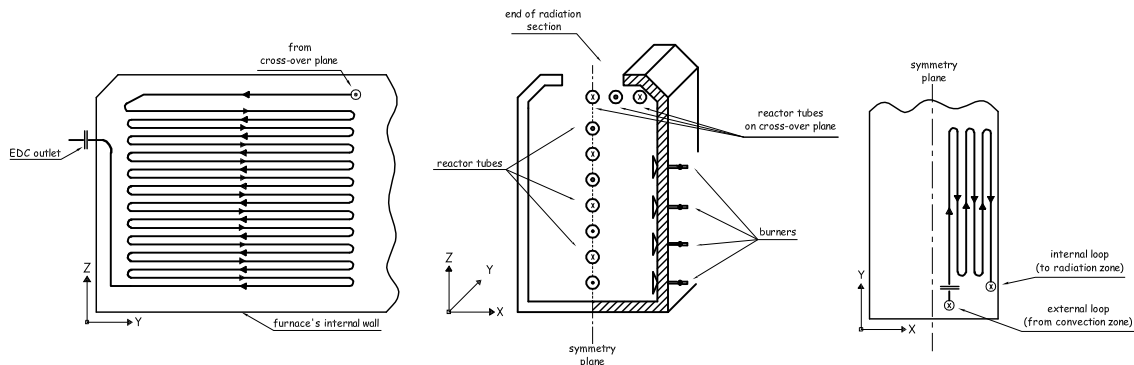


Figure 1: Representation of cracking furnace: load surfaces (reactor tubes) and refractory surfaces (burners)

Heat required for endothermic cracking reaction is provided by means of radiant wall burners located in parallel to the symmetry plane. There're 4 rows (levels) of burners on each side of furnace, each row having 10 burners. These burners provide uniform heat distribution through indirect heating of reactor tubes skin: bulk of radiative heat transfer is due to refractory surfaces (these ones first being heated mainly by burner's flame) emitting and reflecting to reactor tubes. More details about this type of petrochemical furnace may be gathered from John Zink's Handbook (Baukal, 2001).

## 3. Zoning and discretization

Usually, zone size is dictated by characteristic length scale of conduction and convection (Hottel and Sarofim, 1967). For this problem, convective and conductive heat transfer answer for less than 15% of the overall heat flux to load, therefore a rough furnace zoning, as detailed in Fig. (2), already gives a good description of the problem, and is coherent with experimental data about reactor tube skin temperatures. The present zoning scheme is for one half of furnace, as the problem will be admitted plane-symmetric:

- 5 gas volume zones ( $V_0, V_1, V_2, V_3, V_4$ ): main gas species are  $O_2$ ,  $H_2O$ ,  $CO_2$  and  $N_2$ ; in the first four volumes ( $V_0 \dots V_4$ ) there are heat sources (burners); there aren't burners in fifth volume;
- 6 load surface zones ( $S_2, S_6, S_{10}, S_{14}, S_{18}, S_{20}$ ): discontinued load surfaces composed by spaced tubes and voids; there're strong exchanges by radiation ( $S_2, S_6, S_{10}, S_{14}, S_{18}$ ) and convection ( $S_{20}$ );
- 16 refractory surface zones (remaining surfaces): made of refractory composites; on some of these surfaces are installed wall burners ( $S_0, S_4, S_8, S_{12}$ ).

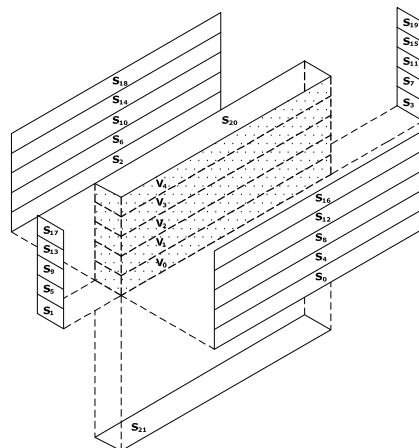


Figure 2: Isometric detail of furnace zoning and identification of all domain zones (surfaces and volumes)

Figure 3 gives a detailed view of unidirectional mass fluxes along volume zones.

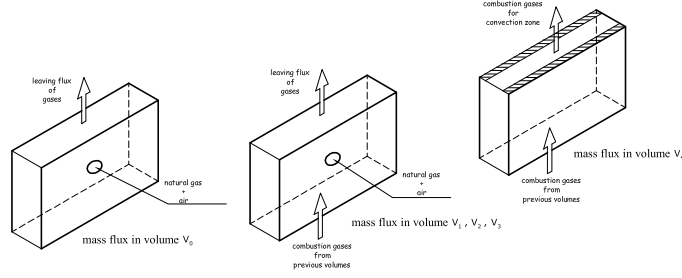


Figure 3: Simplified model for mass flux in each volume zone

#### 4. Zonal analysis

To calculate radiative heat transfer, an integral approach was used employing zonal analysis (Hottel and Sarofim, 1967). Mathematically, the zone method is, perhaps, the simplest numerical quadrature of the differential-integral equation (Radiative Transfer Equation or RTE) governing radiative transfer. Hottel's method make no approximation or assumption to simplify the RTE. By using the concept of radiosity and the principle of energy conservation, this method transforms RTE into a new set of nonlinear algebraic equations.

Together with conventional finite difference techniques to represent convective transports, the zone method provides a way to discretize radiation contribution on a general differential-integral equation.

#### 5. Total balances for zones forming a cavity

Terms related to radiative transport are written as function of exchange factors known as directed flux areas, denoted by  $\overrightarrow{GG}$ ,  $\overrightarrow{GS}$ ,  $\overleftarrow{GS}$  and  $\overleftarrow{SS}$  for gas-gas, gas-surface, surface-gas and surface-surface exchanges, respectively.

Using the theory developed by Hottel and Sarofim, directed flux areas can be calculated from more fundamental exchange factors, as shown in Tab. (1):

Table 1: Exchange areas used in zone method for radiation analysis (Rhine and Tucker, 1991)

Exchange Factor	Denomination	Function of
direct exchange area	$\overline{g_i g_j}, \overline{g_i s_j}, \overline{s_i s_j}$	geometry, gray gas extinction coefficient $K$ .
total exchange area	$\overline{G_i G_j}, \overline{G_i S_j}, \overline{S_i S_j}$	geometry, $K$ , surface emissivity $\epsilon_i$
directed flux area	$\overrightarrow{G_j G_i}, \overleftarrow{G_j G_i}, \overrightarrow{G_j S_i}, \overleftarrow{G_j S_i}, \overrightarrow{S_j S_i}, \overleftarrow{S_j S_i}$	geometry, $K$ , $\epsilon_i$ , $T_{s,i}$ and $T_{g,i}$ temperatures.

Directed flux areas are computed from total exchange areas, which in turn are computed from direct exchange areas. Using these exchange factors, the transfer of thermal radiation for a particular zone may be incorporated in a total energy balance. In this balance all others heat transfer modes are accounted.

Equations are written for all gas and surface zones, giving rise to a simultaneous system of nonlinear equations. This system can be solved to find temperature and overall heat transferred to each zone.

##### 5.1. Total energy balance for a surface $S_i$ with area $A_i$

$$\underbrace{\sum_j \overrightarrow{S_j S_i} E_{S_j}}_{\text{Incident power over } A_i \text{ resulting of the presence of all emitting surfaces.}} + \underbrace{\sum_k \overrightarrow{G_k S_i} E_{g_k}}_{\text{Incident power over } A_i \text{ resulting of the presence of all emitting volumes.}} - \underbrace{A_i \epsilon_i E_{S_i}}_{\text{Emitted power by } A_i.} + \underbrace{\sum_l h_l A_i (T_{g_l} - T_{s_i})}_{\text{Incoming power to } A_i \text{ due to convection.}} = \underbrace{\dot{Q}_{all(i)}}_{\text{Overall power transferred to } A_i.} \quad (1)$$

Where  $h_l$  is a bulk convective heat transfer coefficient and  $E_{z_j}$  is the blackbody emissive power at zone's ( $z_j$ ) temperature, and it's given by Stefan-Boltzmann law ( $E_{z_j} = \sigma T_{z_j}^4$ ).

For surfaces  $S_2, S_6, S_{10}, S_{14}$  and  $S_{20}$ ,  $\dot{Q}_{all(i)}$  stands for the overall heat transferred to each  $i = 2, 6, 10, 12, 14, 20$  load surface (tubular reactor skin). This is the meaningful quantity to evaluate furnace's efficiency. For refractory surfaces,  $\dot{Q}_{all(i)}$  stands for losses by conduction through wall.

### 5.2. Total energy balance for a gas $G_i$ with volume $V_i$

$$\underbrace{\sum_k \vec{G}_k \vec{G}_i E_{g_k}}_{\text{Incident power over } V_i \text{ resulting of the presence of all emitting volumes.}} + \underbrace{\sum_j \overleftarrow{G}_i \overleftarrow{S}_j E_{s_j}}_{\text{Incident power over } V_i \text{ resulting of the presence of all emitting surfaces.}} - \underbrace{4 \sum_{\eta} a_{g,\eta} K_{\eta} V_i E_{g_i}}_{\text{Emitted power by } V_i.} - \underbrace{\sum_l h_l A_l (T_{g_i} - T_{s_l})}_{\text{Leaving power from } V_i \text{ due to convection.}} - \underbrace{\dot{Q}_{e,i}}_{\text{Enthalpy variation associated with gases flux within } V_i.} + \underbrace{\dot{q}_{V,i}}_{\text{Power source released from combustion reaction in volume } V_i.} = 0 \quad (2)$$

Where

$$\dot{Q}_{e,i} = \overleftarrow{\Delta} \mathbf{e} = \sum_{outlet} (\dot{m}_{outlet} \mathbf{e}_{outlet})_i - \sum_{inlet} (\dot{m}_{inlet} \mathbf{e}_{inlet})_i \quad (3)$$

$$\dot{q}_{V,i} = (\dot{m}_{gn})_i HHV_V \quad (4)$$

In volume  $V_i$ , outlet and inlet flowrates (kg/s) for  $O_2, N_2, CO_2, H_2O$  are expressed by  $\dot{m}_{outlet}$  and  $\dot{m}_{inlet}$ , respectively. The terms  $\mathbf{e}_{outlet}$  and  $\mathbf{e}_{inlet}$  – specific enthalpies (kJ/kg) – are calculated at temperatures of volumes  $V_i$  and  $V_{i-1}$ , respectively. Combustion air is referenced at temperature  $T_{ref} = 298K$ . Terms  $a_{g,\eta}$  and  $K_{\eta}$  are weight and absorption coefficients, respectively, of the  $1, 2, \dots, \eta$  gray gas.

### 5.3. Equations system and convergence loop

Generic form of nonlinear equations, to be solved for unknown temperature and heat flux, may be written as bellow.

For total energy balance of  $m$  surfaces:

$$\begin{pmatrix} \alpha_{s_{11}} & \alpha_{s_{12}} & \cdots & \alpha_{s_{1m}} \\ \alpha_{s_{21}} & \alpha_{s_{22}} & \cdots & \alpha_{s_{2m}} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{s_{m1}} & \alpha_{s_{m2}} & \cdots & \alpha_{s_{mm}} \end{pmatrix} \begin{pmatrix} T_{S_1}^4 \\ T_{S_2}^4 \\ \vdots \\ T_{S_m}^4 \end{pmatrix} + \begin{pmatrix} \beta_{s_{11}} & \beta_{s_{12}} & \cdots & \beta_{s_{1m}} \\ \beta_{s_{21}} & \beta_{s_{22}} & \cdots & \beta_{s_{2m}} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{s_{m1}} & \beta_{s_{m2}} & \cdots & \beta_{s_{mm}} \end{pmatrix} \begin{pmatrix} T_{S_1} \\ T_{S_2} \\ \vdots \\ T_{S_m} \end{pmatrix} - \begin{pmatrix} \dot{Q}_{all(1)} \\ \dot{Q}_{all(2)} \\ \vdots \\ \dot{Q}_{all(m)} \end{pmatrix} + \begin{pmatrix} \gamma_{s_1} \\ \gamma_{s_2} \\ \vdots \\ \gamma_{s_m} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (5)$$

$$\alpha_{s_{ij}} = \begin{cases} \sigma \left( \overleftarrow{S}_i \overleftarrow{S}_i - A_i \epsilon_i \right), & \text{if } i = j \\ \sigma \overleftarrow{S}_i \overleftarrow{S}_j, & \text{if } i \neq j \end{cases} \quad \beta_{s_{ij}} = \begin{cases} -\sum_k h_k A_i, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}$$

$$\gamma_{s_i} = \sum_l \overleftarrow{G}_l \overleftarrow{S}_i T_{g,l}^4 + \sum_k A_i T_{g,k} h_k$$

For total energy balance of  $n$  volumes

$$\begin{pmatrix} \alpha_{g11} & \alpha_{g12} & \cdots & \alpha_{g1n} \\ \alpha_{g21} & \alpha_{g22} & \cdots & \alpha_{g2n} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{gn1} & \alpha_{gn2} & \cdots & \alpha_{gnn} \end{pmatrix} \begin{pmatrix} T_{g1}^4 \\ T_{g2}^4 \\ \vdots \\ T_{gn}^4 \end{pmatrix} + \begin{pmatrix} \beta_{g11} & \beta_{g12} & \cdots & \beta_{g1n} \\ \beta_{g21} & \beta_{g22} & \cdots & \beta_{g2n} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{gn1} & \beta_{gn2} & \cdots & \beta_{gnn} \end{pmatrix} \begin{pmatrix} T_{g1} \\ T_{g2} \\ \vdots \\ T_{gn} \end{pmatrix} + \begin{pmatrix} \gamma_{g1} \\ \gamma_{g2} \\ \vdots \\ \gamma_{gn} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad (6)$$

$$\alpha_{g_{ij}} = \begin{cases} \sigma \left( \overleftarrow{G}_i \overleftarrow{G}_i - 4 \sum_{\eta} a_{g,\eta} K_{\eta} V_i \right), & \text{if } i = j \\ \sigma \overleftarrow{G}_i \overleftarrow{G}_j, & \text{if } i \neq j \end{cases} \quad \beta_{g_{ij}} = \begin{cases} - \sum_k h_k A_k, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}$$

$$\gamma_{g_i} = \sum_l \overleftarrow{G}_i S_l T_{s,l}^4 + \sum_k A_k T_{s,k} h_k - \dot{Q}_{e,i} + \dot{q}_{v,i}$$

Systems Eq. (5) and Eq. (6), *a priori*, have  $2m + n$  unknowns and only  $m + n$  equations. Which means one needs to prescribe  $m$  boundary conditions to determinate solutions for those systems. About boundary conditions, see **7. Boundary conditions**.

Briefly, Equation (5) and Eq. (6) were solved by means of a modified Newton-Raphson routine with tuned Powell's functions (Powell, 1970), using trust regions and normalized residues within zero curves of possible solutions. This procedure allowed the nonlinear system to be, relatively, "well behaved" for a given arbitrary initial guess for temperature field.

Convergence loop for computing solution of Eq. (5) and Eq. (6) is shown in Fig. (4).

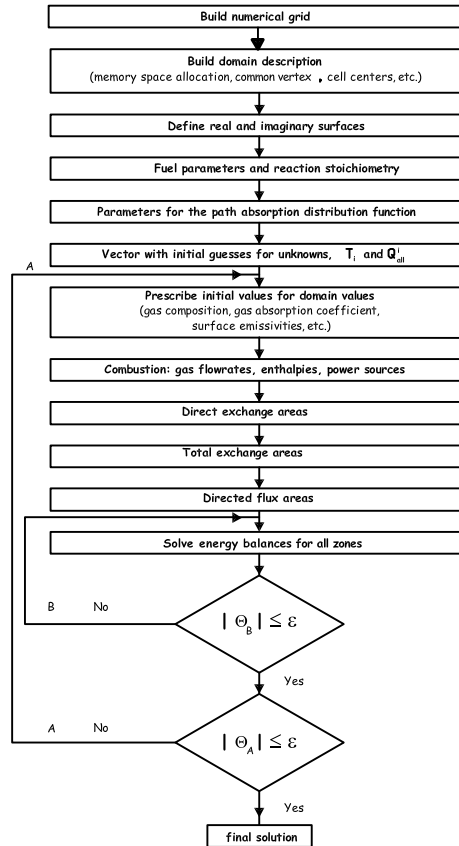


Figure 4: Solution loop of zone method calculus for radiative heat transfer

Inner loop (B) represents the convergence level of an energy balance equation for each zone (for example, equation Eq. (1) for a zone  $i$ ). Outer loop (A) involves a global convergence, not just including a particular balance, but energy balances for all zones (surfaces and volumes), effects of participant medium absorptivities, direct exchange areas, total exchange areas and directed flux areas.

In Fig. (4), convergence criteria are expressed by  $\Theta_B$  and  $\Theta_A$  (here called convergence error) for inner and outer loops, respectively. Some characteristics of convergence stability cannot be inferred only based on the assumption that both source term and radiative intensity are linear with respect to the blackbody intensity of the medium and walls. When the spectral banded nature of absorption and emission phenomena are taken into account (no grey gas assumption), such linearity is not true because of emissivity's dependence of medium and background temperatures.

In this work we propose a expression to monitor convergence based on eigenvalues stability analysis. For example, to measure convergence advance for gas volumes on an outer loop level at  $k + 1$ -th iteration, Equation (7) gives:

$$\Theta_A^{(k+1)} = \chi^{(k+1)} \left( \frac{\chi^{(k+1)}}{\chi^{(k)}} - 1 \right)^{-1} \quad (7)$$

Where  $\chi^{(k+1)} = \sqrt{\sum_{i=0}^n (T_{g,i}^{(k+1)} - T_{g,i}^{(k)})^2}$  is a root mean square (RMS) residual. Figure (5) shows a example for convergence-path measured by  $\Theta_A$ .

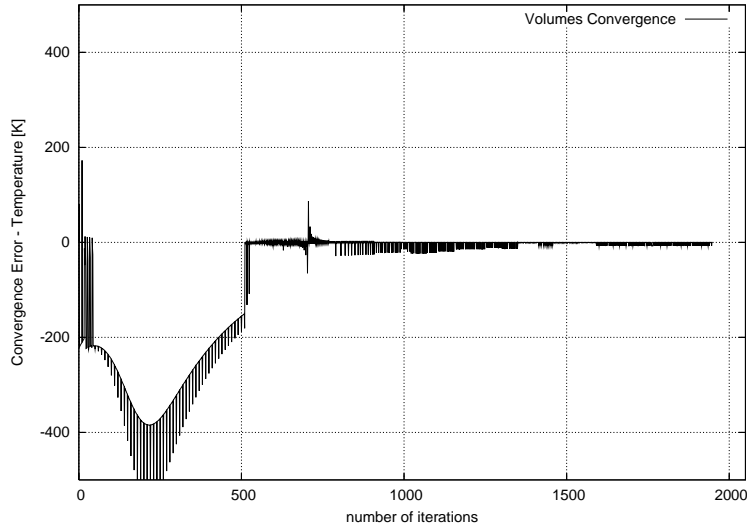


Figure 5: Volume's temperature convergence is measured by Eq. (7)

A more conventional convergence criteria is applied for  $\Theta_B$ .

$$\Theta_B^{(k+1)} = \sqrt{\delta T_{g,i}^{(k+1)} \delta T_{g,i}^{(k)}} \quad (8)$$

Where  $\delta T_{g,i}^{(k+1)} = |T_{g,i}^{(k+1)} - T_{g,i}^{(k)}|$ . Total energy imbalance must be also bellow a target criteria, such as a direct residual  $\delta = \sum_{i=0}^n \dot{q}_{V,i} - \left( \dot{Q}_{e, \text{exit}} + \sum_{j=\text{loads}} \dot{Q}_{all(j)} + \text{wall losses} \right)$ , where  $\dot{Q}_{e, \text{exit}}$  is the enthalpy of flue gases at furnace's exit. This way, one could verify if the problem has, in fact, converged.

#### 5.4. Numerical relaxation

Relaxation factors were used in both outer and inner loops, but outer loop was more affected by stability issues. Figure 5 (outer loop's convergence-path measured in terms of  $\Theta_A$ ) shows a progressive damping of  $\Theta_A$  amplitude, resulting from an adaptive factor relaxation (AFR). Solver allows software's user to change relaxation factor, if needed, after a certain number of iterations or to let itself apply AFR. With the first choice,

user could check, for example,  $\Theta_A$  and  $\delta T_{g,i}$  values and choose suitable relaxation factors to each unknown, dumping iteration instabilities.

One detected source of such instabilities originates from scale differences between terms on Eq. (1) and Eq. (2). In other words, there are terms which are small when compared with some other term with higher magnitude order. For a volume zone total energy balance, specifically, when a great air excess ( $\lambda_{ar} > 1.3$ ) is simulated, the enthalpy term ( $\dot{Q}_{e,i}$ ) becomes, relatively, very big (mainly because of water vapor specific enthalpy). This generates convergence problems in both loops ( $B$  and  $A$ ): for inner loop because of scale differences; and for outer loop because  $\dot{Q}_{e,i}$  is an implicit function of temperature.

When AFR is used, different relaxation schemes are applied in loops  $A$  and  $B$ . For loop  $B$  (Newton-Raphson method), solution is iteratively updated accounting to Eq. (9) and Eq. (10). For a zone  $z_i$  (either surface or volume)

$$T_{z,i}^{(k+1)} = T_{z,i}^{(k)} + \phi^{(k)} \delta T_{z,i}^{(k)} \quad (9)$$

$$\delta T_{z,i}^{(k)} = - \left( J^{(k)} \right)^{-1} F \left( T_{z,i}^{(k)} \right) \quad (10)$$

where  $J$  is the nonlinear system's Jacobian matrix;  $\phi$  is the relaxation factor; and  $F$  is either Eq. (1) or Eq. (2). Factor  $\phi$  is the one (O'Dwyer and O'Donnell, 1995) which minimizes residual  $L_2$  norm in the direction along the solution update, at each iteration (loop  $B$ ). That is,  $\phi^{(k)}$  is chosen as the value which minimizes  $\left\| F \left( T_{z,i}^{(k)} + \phi^{(k)} \delta T_{z,i}^{(k)} \right) \right\|$  or ensures that  $\left\| F \left( T_{z,i}^{(k)} + \phi^{(k)} \delta T_{z,i}^{(k)} \right) \right\| < \left\| F \left( T_{z,i}^{(k)} \right) \right\|$ . For loop  $A$ , Equation (11) was empirically chosen as the relaxation scheme to give best results (great dumping rate and minimum number of iterations needed). Similar method was also proposed by Bastos (Bastos *et al.*, 1995).

$$T_{z,i}^{(k+1)} = (1 - \phi_1) T_{z,i}^{(k)} + \phi_1 \left( T_{z,i}^{(k+1)} + \phi_2^{(k)} \left( T_{z,i}^{(k+1)} - T_{z,i}^{(k)} \right) \sqrt{\frac{T_{z,i}^{(k)}}{T_{z,i}^{(k+1)}}} \right) \quad (11)$$

Equation (11) allows one to choose between two methods: if  $\phi_2$  (which is an adaptive relaxation factor) is set to 0, this equation simplifies to a traditional under-relaxation method ( $0 < \phi_1 \leq 1$ ); if now  $\phi_1 = 1$ , only the term under brackets will be evaluated. Of course, if  $0 < \phi_1 \leq 1$  and  $\phi_2 \neq 0$  a hybrid method will be used (although this case wasn't extensively tested).

The following rules are used to calculate  $\phi_2$ :

- i) For each unknown, set a initial  $\phi_2^{(0)}$  and define an increment  $d\phi$ ;
- ii) For each unknown, calculate  $\Delta^{(k-1)} = T_{z,i}^{(k-1)} - T_{z,i}^{(k-2)}$  and  $\Delta^{(k)} = T_{z,i}^{(k)} - T_{z,i}^{(k-1)}$ ;
- iii) Calculate the product  $\Delta^{(k)} \cdot \Delta^{(k-1)}$ ;
- iv) If the product is negative, oscillation in convergence occurred. In this case set  $T_{z,i}^{(k)} = T_{z,i}^{(k-1)}$  and  $\phi_2 = \phi_{osc}$ ;
- v) If the product is positive, there wasn't oscillation. In this case set  $\phi_2 = \phi_2 + d\phi$ .

where  $\phi_{osc}$  is the value affected if the unknown had oscillated, and normally is smaller than  $\phi_2^{(0)}$ . Unknowns use different relaxation levels determined by their different behavior. Values for  $\phi_2^{(0)}$ ,  $d\phi$  and  $\phi_{osc}$  are chosen depending on problem's individuality. Typical values are (Bastos *et al.*, 1995):  $\phi_2^{(0)} = -0.6$ ,  $d\phi = 0.05$  and  $\phi_{osc} = -0.7$ . To solve Eq. (2), an additional under-relaxation scheme (with  $\phi_2 = 0$ ) was utilized to update just the temperature used to calculate  $\dot{Q}_{e,i}$ , hence minimizing the importance of this term over each iteration.

## 6. Direct exchange areas for rectangular enclosure

Direct exchange area (DEA) represents the geometric and optic relation between each pair of zones (Hottel and Sarofim, 1967). DEA can also be seen as the energy fraction emitted by an element  $z_i$  and absorbed by an element  $z_j$ , after a direct transmission – does not include reflection on others surfaces.

Many authors have been applying different methods to calculate DEA for various geometries: Erkkü (Erkkü, 1959) provided solutions to cylinders and slabs; Vercamen (Vercamen and Froment, 1980) obtained DEAs by calculating photon trajectories using Monte Carlo method.

Works by Erkkü (Erkkü, 1959), Tucker (Rhine and Tucker, 1991) and Arima (Arima, 1998) related to formulation and calculus of DEA admitted the simplification of a uniform zoning and a fixed range of optical



path lengths (given by  $p_g L_m$  product, where  $p_g$  is the sum of partial pressures of absorbing/emitting gas species and  $L_m$  is the mean beam length). In the works by Tucker and Arima, enclosure was approximated by a collection of cubic gas zones bounded by square zone surfaces.

When emissivity and absorptivity of combustion products are represented through use of multiple gray gases, this uniform zoning limits the usage of DEA tables and zone description (size) when high extinction coefficients ( $K > 10 \text{ m}^{-1}$ ) are present. In the simulation of cracking furnace F-1401-A, DEAs were formulated for non uniform zones, using special calculus techniques.

Numerical integration of DEA equations were carried out by Monte Carlo statistic method, with an adaptive algorithm (Janny and Sousa, 2005) which evaluates the integrand on a hypercubic region along random directions. The integrand may be seen as radiation beams emitted by a zone which directly arrive on another zone, accounting their interaction with intervenient medium (gases).

### 6.1. DEA equations

There are three types of DEA: surface-surface ( $s_i s_j$ ), gas-surface ( $g_i s_j$ ) and gas-gas ( $g_i g_j$ ), which can be expressed by the following equations:

$$\overline{s_i s_j} = \int_{A_i} \int_{A_j} \xi(A_i, \vec{r}_{ij}) \tau(R) \frac{\cos(\theta_i) \cos(\theta_j)}{\pi R^2} dA_j dA_i \quad (12)$$

$$\overline{g_i s_j} = \int_{V_i} \int_{A_j} \xi(V_i, \vec{r}_{ij}) \tau(R) \frac{K_i \cos(\theta_j)}{\pi R^2} dA_j dV_i \quad (13)$$

$$\overline{g_i g_j} = \int_{V_i} \int_{V_j} \xi(V_i, \vec{r}_{ij}) \tau(R) \frac{K_i K_j}{\pi R^2} dV_j dV_i \quad (14)$$

where  $\tau(R) = \exp\left(-\int_0^R K(r) dr\right)^1$  is the transmittance;  $R = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$  is the absorption path (see Fig. (6)); and  $\xi(A_i, r_{ij})$  is the characteristic function which accounts for the presence of shadow zones (Janny and Sousa, 2005). Function  $\xi(A_i, r_{ij}) = 1$  when a beam leaving a point  $(x_i, y_i, z_i)$  at surface zone  $A_i$  (or volume  $V_i$ ), locally traveling along a per unit solid angle direction  $\vec{r}_{ij}$ , is seen when looking from a point  $(x_j, y_j, z_j)$  at the other surface zone  $A_j$ . Otherwise (not seen),  $\xi(A_i, \vec{r}_{ij}) = 0$ .

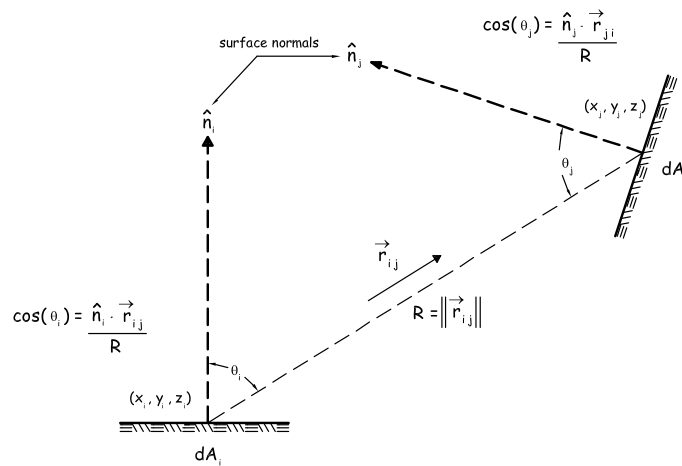


Figure 6: Radiative exchange between two differential surface elements

<sup>1</sup>Instead of fundamental models for transmittance, the exponential law given by Hottel (Hottel and Sarofim, 1967) is used.

Transmittance  $\tau(R)$  quantifies the energy fraction not transmitted by a radiation beam to the volume zones by which this beam goes through, during a trek from its origin point until its final absorption point. As an approximation, it may be written as

$$\tau(R) \approx \int_{R_{min}}^{R_{max}} \exp\left(-\sum_m K_{r_m} \Delta r_m\right) f(R) dR \quad (15)$$

where  $\Delta r_m$  is the fraction of total beam length ( $R$ ) which lies inside a given volume zone with absorption coefficient  $K_{r_m}$ . It's calculated based on the intersection of the radiation beam with bounding planes of the volume zone. The factor  $f(R)$  is a differential distribution of beam lengths, i.e. the probability that an arbitrarily chosen value of beam length lies between  $R$  and  $R + dR$  (Vercamen and Froment, 1980). A statistic fit is used because there exist an infinite number of possible absorption paths between a pair of zones.

By example, direct exchange area between a refractory surface and a load surface, without any shadow zones, may be written as (see Fig. (7)):

$$\overline{s_i s_j} = \int_{\bar{x}_i}^{\bar{x}_i + \Delta x_i} \int_{\bar{z}_i}^{\bar{z}_i + \Delta z_i} \int_{\bar{x}_j}^{\bar{x}_j + \Delta x_j} \int_{\bar{z}_j}^{\bar{z}_j + \Delta z_j} \tau(R) \frac{\bar{y}_j^2}{\pi R^4} dz_j dx_j dz_i dx_i \quad (16)$$

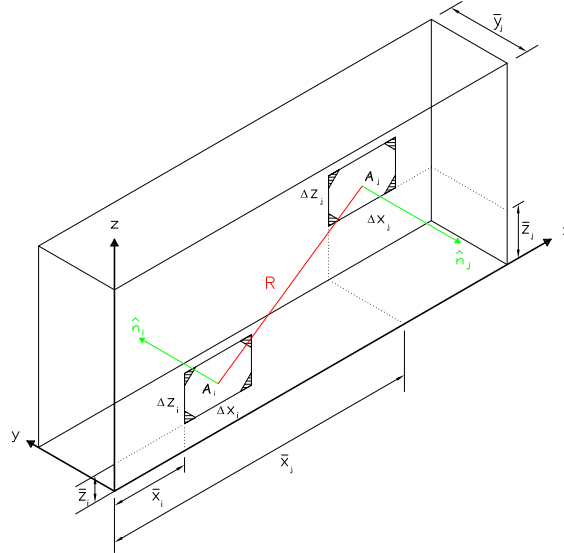


Figure 7: refractory wall zone ( $A_i$ ) and reactor tube zone ( $A_j$ )

## 6.2. Reduction of the dimension of a multiple integral

A special technique to transform double integral into a sum of simple integrals were developed by Erkkü (Erkkü, 1959). This technique can be extended to non uniform grids using an additional argument, as proposed by Tian (Tian and Chiu, 2003).

This procedure leads, altogether with the reduction of integral dimension, to a minimization of singularities of function  $\overline{s_i s_j}$  within its integration domain. This is a result of great importance, since Eq. (12), Eq. (13) and Eq. (14) have integrands with abrupt variation (singularity) within integration domain for some pair of zones (self radiating or adjacent zones).

## 7. Boundary conditions

As already mentioned, to solve systems Eq. (1) and Eq. (2) it's necessary to specify  $m$  boundary conditions. In the case of the actual zoning of furnace F-1401-A, it'll be prescribed 22 conditions. From these, 6 will be Dirichlet boundary conditions – of first kind, which specifies a value  $T = b(x, y, z)$  – prescribed to load surfaces ( $S_2, S_6, S_{10}, S_{14}, S_{18}, S_{20}$ ). Experimental data from Braskem's DCS (Distributed Control System) provided

information about the temperature of fluid process (inside reactor tube), measured by means of thermocouples allocated immediately upstream ( $T_{in}$ ) and downstream ( $T_{out}$ ) from equivalent locations of load surface zones adopted in this work. Starting from arithmetic mean of these values ( $(T_{in} + T_{out})/2$ ), in Tab. (2) tubular reactor skin temperature were calculated accounting for gradients within boundary layer, hence summing a  $\Delta T$  value to this arithmetic mean.

Table 2: Known (experimental) temperatures of load surfaces

Zone	Temperature [K]	Zone	Temperature [K]
$S_2$	$T_{S2} = 802$	$S_{14}$	$T_{S14} = 655$
$S_6$	$T_{S6} = 767$	$S_{18}$	$T_{S18} = 579$
$S_{10}$	$T_{S10} = 732$	$S_{20}$	$T_{S20} = 542$

The 16 remaining information will be Neumann boundary conditions – of second kind, which specifies a value for  $\partial T/\partial \vec{n} = \hat{n} \cdot \nabla T$  – prescribed to refractory surfaces ( $S_0, S_1, S_3, S_4, S_5, S_7, S_8, S_9, S_{11}, S_{12}, S_{13}, S_{15}, S_{16}, S_{17}, S_{19}, S_{21}$ ). In Eq. (2) and Eq. (3), this condition will be associated with term  $\dot{Q}_{all}$ .

If furnace walls are not admitted adiabatic ( $\dot{Q}_{all} = 0$ ), there will be losses through these walls and  $\dot{Q}_{all}$  wont be null. In this case,  $\dot{Q}_{all}$  will be iteratively computed in the following way (Janny and Sousa, 2005):

$$\dot{Q}_{all(i)}^{(k)} = \text{Losses}_i^{(k)} = \dot{Q}_{conduction(i)}^{(k)} = A_i \frac{\kappa_i^{(k)}}{\Delta l_i} \left( T_{si}^{(k)} - T_{i,e}^{(k)} \right) \quad (17)$$

$$\left[ T_{i,e}^{(k)} \right]^4 + \left( \frac{\Delta l_i h_{i,e}^{(k-1)} + \kappa_i^{(k-1)}}{\Delta l_i \epsilon_{i,e}^{(k-1)} \sigma} \right) T_{i,e}^{(k)} - \frac{\kappa_i^{(k-1)} T_{si}^{(k-1)} + \Delta l_i h_{i,e}^{(k-1)} T_{\infty,e}}{\Delta l_i \epsilon_{i,e}^{(k-1)} \sigma} = 0 \quad (18)$$

where  $T_{i,e}$  is furnace external wall's temperature (adjacent to surface  $S_i$ );  $\Delta l_i$  is refractory block thickness (adjacent to surface  $S_i$ );  $\kappa_i$  is refractory thermal conductivity (adjacent to surface  $S_i$ );  $\epsilon_{i,e}$  is furnace external wall's emissivity (adjacent to surface  $S_i$ );  $h_{i,e}$  is furnace external wall's convection coefficient (adjacent to surface  $S_i$ ); and  $T_{\infty,e}$  is the environment temperature. The quartic Eq. (18) has a known analytic solution.

### 7.1. Model to represent an array of tubes

A model was developed (Janny and Sousa, 2005) to replace a set of tubes and voids, by a gray plane (in the same averaged temperature of the array of tubes within the discredited zone), whose emissivity is calculated as follows (see Fig. (8)):

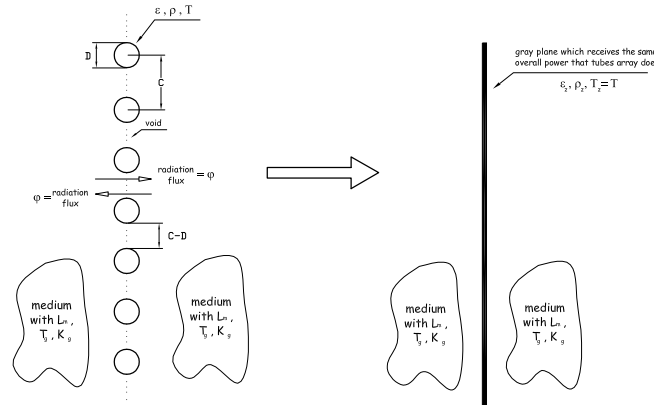


Figure 8: Replacing a set of tubes and voids by a gray plane

$$\epsilon_z = \frac{\epsilon}{\rho} \frac{\pi D}{2C} \left[ \frac{D}{2} Y^2 + (C - D)(Y - \pi) \right] \left[ \frac{D}{2} Y^2 + (C - D) \left( Y - \frac{\pi}{\rho} \right) \right]^{-1} \quad (19)$$

where  $Y = \arcsin\left(\frac{D}{C}\right) + \left[\left(\frac{C}{D}\right)^2 - 1\right]^{1/2} - \frac{C}{D}$ . Assuming symmetric conditions on both sides of furnace, equal amount of radiation which comes from the right side and irradiates over a void with  $C - D$  width, also leaves the left side towards right side. Therefore, it's like if the void  $C - D$  would reflect all incident radiation. In other words, the void  $C - D$  behaves like a reflector ( $\epsilon = 0, \rho = 1$ ). Similar procedure is detailed by Hottel (Hottel and Sarofim, 1967).

### 8. Results

Simulation results are compared with measured experimental data presented in Tab. (3). Operating conditions of furnace are given in Tab. (4).

Table 3: Experimental data from furnace F-1401-A operation

Experimental data	Value
Overall heat transferred to reactor tube skin (both sides of furnace)	9206 [kW]
Overall heat transferred to reactor tube skin (one side)	4603 [kW]
Volume $V_0$ temperature	1144 [K]
Volume $V_2$ temperature	1184 [K]
Volume $V_3$ temperature	1189 [K]
Flue gas temperature at radiation section's exit ( $V_4$ )	1062 [K]

Table 4: Furnace F-1401-A operating conditions

Condition	Value (half furnace)	Condition	Value
Fuel flowrate in zone $S_0$	0.0481 [kg/s]	Fuel's carbon content	0.7245
Fuel flowrate in zone $S_4$	0.0349 [kg/s]	Fuel's hydrogen content	0.2323
Fuel flowrate in zone $S_8$	0.04134 [kg/s]	Fuel's sulfur content	0.000
Fuel flowrate in zone $S_{12}$	0.04134 [kg/s]	Fuel's nitrogen content	0.0297
Fuel flowrate in zone $S_{16}$	0.0 [kg/s]	Fuel's oxygen content	0.0135
Air excess	1.05 [air/stoich air]	Fuel's higher heating value	49113 [kJ/kg]
Heat input (total)	8137 [kW]	Fuel's density	0.7729 [kg/Nm <sup>3</sup> ]

Figure (9) gives the simulation results for predicted temperatures and Fig. (10) gives the predicted overall heat transferred to each load (tubular reactor skin) zone, summing a total of 4375 kW. Hence, a deviation of  $-4.9\%$  from the value in Tab. (3).

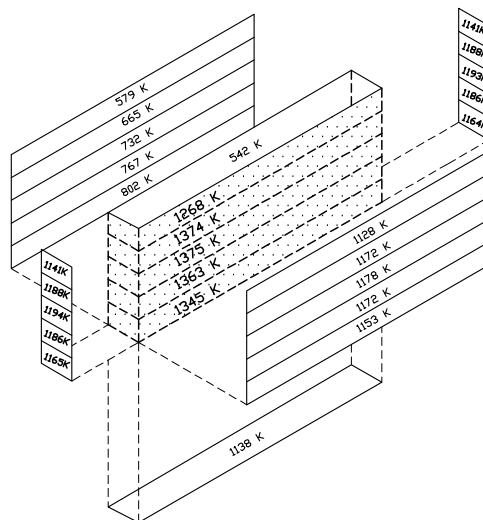


Figure 9: Temperature results from simulation

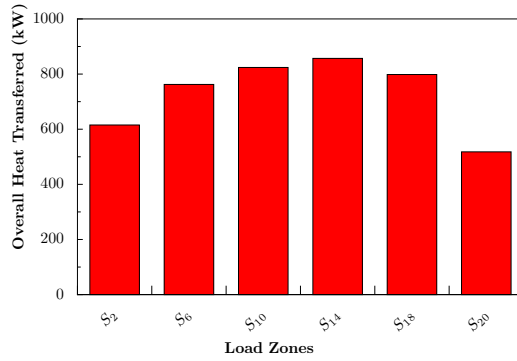


Figure 10: Overall heat transferred to each load surface (simulation result)

The deviation from the experimental overall heat transfer – see Table (5) – is acceptable in terms of mathematical modeling, given the actual simplifications about zoning and flow descriptions.

Table 5: Relative deviations between simulated and experimental temperatures

Volume zone	Experimental value [K]	Simulated value [K]	Deviation [%]
V <sub>0</sub>	1144	1345	-17.6
V <sub>2</sub>	1184	1375	-16.1
V <sub>3</sub>	1189	1374	-15.6
V <sub>4</sub>	1062	1268	-19.4

These significant differences between simulated and experimental values are due to many factors related to both mathematical model and furnace operating conditions when measurements were taken:

- a) During fieldwork, combustion conditions in all volumes were sub-stoichiometric (*CO* content in flue gases well over 5000 ppm). In simulated cases, combustion was admitted super-stoichiometric in all volumes ( $\lambda_{ar} = 1.05$ ). This condition results in an overestimation of simulated  $T_{g,i}$  ( $i = 0, 2, 3, 4$ );
- b) Experimental values for volume  $V_4$  temperature were continually measured by a bare thermocouple, inserted into the gas stream immediately before radiation section's exit. Hence allowing thermocouple to "view" much cooler surfaces than gases of the stream where it was inserted. This condition results in a sub-estimation of the experimental value for  $T_{g,4}$ .

## 9. References

Arima, M. N., 1998, "Aplicação do Método de Zonas a um Forno de Refino de Cobre", Graduation Project, Escola Politécnica da Universidade de São Paulo, São Paulo.

Bastos, J. A., Ida, N., and Mesquita, R. C., 1995, "A Variable Relaxation Technique in Non-linear Problems", IEEE Transactions of Magnetics, Vol. 31, pp. 1733–1736.

Baukal, C. E., 2001, "The John Zink Combustion Handbook", CRC Press, New York, United States.

Erkku, H., 1959, "Radiant Heat Exchange in Gas-filled Slabs and Cylinders", PhD thesis, Massachusetts Institute of Technology, Cambridge.

Hottel, H. C. and Sarofim, A. F., 1967, "Radiative Transfer", McGraw-Hill, New York, United States.

Janny, R. and Sousa, F. D. A., 2005, "Simulação Tridimensional do Forno F-1401-A" (Technical Report N°80337-205), Instituto de Pesquisas Tecnológicas de São Paulo - IPT.

O'Dwyer, J. and O'Donnell, T., 1995, "Choosing the Relaxation Parameter for the Solution of Nonlinear Magnetic Field Problems by the Newton-Raphson Method", University of College Dublin.

Powell, M. J. D., 1970, "A Hybrid Method for Nonlinear Equations", Numerical Methods for Nonlinear Algebraic Equations (Gordon and Breach), pp. 86–114.

Rhine, J. M. and Tucker, R. J., 1991, "Modelling of Gas-Fired Furnaces and Boilers", McGraw-Hill, London, United Kingdom.

Tian, W. and Chiu, W. K. S., 2003, "Calculation of Direct Exchange Areas for Nonuniform Zones Using A Reduced Integration Scheme", ASME Int. J. Heat Transfer, Vol. 125, pp. 839–844.

Vercamen, H. A. J. and Froment, G. F., 1980, "An Improved Zone Method Using Monte-Carlo Techniques for Simulaion of Radiation in Industrial Furnaces", Int. J. Heat Mass Transfer, Vol. 23, pp. 329–337.

# IDENTIFICAÇÃO NARMAX POLINOMIAL DE UM TROCADOR DE CALOR USANDO ALGORITMO DOS MÍNIMOS QUADRADOS ORTOGONAIS

**Leandro dos Santos Coelho, Marcelo Wichhoff Pessoa e Ricardo Massao Suguimoto**

Laboratório de Automação e Sistemas, LAS

Programa de Graduação em Engenharia de Produção e Sistemas, PPGEPS

Pontifícia Universidade Católica do Paraná – PUCPR

Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil

leandro.coelho@pucpr.br; celowp@gmail.com; ricardo.ms@gmail.com

*Resumo. O problema de estimação de parâmetros é de grande relevância em problemas de identificação de sistemas levando em consideração a representação não linear escolhida. Na literatura, este problema tem sido resolvido de diferentes formas dependendo da representação utilizada. Neste trabalho, um procedimento de identificação de sistemas não lineares baseado em uma representação NARMAX (Nonlinear AutoRegressive Moving Average with eXogenous inputs) polinomial e algoritmo dos mínimos quadrados ortogonais é aplicado a um trocador de calor. Os modelos NARMAX polinomiais descrevem sistemas não lineares em termos de equações a diferenças lineares nos parâmetros relacionando a saída atual a combinações (possivelmente não lineares) de entradas e saídas passadas. Isto é apropriado a modelagem de componentes estocásticos e determinísticos de um sistema e é capaz de descrever uma vasta variedade de sistemas não lineares. Neste trabalho, o vetor regressor utilizado para compor o modelo NARMAX polinomial é cuidadosamente escolhido, entre centenas de candidatos, por um procedimento automático. O resultado de simulação obtido foi muito satisfatório na previsão “infinitos-passos-a-frente” da saída do trocador de calor.*

*Palavras chave: trocador de calor, identificação não-linear, modelo NARMAX, algoritmo dos mínimos quadrados ortogonais.*

## 1. Introdução

A utilização de modelos matemáticos é importante para o desenvolvimento da ciência e tecnologia, pois com estes modelos é possível determinar o comportamento dinâmico de um sistema dinâmico, por exemplo, um sistema térmico. As técnicas de modelagem podem ser úteis para a construção e análise de modelos matemáticos que reproduzam alguns comportamentos observados na natureza (Rodrigues, 1996).

A utilização de um modelo para representar um determinado sistema pode estar relacionada com os mais diversos objetivos, tais como: (i) compreender certas dinâmicas do sistema dinâmico (processo) estudado, (ii) prever o comportamento do sistema sob diversas condições de operação, (iii) analisar e projetar controladores, (iv) estimar variáveis do processo que não podem ser medidas diretamente, (v) otimizar o comportamento do sistema, (vi) permitir detecção eficiente de falhas no sistema, (vii) permitir o estudo do sistema em regiões de operação dispendiosas ou problemáticas no sistema real, permitindo um treinamento de operação seguro e eficiente (Matko *et al.*, 1992).

Os modelos matemáticos podem ser compostos por conjuntos de equações diferenciais ou equações a diferenças que representam o comportamento dinâmico do processo a ser modelado. Estes modelos matemáticos não conseguem representar na totalidade o processo real, pois há vários fenômenos que são complexos para serem representados matematicamente.

Os sistemas dinâmicos em uma análise detalhada são não lineares, mas há casos em que aproximações lineares são suficientes para utilização prática. Entretanto, há casos que representações lineares são limitadas e não conseguem reproduzir bem certas dinâmicas desejadas, com isso é necessário o uso de representações não lineares.

As técnicas de modelagem de sistemas são agrupadas em três conjuntos, denominadas: (i) modelagem caixa branca, (ii) modelagem caixa preta e (iii) modelagem caixa cinza. A modelagem caixa branca é utilizada quando há um bom conhecimento do processo a ser modelado, pois nele são utilizados de leis da física para obter as equações do modelo. Quando não há nenhum conhecimento da física do processo, e se tem em mãos apenas dados de entrada e saída, ou saída apenas, se utiliza a abordagem da identificação caixa preta, que é uma alternativa a caixa branca (Aguirre, 2004). Pois nela agrupa-se o ferramental matemático para se obter o modelo aproximado do processo. A terceira abordagem, a identificação caixa cinza se utiliza do conhecimento prévio da física do processo aliado a identificação caixa preta. Nela se utiliza de conhecimentos auxiliares que não se encontram em dados de entrada e saída do sistema sob análise.

A identificação de sistemas também pode ser dividida em identificação linear e não linear. A identificação linear é mais simples em relação a identificação não linear, pois esta não pode ser composta por quaisquer dados (regressores),

o que causaria uma dinâmica não desejada, quando utilizado o modelo não linear, pois os termos redundantes se cancelam através de pólos e zeros, com isso o modelo obtido não geraria uma dinâmica não desejada.

A concepção de novas e eficientes abordagens para identificação de sistemas dinâmicos não lineares tem sido tema freqüente de pesquisas apresentadas na literatura recente. Neste contexto, pode-se citar, por exemplo, a utilização de modelos de Wiener, Hammerstein, bilinear, Volterra, redes neurais artificiais, sistemas nebulosos, *wavelets*, entre outros (Haber e Unbehauen, 1990; Johansson, 1993; Sjöberg *et al.*, 1995; Haykin, 1996; Ljung, 2001).

Dentre as contribuições significativas na área de identificação de sistemas dinâmicos não lineares estão o da utilização de estimação de modelos matemáticos NARMAX (*Nonlinear AutoRegressive Moving Average with eXogenous inputs*) usando o algoritmo dos mínimos quadrados ortogonais (Leontaris e Billings, 1987; Korenberg *et al.*, 1988; Chen *et al.*, 1989; Kukreja *et al.*, 2003; Zito e Landau, 2005).

Este trabalho aborda a identificação de um trocador de calor usando modelo matemático NARMAX e algoritmo dos mínimos quadrados ortogonais (*Orthogonal Least Squares*, OLS). O trocador de calor analisado consiste de um trocador de calor de fluxo de vapor líquido saturado, onde a água aquecida é aquecida por vapor de fluxo saturado pressurizado em um tubo de cobre.

O artigo é organizado da seguinte forma. A descrição do modelo NARMAX e o algoritmo OLS são detalhados na seção 2. Uma breve descrição do trocador de calor e a análise dos resultados obtidos na identificação são detalhadas na seção 3. Finalizando, a conclusão e comentários sobre futura pesquisa são apresentados na seção 4.

## 2. Modelos matemáticos NARMAX

Para sistemas lineares discretos utilizam-se modelos de equações lineares a diferenças que relacionam variáveis de entrada e saída do sistema. Neste contexto, pode-se citar os modelos do tipo ARX (*AutoRegressive with eXogenous inputs*), ARMAX (*AutoRegressive Moving Average with eXogenous inputs*), dentre outros. Uma descrição análoga para sistemas não lineares discretos são os modelos NARMAX que realizam um mapeamento não linear das entradas e saídas passadas para a saída atual (Sjöberg, 1995). Tais modelos são capazes de representar uma ampla classe de sistemas não lineares e ao contrário do que ocorre com as séries de Volterra e Wiener (Billings, 1980), evitam a necessidade de grande número de parâmetros (Chen e Billings, 1989; Cassini, 1999).

Os modelos polinomiais apresentam vantagens em relação a outros tipos de representações pois ele se transforma em uma representação linear quando o ponto de operação do sistema é mantido aproximadamente fixo e também a facilidade sobre a informação analítica sobre o modelo pode ser obtido. Este modelo consegue representar com exatidão, desde que os dados não apresentem variações abruptas.

A utilização de algoritmos de estimação de parâmetros de sistemas lineares, que são fáceis de implementar e que convergem rapidamente, é possível de ser utilizada, pois funções não-lineares polinomiais são lineares nos parâmetros.

O modelo NARMAX pode ser representado pela seguinte forma geral:

$$y(t) = F^l [y(t-1), \dots, y(t-n_y), u(t-\tau_d), \dots, u(t-\tau_d-n_u+1), e(t-1), \dots, e(t-n_e)] + e(t) \quad (1)$$

onde  $\tau_d$  é o atraso de transporte ou tempo morto,  $e(t)$  são os efeitos que não são bem representados por  $F^l[\cdot]$ , que é uma função polinomial não linear de  $y(t)$ ,  $u(t)$  e  $e(t)$ . A função  $F^l[\cdot]$  possui um grau de não-linearidade  $l \in \mathbb{Z}^+$  e o período de amostragem  $T_s$  é considerado normalizado, por isso não foi considerado na equação (1). A equação (1) é composta por uma parte estocástica,  $e(t)$ , e uma parte determinística, ou seja, a parte que envolve  $F^l[\cdot]$ . Nota-se que de acordo com o grau de não-linearidade  $l$  e das ordens  $n_y$ ,  $n_u$  e  $n_e$ , o número de termos pode aumentar acentuadamente no modelo polinomial representado pela equação (1).

Em casos lineares há cancelamento de pólos e zeros, caso haja termos redundantes, já em sistemas não lineares (neste trabalho em questão) quando ocorre uma sobre-parametrização tende a gerar regimes dinâmicos não apresentados pelo sistema real.

Em muitos casos não é necessário utilizar de uma excessiva quantidade de termos para representar bem um sistema, basta apenas uma quantidade pequena para isso, e isto é feito por algoritmos de *detecção de estrutura*. Neste trabalho será utilizado o algoritmo de taxa de redução de erro (*Error Ratio Reduction*, *ERR*) que é muito utilizado na detecção de estrutura dos modelos NARMAX polinomiais. O *ERR* indica o quanto cada termo inserido no modelo pode melhorar a representação do sistema. Normalmente o *ERR* é utilizado junto com métodos de ortogonalização.

### 2.1. Estimação dos parâmetros do modelo NARMAX

O projetista após escolher o modelo matemático NARMAX para representar um sistema dinâmico, deve-se estimar os parâmetros, para que o modelo identificado consiga gerar uma dinâmica semelhante ao do sistema. Normalmente se utiliza algoritmos baseados em mínimos quadrados aos dados de entrada e saída. Neste caso, o modelo NARMAX pode ser apresentado da seguinte forma:

$$y(t) = \sum_{i=1}^{n\theta} p_i(t)\theta_i + e(t) \quad (2)$$

onde  $p_i(t)$  representam os regressores e os  $\theta_i$  são os respectivos parâmetros.

Aplicando a técnica de mínimos quadrados (*Least Squares*, LS) a função custo, chega-se a equação (3),

$$J_N(\theta) = \frac{1}{N} \sum_{i=1}^N \xi(t, \theta)^T \xi(t, \theta) \quad (3)$$

onde  $\xi(t, \theta)$  é o sinal de erro, ou seja,  $\xi(t, \theta) = y(t) - \hat{y}(t, \theta)$ , onde  $y(t)$  é a saída real do sistema e  $\hat{y}(t, \theta)$  é a saída estimada. Para conseguir determinar os parâmetros  $\theta$ , basta minimizar (4) e a solução é dada por:

$$\hat{\Theta}_{LS} = (P^T P)^{-1} P^T Y \quad (4)$$

As matrizes  $Y$  e  $P$  são as matrizes já definidas *a priori*, a matriz  $(P^T P)$  é a conhecida como matriz de informação, ela é simétrica e positiva definida quando  $P$  tem posto pleno de colunas (colunas são linearmente independentes e formam uma base num espaço vetorial gerado por  $P\Theta$ ). Mas em casos não lineares, pode haver um mal condicionamento numérico da matriz  $P$  e ela pode não ter estas propriedades.

Na maioria dos casos,  $Y$  não pertence ao espaço vetorial gerado pelas colunas de  $P$ , com isso os dados do sistema não são completamente explicados pelos regressores. Nesse caso,  $P\hat{\Theta}_{LS}$ , onde  $\hat{\Theta}_{LS}$  é o vetor de parâmetros calculado pelo método dos mínimos quadrados (*Least Squares*, LS), representará a projeção ortogonal de  $Y$  no espaço de imagem de  $P$  e os resíduos de estimação  $\xi(t)$  serão mínimos e ortogonais a tal espaço.

## 2.2. Detecção da estrutura usando ERR

Analisando o resíduo do modelo identificado, e considerando a esperança matemática  $E[\xi(t)] = 0$ , pode-se definir a variância do erro,  $\xi(t)$ , pela equação

$$Var[\xi(t)] = \lim_{N \rightarrow \infty} \frac{1}{N} \left[ Y^T Y - \sum_{i=1}^{n\theta} g_i^2 w_i^T w_i \right] \quad (5)$$

onde  $g_i$  indica os elementos do vetor de parâmetros,  $g$  e  $w_i$  indica os regressores ortogonais.

Se nenhum termo fosse incluído no modelo, a variância de  $\xi(t)$  seria igual ao erro quadrático médio da saída  $y(t)$ . A cada inclusão de termo, a variância de  $\xi(t)$  é decrescida de um fator  $1 / \sum_{i=1}^{n\theta} g_i^2 w_i^T w_i$ , onde  $w_i$  indica o termo incluído e  $g_i$  o seu respectivo parâmetro. Para reduzir a variância deve-se normalizar o erro quadrático médio do sinal de saída. Deste modo, o ERR para cada termo é definido como:

$$[ERR]_i = \frac{g_i^2 w_i^T w_i}{Y^T Y}, \quad 1 \leq i \leq n\theta \quad (6)$$

O ERR indica a porção da variância na saída, quando se insere um novo termo no modelo, ou seja, os termos que possuem maior ERR, constituiriam o modelo final.

Quanto maior o índice ERR, maior é a importância do termo para o modelo. Assim, os termos que apresentarem um índice ERR muito baixo, não são incluídos no modelo. Desta forma, este critério evita o mal condicionamento numérico já que termos linearmente dependentes ou irrelevantes são eliminados (Cassini, 1999).

O índice  $[ERR]_i$  do  $i$ -ésimo termo,  $w_i$ , indica a proporção do valor médio quadrático da saída,  $Y^T Y$ , explicada por este termo. Consequentemente,

$$\sum_{i=1}^{n\theta} ERR = 1, \quad (7)$$

e desta forma, os termos irrelevantes para o modelo, contribuirão com um percentual desprezível para o índice ERR, podendo ser eliminados.



### 2.3. Expansão do modelo polinomial NARMAX

Para a representação polinomial serão consideradas *a priori* as seguintes condições: (i) o sistema possui um atraso de transporte ou tempo morto ( $\tau_d$ ) e (ii) nenhum termo cujo parâmetro tenha que ser estimado pode depender do resíduo no mesmo instante, e é representado pelo termo  $e(t)$ , logo o modelo NARMAX será representado pela equação (1) (Aguirre, 2004).

A equação (1) é composta por uma parte estocástica,  $e(t)$ , e uma parte determinística, ou seja, a parte que envolve  $F^l[\cdot]$ .

A parte determinística pode ser expandida em um somatório de termos com graus de não linearidade variando de  $1 \leq m \leq l$ . Deste modo cada termo de grau  $m$  pode conter um fator de grau  $p$  do tipo  $y(t-i)$  e um fator de grau  $(m-p)$  do tipo  $u(t-i)$  sendo multiplicado por um parâmetro representado por  $C_{p,m-p}(n_1, \dots, n_m)$ , onde obtém-se a equação (7), dada por (Aguirre, 2004):

$$y(t) = \sum_{m=0}^l \sum_{p=0}^m \sum_{n_1, n_m}^{n_y, n_u} C_{p,m-p}(n_1, \dots, n_m) \prod_{i=1}^p y(t-n_i) \prod_{i=p+1}^m u(t-n_i) \quad (7)$$

sendo que,

$$y(t) = \sum_{n_1, n_m}^{n_y, n_u} = \sum_{n_1=1}^{n_y} \dots \sum_{n_m=1}^{n_u} \quad (8)$$

e o  $n_y$  é o limite superior para os somatórios que possuem fatores  $y(t-n_i)$  ou  $n_u$  para os fatores  $u(t-n_i)$ . Neste trabalho considerou-se um modelo NARMAX incluindo-se também os termos lineares ( $n_{eli}$ ) para o sinal de erro atrasado. Neste caso, o modelo NARMAX pode ser representado pela seguinte forma geral:

$$y(t) = F^l[y(t-1), \dots, y(t-n_y), u(t-\tau_d), \dots, u(t-\tau_d-n_u+1), e(t-1), \dots, e(t-n_e)] + [e(t-1), \dots, e(t-n_{eli})] \quad (9)$$

Neste trabalho, o atraso de transporte ou tempo morto,  $\tau_d$ , foi adotado como conhecido *a priori* e com valor unitário.

### 2.4. Estimador dos mínimos quadrados ortogonais

Após determinar a estrutura NARMAX, o próximo passo é determinar os parâmetros do modelo. Os algoritmos usados para estimar os parâmetros são os baseados em OLS, que utilizam o fato de que a matriz  $\varphi^T \varphi$  é simétrica por construção; pois a matriz de regressores ( $\varphi$ ) pode possuir muitas colunas, que seria um caso típico de sistemas não-lineares e com isso não resultaria em boas estimativas. Tais algoritmos são: método clássico de Gram-Schmidt (CGS), método Modificado de Gram-Schmidt (MGS) e método de Golub-Householder (GH) (Aguirre, 2004).

Este algoritmo é baseado no método dos mínimos quadrados clássico, ou seja, uma variante para ser utilizado como estimador de parâmetros em modelos não lineares. Nele aplicando-se a equação (10)  $N$  vezes, ao longo dos dados, chega-se na equação (11), tal que

$$y(t) = \varphi^T(t-1) \hat{\theta} + \xi(t) \quad (10)$$

$$y = \Psi \hat{\theta} + \xi \quad (11)$$

onde  $\Psi$ , é o vetor de medições (regressores) e para tempos negativos são assumidos valores nulos. Os elementos da equação (10) são:

$$y = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix}; \quad \Psi = [\Psi_1 \quad \dots \quad \Psi_{n\theta}]; \quad \hat{\theta} = \begin{bmatrix} \hat{\theta}_1 \\ \hat{\theta}_2 \\ \vdots \\ \hat{\theta}_{n\theta} \end{bmatrix}; \quad \xi = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_{n\theta} \end{bmatrix} \quad (12)$$

e

$$\Psi_i = \begin{bmatrix} \Psi_i(1) \\ \Psi_i(2) \\ \vdots \\ \Psi_i(N) \end{bmatrix}, \quad i = 1, 2, \dots, n_\theta \quad (13)$$

Sabe-se que em identificação de sistemas não lineares a matriz de regressores ( $\psi$ ) possui muitas colunas, e quando se for calcular os parâmetros estimados ( $\hat{\theta}$ ) em  $\hat{\theta} = [\psi^T \psi]^{-1} \psi^T y$ , resultaria em estimativas inadequadas. Logo a utilização de métodos que baseados em OLS dos métodos CGS, MGS, e GH; seriam numericamente interessantes. Neste trabalho é utilizado o método de fatoração QR e decomposição de Householder. Detalhes deste método são encontrados em Golub e Van Loan (1996), Goldberg (1992), Stewart (1998) e Aguirre (2004). Na figura 1 é apresentado um fluxograma do procedimento de identificação de modelos NARMAX polinomiais baseado em Rodrigues (1996) utilizado neste trabalho.

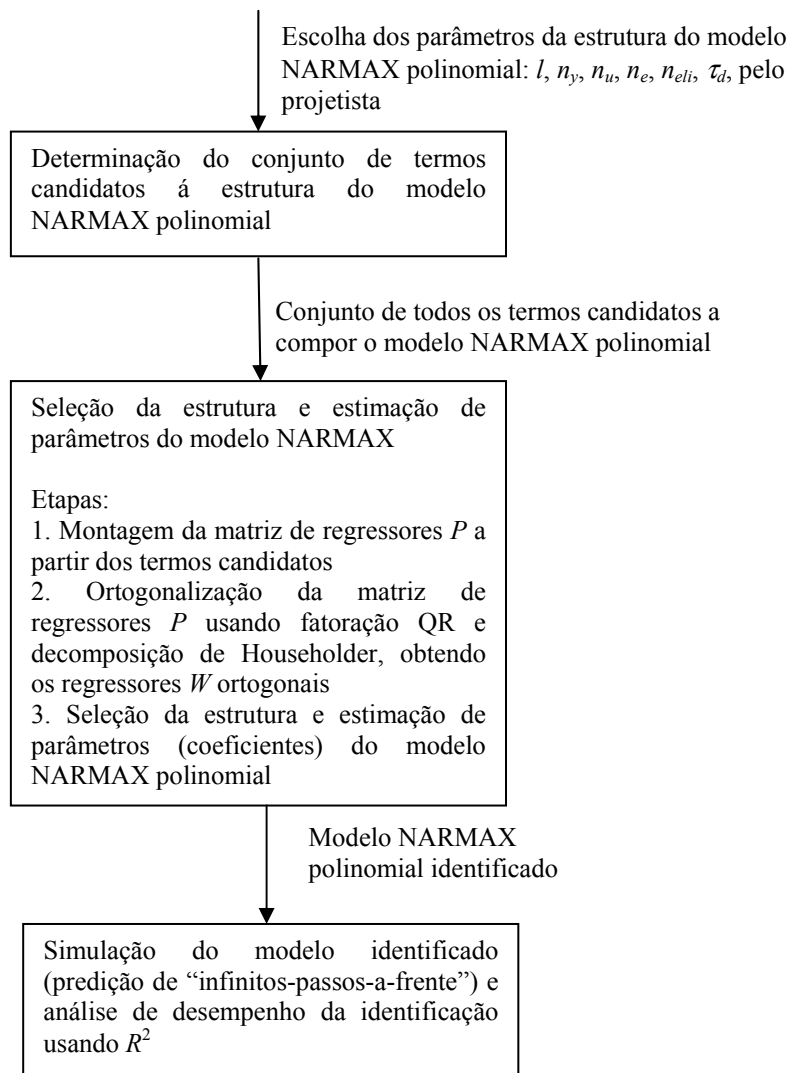


Figura 1 - Fluxograma do procedimento de identificação de modelos NARMAX polinomiais.

### 3. Identificação de um sistema não-linear

#### 3.1 Estudo de caso: Trocador de calor

O trocador de calor analisado consiste de um trocador de calor de fluxo de vapor líquido saturado, onde a água aquecida é aquecida por vapor de fluxo saturado pressurizado em um tubo de cobre. A variável de saída é a temperatura do líquido de saída. As variáveis de entrada são a taxa de fluxo de líquido, a temperatura de vapor, e temperatura do líquido de entrada. Neste contexto, a temperatura do vapor e a temperatura do líquido de entrada são mantidas

constantes em seus valores nominais (Daisy, 2006). O período de amostragem dos dados é de 1 segundo e o procedimento de identificação, adotado neste trabalho, utiliza como entrada a taxa de fluxo do líquido e como variável de saída a temperatura do líquido de saída. Uma representação do trocador de calor é apresentada na figura 2 (Bittanti & Piroddi, 1996).

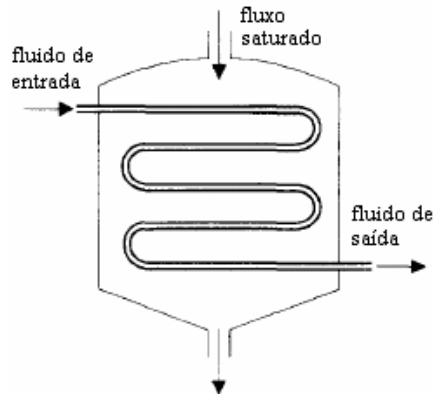


Figura 2 - Representação do trocador de calor.

O experimento foi configurado para utilizar as amostras 1 a 1200 ( $N=1200$ ) para a fase de estimação e as amostras 1201 a 1600, para a fase de validação do modelo NARMAX. Foram realizados diversos testes variando os valores das ordens  $n_y$ ,  $n_u$ ,  $n_e$  e o grau de não linearidade  $l$ . Na tabela 1 são apresentados os resultados dos testes realizados.

A identificação do trocador de calor é apropriada se um critério de erro definido previamente pelo projetista está entre valores admissíveis às necessidades do projeto. O critério escolhido neste estudo foi o coeficiente de correlação múltipla regido pela equação

$$R^2 = 1 - \frac{\sum_{t=1}^N [y(t) - \hat{y}(t)]^2}{\sum_{t=1}^{Na} [y(t) - \bar{y}]^2}, \quad (14)$$

onde  $N$  é o número de amostras avaliado,  $y(t)$  é saída real do processo,  $\hat{y}(t)$  é a saída estimada pelo modelo NARMAX, e  $\bar{y}$  é a média das medidas do sistema térmico. Quando o valor de  $R^2$  é igual a 1,0 indica uma aproximação exata do modelo aos dados medidos do processo. O valor de  $R^2$  entre 0,9 e 1,0 é considerado suficiente para aplicações práticas, principalmente em projetos de identificação e sistemas de controle baseados em modelo.

Tabela 1. Resultados das simulações na identificação do trocador de calor (predição de “infinitos-passos-a-frente”).

teste realizado	termos lineares do erro, $n_{eli}$	$l$	$n_y$	$n_u$	$n_e$	$R^2$ (estimação)	$R^2$ (validação)	$R^2$ (média harmônica)
1	0	1	1	1	0	0,7639	0	0
2	0	2	1	1	0	0,5771	0	0
3	0	1	2	1	0	0,2933	0	0
4	0	1	3	1	0	0,4298	0	0
5	0	1	1	2	0	0,8564	0,2656	0,4055
6	0	1	1	3	0	0,8888	0,4553	0,6021
7	0	1	1	4	0	0,9131	0,6129	0,7335
8	0	1	1	5	0	0,9056	0,5516	0,6856
9	0	1	2	4	0	0,8958	0,5457	0,6782
10	0	1	3	4	0	0,8997	0,5419	0,6764
11	0	2	1	4	0	0,9077	0,2009	0,3290
12	0	3	1	4	0	0,9476	0,6028	0,7369
13	0	3	2	4	0	0,9286	0,6263	0,7480
14	0	3	3	4	0	0,9154	0,5370	0,6769
15	1	3	2	4	1	0,9289	0,6394	0,7574
16	<b>1</b>	<b>3</b>	<b>2</b>	<b>4</b>	<b>2</b>	<b>0,9375</b>	<b>0,6624</b>	<b>0,7763</b>
17	2	3	2	4	1	0,9313	0,6436	0,7612
18	3	3	2	4	2	0,9289	0,6269	0,7486

Nota-se dos resultados apresentados na tabela 1, que o melhor resultado (melhor média harmônica de  $R^2$ ) foi para o teste 16 com  $n_y=2$ ,  $n_u=4$ ,  $n_e=2$ , grau de não linearidade  $l=3$  e um termo linear para o erro (isto é,  $e(t-1)$ ). Neste caso, a equação obtida para representar o trocador de calor possui 10 termos e é dada por:

$$\begin{aligned}
 y(t) = & 7,8267 + 1,0574y(t-1) - 0,13218y(t-2) - 0,47121u(t-3)u(t-2)y(t-1) \\
 & + 43,274u(t-3)u(t-2) - 0,55339u(t-4)u(t-1)y(t-1) + 53,357u(t-4)u(t-1) \\
 & - 2,8004u(t-2)u(t-1)u(t-1) - 1,4147u(t-4)u(t-4)u(t-4) - 0,18872 \cdot 10^{-4}e(t-1)y(t-1)y(t-1)
 \end{aligned}
 \tag{15}$$

Na figura 3 é apresentado o sinal de saída estimada e também o sinal de erro obtido com a utilização da equação (15). Neste caso o erro médio quadrático das fases de estimação e validação foram de 0,2116 e 0,2092, respectivamente.

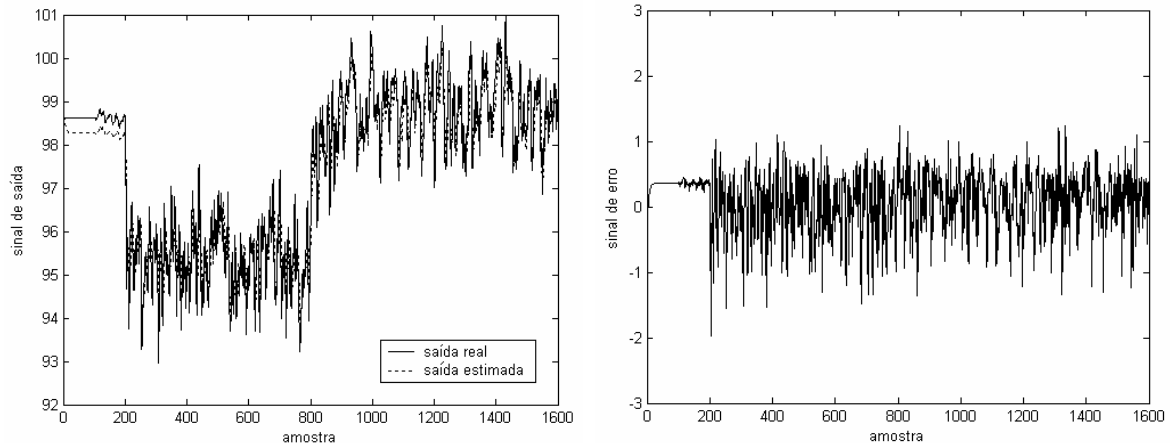


Figura 3 – Melhor modelo NARMAX obtido para representar o trocador de calor.

#### 4. Conclusão e futura pesquisa

Existem diversas maneiras de modelar um sistema, tais como a modelagem pela física do processo e a modelagem através de técnicas de identificação. A modelagem pela física do processo é uma abordagem analítica realizada através de conhecimento de leis físicas do processo, exigindo um conhecimento detalhado do mesmo, o que torna em alguns casos difícil a sua aplicação. A identificação, por outro lado, é uma abordagem experimental, que consiste no desenvolvimento de técnicas para a estimação de modelos de sistemas dinâmicos através de dados experimentais, não exigindo conhecimento prévio do processo, embora tal conhecimento possa ser utilizado se disponível.

A obtenção do modelo matemático de um sistema dinâmico é uma tarefa de vital importância na engenharia. A identificação de sistemas é uma abordagem que possibilita obter modelos matemáticos, através de dados de medição. Neste trabalho, um procedimento de identificação de sistemas não lineares baseado em uma representação polinomial *NARMAX* e algoritmo dos mínimos quadrados ortogonais é aplicado a um trocador de calor. Os resultados de simulação foram promissores e um modelo com 10 termos com  $n_y=2$ ,  $n_u=4$ ,  $n_e=2$ , grau de não linearidade  $l=3$  e um termo linear para o erro (isto é,  $e(t-1)$ ) foi o mais apropriado para representar o comportamento dinâmico do trocador de calor.

Em pesquisa futura, os autores desejam aprimorar o projeto do modelo *NARMAX* pela utilização de métodos de otimização não-linear para a determinação da ordem ( $n_y$ ,  $n_u$ ,  $n_e$ ) e o grau de não linearidade mais adequada para a identificação de sistemas térmicos.

#### 5. Referências

- Aguirre, L. A., 2004, “Introdução à Identificação de Sistemas: Técnicas Lineares e Não Lineares Aplicadas a Sistemas Reais”, Editora da UFMG, Belo Horizonte, MG.
- Billings, S. A., 1980, “Identification of Nonlinear Systems — a Survey”, IEE Proceedings Pt. D, Vol. 127, No. 6, pp. 272-285.
- Bittanti, S., Piroddi, L., 1996, “Nonlinear Identification and Control of a Heat Exchanger: a Neural Network Approach”, Journal of the Franklin Institute, Vol. 334B, No. 1, pp. 135-153.
- Cassini, C. C. S., 1999, “Estimação Recursiva de Características Estáticas Não Lineares Utilizando Modelos Polinomiais *NARMAX*”, Dissertação de Mestrado do PPGE, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.
- Chen, S., Billings, S. A, Luo, W., 1989, “Orthogonal Least Squares Methods and their Application to Non-linear System Identification”, International Journal of Control, Vol. 50, No. 5, pp. 1873-1896.

- Chen, S., Billings, S. A., 1989, "Representations of Nonlinear Systems: the NARMAX model", International Journal of Control, Vol. 49, No. 3, pp. 1013-1032.
- Daisy, 2006, Database for the Identification of Systems, <http://www.esat.kuleuven.ac.be/sista/daisy/> [Acesso 16/02/2006].
- Goldberg, J. L., 1992, "Matrix Theory with Applications", McGraw-Hill International Editions, New York, NY, USA.
- Golub, G. H., Van Loan, C. F., 1996, "Matrix Computations", 3rd edition, The Johns Hopkins University Press, London, UK.
- Haber, R., Unbehauen, H., 1990, "Structure Identification of Nonlinear Dynamic Systems — a Survey on Input/Output Approaches", Automatica, Vol. 26, No. 4, pp. 651-677.
- Haykin, S., 1996, "Neural Networks", 2nd edition, Prentice-Hall, Upper Saddle River, NJ, USA.
- Johansson, R., 1993, "System Modeling and Identification", Prentice-Hall, Englewood Cliffs, USA.
- Korenberg, M., Billings, S. A., Liu, Y. P.; McIlroy, P. J., 1988, "Orthogonal Parameter Estimation Algorithm for Non-Linear Stochastic Systems", International Journal of Control, Vol. 48, No. 1, pp. 193-210.
- Kukreja, S. L., Galiana, H. L., Kearney, R. E., 2003, "NARMAX Representation and Identification of Ankle Dynamics", IEEE Transactions on Biomedical Engineering, Vol. 50, No. 1, pp. 70-81.
- Leontaris, I. J., Billings, S. A., 1987, "Input-output Parametric Models for Non-linear Systems – Part 1: Deterministic Non-linear Systems", International Journal of Control, vol. 41, pp. 303-344.
- Ljung, L., 2001, "Black-box Models from Input-output Measurements", Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference, Budapest, Hungary, pp. 138-146.
- Matko, D., Zupancic, B., Karba\_R., 1992, "Simulation and Modelling of Continuous Systems - A Case Study Approach", Prentice Hall International, UK.
- Rodrigues, G. G., 1996, "Identificação de Sistemas Dinâmicos Não Lineares Utilizando Modelos NARMAX Polinomiais - Aplicação a Sistemas Reais", Dissertação de Mestrado do PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, MG, Brasil.
- Sjöberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, P. G. B., Hjalmarsson, H., Juditsky, A., 1995, "Nonlinear Black-box Modeling in System Identification: a Unified Overview", Automatica, Vol. 31, No. 12, pp. 1691-1724.
- Stewart, G. W., "Matrix Algorithms: Volume I: Basic Decompositions", SIAM, Society for Industrial and Applied Mathematics, Philadelphia, USA.
- Zito, G.; Landau, I. D., 2005, "NARMAX Model Identification of a Variable Geometry Turbocharged Diesel Engine", Proceedings of American Control Conference, Portland, OR, USA, pp. 1021-1026.

## POLYNOMIAL NARMAX IDENTIFICATION OF A HEAT EXCHANGER USING ORTHOGONAL LEAST-SQUARES ALGORITHM

Leandro dos Santos Coelho, Marcelo Wicthoff Pessôa e Ricardo Massao Suguimoto

Laboratório de Automação e Sistemas, LAS  
Programa de Graduação em Engenharia de Produção e Sistemas, PPGEPS  
Pontifícia Universidade Católica do Paraná – PUCPR  
Rua Imaculada Conceição, 1155, CEP 80215-901, Curitiba, PR, Brasil  
leandro.coelho@pucpr.br; celowp@gmail.com; ricardo.ms@gmail.com

### Abstract

The problem of parameter estimation is of great relevance in system identification problems regardless of which nonlinear representation is chosen. In the literature, this problem has been solved in different ways depending on the representation used. In this work, a nonlinear system identification procedure, based on a polynomial NARMAX (*Nonlinear AutoRegressive Moving Average with eXogenous inputs*) representation and orthogonal least-squares algorithm, is applied to a heat exchanger. Polynomial NARMAX models describe nonlinear systems in terms of linear-in-the-parameters difference equations relating the current output to (possibly nonlinear) combinations of inputs and past outputs. It is suitable for modeling both the stochastic and deterministic components of a system and is capable of describing a wide variety of nonlinear systems. In this work, the regressor vector used to compose the NARMAX model is carefully chosen, among hundreds of candidates, by an automatic procedure. The simulation result obtained was very satisfactory for the infinite-step ahead identification of heat exchanger output.

Keywords: heat exchanger, nonlinear identification, NARMAX model, orthogonal least-squares algorithm.

## EFEITO DE PARÂMETROS DO MÉTODO MULTIGRID CS E FAS SOBRE O TEMPO DE CPU PARA A EQUAÇÃO DE LAPLACE BIDIMENSIONAL

**Márcio Augusto Villela Pinto**

Departamento de Matemática e Estatística, Universidade Estadual de Ponta Grossa, Ponta Grossa, PR, Brasil  
marciovp@demec.ufpr.br

**Carlos Henrique Marchi**

Departamento de Engenharia Mecânica, Universidade Federal do Paraná, Curitiba, PR, Brasil  
marchi@demec.ufpr.br

**Resumo.** Sobre o tempo de CPU necessário para resolver numericamente a equação de Laplace bidimensional, verifica-se o efeito causado por: número de nós, de iterações internas e de malhas, três solvers (Gauss-Seidel, MSI e ADI), e esquemas de correção (CS) e de aproximação completa (FAS) com multigrid geométrico e ciclo V. O método de diferenças finitas é usado para discretizar a equação diferencial com um esquema de 2<sup>a</sup> ordem de acurácia. Verificou-se que: (1) o esquema FAS é mais rápido do que o CS; (2) o solver MSI é mais rápido do que Gauss-Seidel e ADI; (3) o número ótimo de iterações internas é 1 para o CS e entre 3 e 5 para o FAS; e (4) recomenda-se usar o número máximo possível de malhas.

*Palavras-chave:* solver, CFD, diferenças finitas, transferência de calor, métodos numéricos.

### 1. Introdução

Modelos matemáticos na dinâmica dos fluidos computacional surgem em fenômenos físicos que envolvem fluidos em movimento, com ou sem troca de calor (Fortuna, 2000; Maliska, 2004). Estes modelos matemáticos, em geral, não têm soluções analíticas conhecidas. Buscam-se então soluções numéricas transformando-se o modelo contínuo em um modelo discreto. Um método de discretização muito usado é o método de diferenças finitas (Golub e Ortega, 1992; Tannehill *et al.*, 1997), onde, em problemas bidimensionais, o domínio  $(x, y) \in \mathbb{R}^2 : 0 \leq x, y \leq 1$  é particionado em um número de incógnitas (ou número de pontos), dado por:

$$N = N_x N_y, \quad (1)$$

onde  $N_x$  e  $N_y$  são o número de incógnitas nas direções coordenadas  $x$  e  $y$ , respectivamente.

Isto introduz uma malha com os pontos

$$(x_i, y_j) = ((i-1)h_x, (j-1)h_y), \text{ com } h_x = \frac{1}{N_x - 1} \text{ e } h_y = \frac{1}{N_y - 1}, \quad (2)$$

onde  $i = 1, \dots, N_x$ ,  $j = 1, \dots, N_y$  e  $h_x$  e  $h_y$  os comprimentos de cada elemento nas direções coordenadas  $x$  e  $y$ , respectivamente. Neste caso, estabelece-se uma malha com elementos de tamanho  $h_x$  por  $h_y$  que se denota por  $\Omega^h$ .

A discretização desses modelos matemáticos conduz a grandes sistemas de equações algébricas do tipo

$$\mathbf{A}\vec{v} = \vec{f}, \quad (3)$$

onde  $\mathbf{A}$  é uma matriz quadrada,  $\vec{f}$  é o vetor independente e  $\vec{v}$  é o vetor de incógnitas. A estrutura da matriz  $\mathbf{A}$  depende do método e das aproximações numéricas usados para discretizar o modelo matemático.

Várias técnicas numéricas têm sido estudadas para resolver o sistema dado pela Eq. (3) com o menor custo computacional e a solução a mais próxima possível da exata (sem erros de iteração, Ferziger e Peric, 1999). A resolução por métodos diretos não é recomendável, visto que na prática, a matriz dos coeficientes é muito grande e o custo da inversão da matriz é alto (Golub e Van Loan, 1989). Para problemas de grande porte os métodos iterativos são mais adequados (Burden e Faires, 1997). O método do gradiente conjugado (Burden e Faires, 1997; Golub e Ortega, 1992), introduzido por Hestenes e Stiefeld (1952), usa técnicas que são mais específicas para geometrias simples e é um método sensível ao condicionamento da matriz.

O método *multigrid*, proposto originalmente por Fedorenko (1964), é atualmente um método numérico muito usado para resolver iterativamente sistemas de equações do tipo da Eq. (3). A idéia básica é usar um conjunto de malhas e executar alternativamente iterações em cada nível de malha e soluções aproximadas desta equação em malhas mais grossas (Briggs *et al.*, 2000). São usados operadores para transferir informações da malha fina para a malha imediatamente mais grossa (processo chamado de restrição) e da malha grossa para a malha imediatamente mais fina (processo de prolongação). Briggs *et al.* (2000) trabalharam com a razão de engrossamento  $r = 2$  afirmando ser uma prática universal e que  $r \neq 2$  não traz vantagens. Em cada malha o sistema de equações é resolvido com um método iterativo com propriedades de reduzir rapidamente os erros oscilatórios (propriedades de suavização). Com este conceito, vários trabalhos (Brandt, 1977; Stüben, 1999; Wesseling e Oosterlee, 2001; Moro, 2004; Pinto *et al.*, 2005a e 2005b), apresentaram bons resultados numéricos para problemas de dinâmica dos fluidos. Os resultados de Fedorenko (1964) mostram que a taxa de velocidade de convergência com o uso da técnica *multigrid* é muito melhor que a dos métodos iterativos puros. O objetivo do método *multigrid* é acelerar a convergência a fim de reduzir o tempo de CPU necessário para resolver a Eq. (3). Os melhores desempenhos do método *multigrid* são obtidos em problemas totalmente elípticos (Wesseling, 1992), ou seja, problemas dominados pela difusão; e os menores em problemas dominados pela advecção (Ferziger e Peric, 1999).

O método *multigrid* pode ser aplicado a malhas estruturadas, conhecido como *multigrid* geométrico (Wesseling e Oosterlee, 2001), bem como a malhas não-estruturadas, conhecido como *multigrid* algébrico (Stüben, 2001). Em Wesseling e Oosterlee (2001) são apontados vários desafios para o *multigrid* geométrico, como: a solução das equações de Navier-Stokes, problemas com perturbações singulares, problemas de camada limite onde aparecem as malhas fortemente alongadas, ou mesmo a paralelização de algoritmos.

Neste artigo os seguintes parâmetros são estudados: número de incógnitas, número de iterações internas (número de iterações do método numérico a fim de suavizar as componentes de erro), o número de níveis (número de malhas percorridas) e os *solvers* Gauss-Seidel (GS), MSI e ADI (Tannehill *et al.*, 1997). Os objetivos são: verificar o efeito desses parâmetros sobre o tempo de CPU para o *multigrid* geométrico e realizar comparações entre o esquema de correção (CS) e o esquema de aproximação completa (FAS) com ciclo V proposto em Wesseling (1992). Os resultados são comparados com os obtidos na bibliografia. São usados operadores de restrição por injeção e prolongação por interpolação bilinear (Briggs *et al.*, 2000). O modelo matemático considerado neste trabalho envolve um problema bidimensional linear de condução de calor, ou seja, a equação de Laplace com condições de contorno de Dirichlet.

Este artigo está organizado da seguinte forma: na seção 2 é apresentada uma visão geral do método *multigrid*. Na seção 3, é apresentado o modelo matemático e numérico. Na seção 4 são descritos os experimentos numéricos e seus resultados. E na seção 5 é apresentado a conclusão do trabalho.

## 2. O método *multigrid*

Para reduzir o erro de discretização, malhas muito refinadas são necessárias a fim de se resolver problemas de mecânica dos fluidos e transferência de calor. Isso gera sistemas de equações muito grandes. A resolução destes problemas através de métodos numéricos requer um custo computacional demasiadamente alto e muitas vezes inviável devido ao grande número de equações a serem resolvidas em cada passo iterativo. Uma opção para melhorar a taxa de convergência destes problemas é o método *multigrid* (Briggs *et al.*, 2000), que acelera consideravelmente a resolução dos sistemas lineares envolvidos no problema. Métodos *multigrid* são métodos iterativos para a solução de sistemas lineares, sendo, portanto, fortemente dependentes da estimativa inicial atribuída às incógnitas do problema.

Uma técnica eficiente usada para aliviar as fortes oscilações do resíduo da Eq. (3) em cada malha, definido por:

$$\bar{R} = \bar{f} - \mathbf{A}\bar{v}, \quad (4)$$

é suavizar as oscilações por um método de relaxação (método iterativo). Neste trabalho pretende-se comparar o desempenho dos métodos: GS, MSI e ADI.

As primeiras iterações deste processo, geralmente, têm rápida convergência, caracterizando a presença de modos oscilatórios de erro. Porém, após algumas iterações o processo torna-se lento, sinalizando a predominância de modos suaves (Brandt, 1977). Este é exatamente o momento onde é recomendável transferir o problema de relaxação para uma malha mais grossa, pois os modos de erros suaves na malha fina tornam-se erros oscilatórios na malha grossa (Wesseling, 1992).

Podem ser usados dois tipos de esquemas com o método *multigrid* (Briggs *et al.*, 2000): o esquema de correção (*Correction Scheme*, CS) e o esquema de aproximação completa (*Full Approximation Scheme*, FAS). NO esquema CS, a Eq. (3) é resolvida apenas na malha mais fina; nas malhas mais grossas, resolve-se a equação do resíduo. Já no caso do esquema FAS, a Eq. (3) é resolvida em todas as malhas. O esquema CS é geralmente utilizado em problemas lineares e o esquema FAS em problemas não-lineares.

A taxa de convergência ideal (teórica) do *multigrid* é independente do tamanho da malha, isto é, independe do número de pontos da malha (Hirsch, 1988; Ferziger e Peric, 1999). Não é muito efetivo usar somente dois níveis de malha (Roache, 1998); para obter um bom desempenho do *multigrid*, diversos níveis de malhas devem ser usados (Tannehill *et al.*, 1997). Pinto *et al.* (2005a e 2005b) recomendam usar todos os níveis.

Os operadores de transferência da malha fina para a malha grossa são chamados de operadores de restrição e são denotados genericamente por  $I_h^H \vec{\phi}^h = \vec{\phi}^H$ . Onde  $\vec{\phi}$ , no caso do esquema CS, assume o resíduo  $\vec{R}$  dado pela Eq. (4) e no caso do esquema FAS assume a solução aproximada do problema além do resíduo  $\vec{R}$ . Neste trabalho faz-se uso do operador de injeção para problemas bidimensionais que pode ser visto em Briggs *et al.* (2000).

Os operadores de transferência da malha grossa para a malha fina são chamados de operadores de prolongação, ou interpolação, e são denotados genericamente por  $I_H^h \vec{\phi}^H = \vec{\phi}^h$ . Onde  $\vec{\phi}$ , no caso do esquema CS assume a aproximação do erro na equação residual, ou seja, a correção, e no caso do esquema FAS assume a solução aproximada do problema, além da correção. Neste trabalho faz-se uso do operador de interpolação bilinear que também pode ser visto em Briggs *et al.* (2000).

### 3. Modelos matemático e numérico

O problema linear de condução de calor bidimensional (equação de Laplace) com condições de contorno de Dirichlet, em coordenadas cartesianas, considerado neste trabalho é (Maliska, 2004):

$$\begin{cases} \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0, & 0 < x, y < 1 \\ T(x,1) = \text{sen}(\pi \cdot x), & T(x,0) = T(0, y) = T(1, y) = 0 \end{cases}, \quad (5)$$

onde  $T$  é a incógnita e representa a temperatura. A solução analítica do problema é

$$T(x, y) = \text{sen}(\pi \cdot x) \frac{\text{senh}(\pi \cdot y)}{\text{senh}(\pi)}. \quad (6)$$

A discretização do domínio é feita com malhas uniformes cujos pontos são dados pela Eq. (2). Para cada um dos  $(N_x - 2) \times (N_y - 2)$  pontos interiores da malha, a Eq. (5) é discretizada com o método de diferenças finitas com diferença central (CDS) (Tannehill *et al.*, 1997), resultando em

$$\begin{cases} \frac{v_{i-1,j} - 2v_{i,j} + v_{i+1,j}}{h_x^2} + \frac{v_{i,j-1} - 2v_{i,j} + v_{i,j+1}}{h_y^2} = 0, & 2 \leq i \leq N_x - 1, \quad 2 \leq j \leq N_y - 1 \\ v_{i,N_y} = \text{sen}(\pi \cdot x_{i,N_y}), & v_{i,1} = v_{1,j} = v_{N_x,j} = 0 \end{cases}, \quad (7)$$

onde  $v_{i,j}$  é uma aproximação (solução numérica) para a solução exata  $T(x_i, y_j)$ .

Rearranjando os termos da Eq. (7), obtém-se

$$a_p v_p = a_n v_n + a_s v_s + a_w v_w + a_e v_e + b_p, \quad (8)$$

onde os coeficientes são dados por  $a_p = 2/h_x^2 + 2/h_y^2$ ,  $a_n = 1/h_x^2$ ,  $a_s = 1/h_y^2$ ,  $a_w = 1/h_x^2$ ,  $a_e = 1/h_x^2$  e  $b_p = 0$ .

Se  $\vec{v}$  e  $\vec{f}$  são denotados por  $\vec{v} = (v_1, \dots, v_N)^t$  e  $\vec{f} = (f_1, \dots, f_N)^t$ , respectivamente, onde  $\vec{f}$  é o vetor independente formado pelos termos  $b_p$ , então o sistema da Eq. (7), pode ser representado por um sistema de equações algébricas do tipo dado pela Eq. (3), onde  $\mathbf{A}$  é uma matriz pentadiagonal  $N$  por  $N$ , simétrica e definida positiva (Briggs *et al.*, 2000; Burden e Faires, 1997).

A Eq. (3) é resolvida com o método *multigrid* fazendo-se uso dos dois esquemas (CS e FAS) citados na seção anterior. Neste caso, os sistemas de equações do tipo da Eq. (3), onde  $\vec{f}$  agora representa o termo fonte (resíduo ou solução aproximada) a cada nível de malha, são resolvidos com os *solvers* GS, MSI e ADI. Resolve-se também o problema, apenas na malha mais fina, com os métodos *singlegrid*: GS, MSI, ADI e Eliminação de Gauss (Elim.Gauss). O número de iterações internas (*ITI*) do método *multigrid* é o número de iterações do *solver* (método numérico) a fim de suavizar os componentes do erro.



## 4. Resultados

Os algoritmos foram implementados na linguagem FORTRAN 95 com o uso do aplicativo Compaq Visual Fortran 6.6 usando-se precisão dupla. As simulações foram realizadas num microcomputador com processador Intel Pentium 4 com 2.66 GHz e 1 GB de RAM.

Cerca de 400 simulações foram realizadas com as seguintes variantes: número de incógnitas ( $N$ ), número de iterações internas ( $ITI$ ), número de níveis de malhas ( $L$ ) e *solvers* (GS, MSI e ADI). São apresentados neste trabalho os resultados mais representativos.

O critério de convergência para as iterações externas ( $ITE$ ) (número de ciclos necessários para suavizar as componentes de erro) é baseado na razão entre a norma  $L_1$  do resíduo (Ferziger e Peric, 1999) numa determinada iteração e a norma do resíduo da estimativa inicial. O resíduo de cada nó é calculado através da Eq. (4). Neste trabalho adota-se  $r = 2$ . Adota-se também  $\varepsilon = 10^{-7}$  e  $\vec{v} = (0,0,\dots,0)$  para a tolerância sobre o critério de convergência e a estimativa inicial, respectivamente.

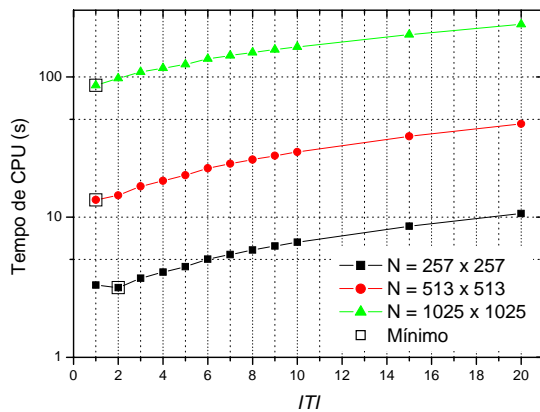
O foco deste trabalho é a minimização do tempo de CPU. Entende-se por tempo de CPU, o tempo gasto para gerar malhas, atribuir a estimativa inicial, calcular os coeficientes e resolver o sistema linear da Eq. (3). Este tempo é medido usando-se a função TIMEF da biblioteca PORTLIB do FORTRAN 95. Através de testes realizados verificou-se que a incerteza desta função é aproximadamente de  $\pm 0.05$  s.

### 4.1. Esquema de correção (CS)

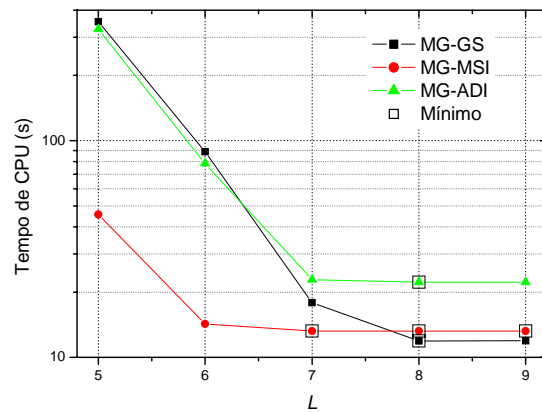
#### 4.1.1. Iterações internas ( $ITI$ )

A Fig. 1a mostra a influência do número de iterações internas ( $ITI$ ) sobre o tempo de CPU para alguns valores de  $N$  e o *solver* MSI. Verificou-se que, para cada malha, em geral o menor tempo de CPU ocorre com o menor valor de  $ITI$ . Aumentando-se o valor de  $ITI$ , em geral aumenta-se significativamente o tempo de CPU. Este comportamento da Fig. 1a também ocorre com os outros *solvers* testados: GS e ADI.

Para os três *solvers* testados, a Tab. 1 mostra o número ótimo de iterações internas ( $ITI_{\text{ótimo}}$ ), que é o  $ITI$  que resulta no menor tempo de CPU. Observa-se que  $N$  e o *solver* influenciam muito pouco o valor de  $ITI_{\text{ótimo}}$ . Para  $N$  muito grande,  $ITI_{\text{ótimo}} = 1$  com qualquer *solver*. Portanto, recomenda-se utilizar  $ITI = 1$  com o esquema CS e os *solvers* GS, MSI e ADI. Este valor de  $ITI_{\text{ótimo}}$  difere daquele encontrado (Pinto *et al.*, 2005) em problemas unidimensionais lineares (equações de Poisson e advecção-difusão), onde  $ITI_{\text{ótimo}} = 3$  para o *solver* GS e o mesmo método *multigrid*.



a) Número de iterações internas ( $ITI$ ) com *solver* MSI.



b) Número de níveis de malhas ( $L$ ) para  $N = 513 \times 513$ .

Figura 1. Método *multigrid* (MG) com esquema CS.

Tabela 1. Número ótimo de iterações internas ( $ITI_{\text{ótimo}}$ ) para o esquema CS.

$N$	GS	MSI	ADI
257 x 257	2	2	1
513 x 513	1	1	1
1025 x 1025	1	1	1

#### 4.1.2. Níveis de malhas ( $L$ )

A Fig. 1b mostra a influência do número de níveis de malhas ( $L$ ) sobre o tempo de CPU, para  $N = 513 \times 513$  nós e os três *solvers* testados. Para esta malha,  $L_{máximo} = 9$ , isto é, usando-se a razão de engrossamento  $r = 2$ , para resolver a malha mais fina de  $513 \times 513$  nós, pode-se usar no máximo mais 8 malhas, que são:  $257 \times 257$ ,  $129 \times 129$ ,  $65 \times 65$ ,  $33 \times 33$ ,  $17 \times 17$ ,  $9 \times 9$ ,  $5 \times 5$  e  $3 \times 3$ , que apresenta apenas um nó interno. Verificou-se que, para cada *solver*, o número ótimo de níveis de malha ( $L_{ótimo}$ ), ocorre em geral para  $L_{ótimo} = L_{máximo} - 1$ , onde  $L_{ótimo}$  é o  $L$  que resulta no menor tempo de CPU. Diminuindo-se o valor de  $L$ , em geral aumenta-se significativamente o tempo de CPU. Este resultado é praticamente o mesmo encontrado (Pinto *et al.*, 2005) em problemas unidimensionais lineares (equações de Poisson e advecção-difusão), onde  $L_{ótimo} \approx L_{máximo}$  para o *solver* GS e o mesmo método *multigrid*. Portanto, recomenda-se utilizar  $L_{ótimo} = L_{máximo}$  com o esquema CS e os *solvers* GS, ADI e MSI.

#### 4.1.3. Número de incógnitas ( $N$ )

A Fig. 2 mostra a influência do número de incógnitas ( $N$ ) sobre o tempo de CPU para os três *solvers* (GS, MSI e ADI) usados com o método *multigrid* (MG). Também são mostrados resultados para os mesmos três *solvers* mas usando-se o método padrão de malha única, aqui chamado de *singlegrid* (SG). Além disso são apresentados resultados para um método direto, eliminação de Gauss (Elim.Gauss), também baseado em malha única. As malhas usadas são:  $5 \times 5$ ,  $9 \times 9$ , ... até  $65 \times 65$  (Elim.Gauss) ou  $513 \times 513$  (SG) ou  $2049 \times 2049$  (MG) nós.

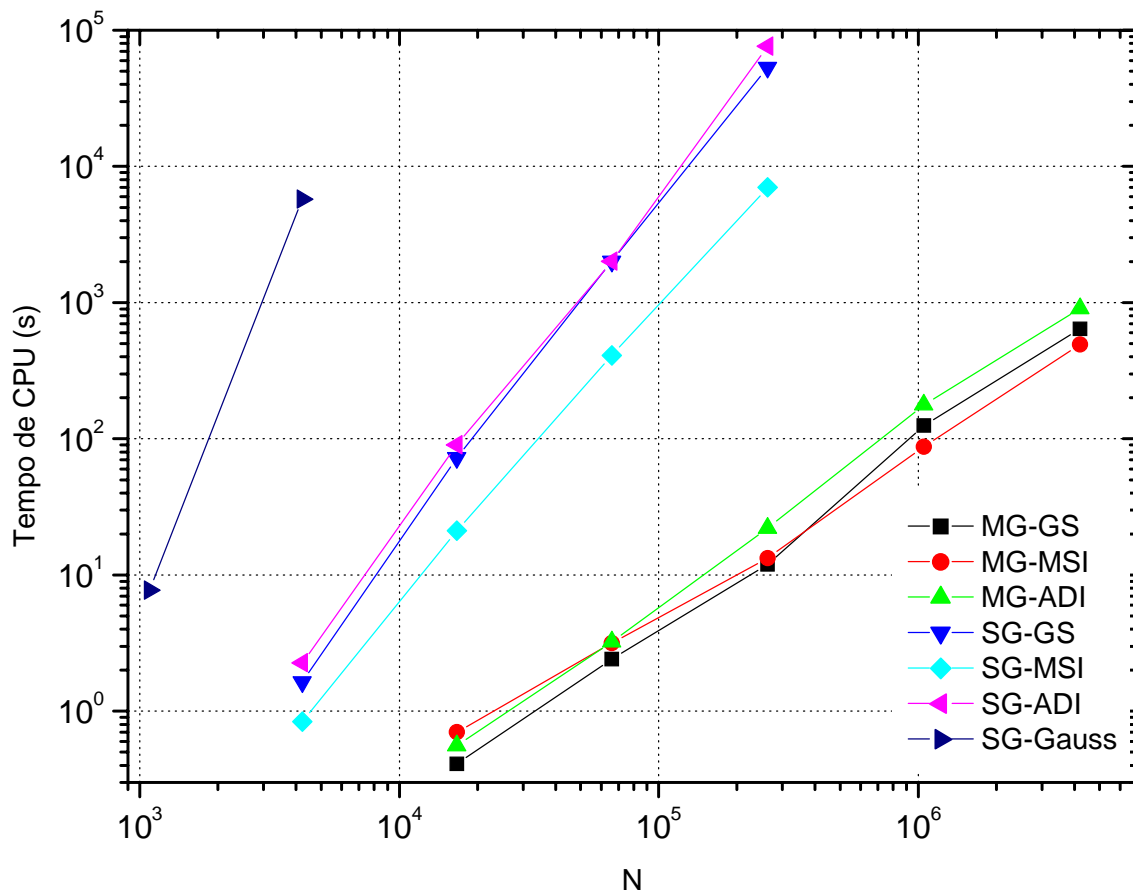


Figura 2. Tempo de CPU *versus* número de incógnitas ( $N$ ) com esquema CS.

Na Fig. 2 são mostrados apenas os pontos cujo tempo de CPU não é muito influenciado pela incerteza de sua medição. Verificou-se que, para malhas com  $N > 10^4$ , o tempo de CPU do método *multigrid* com qualquer *solver* é significativamente menor do que o método *singlegrid* iterativo, que por sua vez é significativamente menor do que o método *singlegrid* direto (Elim.Gauss). Por exemplo, para a malha  $65 \times 65$  nós, o tempo de CPU do SG-MSI e SG-Elim.Gauss é de 0,84 s e 5737,22 s, respectivamente, ou seja, o SG-MSI é cerca de 6830 vezes mais rápido do que o SG-Elim.Gauss. Outro exemplo, para a malha  $513 \times 513$  nós, o tempo de CPU do MG-MSI e SG-MSI é de 13,3 s e 7006 s, respectivamente, ou seja, o MG-MSI é cerca de 527 vezes mais rápido do que o SG-MSI. Estas diferenças ficam cada vez maiores à medida que  $N$  aumenta, pois as inclinações das curvas dos *solvers* com MG são menores do que com SG. Entre os métodos iterativos, em geral, observa-se que:

$$t_{CPU}(MSI) < t_{CPU}(GS) < t_{CPU}(ADI). \tag{9}$$

Para os pontos da Fig. 2, a Tab. 2 apresenta a inclinação ( $p$ ) das curvas, obtida por ajuste geométrico de mínimos quadrados considerando a seguinte função:

$$t_{CPU} = cN^p, \tag{10}$$

onde  $p$  representa a ordem do *solver* associado ao método empregado, e  $c$  é um coeficiente que depende de cada método e cada *solver*. O método *multigrid* ideal é aquele cujo  $p = 1$ , isto é, aquele cujo tempo de CPU cresce linearmente com o tamanho do problema, ou com o número de incógnitas  $N$ . Portanto, para cada método e *solver*, quanto mais próximo da unidade estiver seu  $p$ , melhor é o seu desempenho. A ordem dos *solvers* com o *multigrid* é pouco afetada se o esquema adotado é o CS ou o FAS. Mas a ordem é muito afetada em função do método adotado, se direto ou iterativo, e se *singlegrid* ou *multigrid*. Entre os três *solvers* testados, o MSI é o mais eficiente com qualquer método empregado, isto é, requer o menor tempo de CPU para resolver um dado sistema com  $N$  incógnitas. E entre os métodos, o MSI é o mais eficiente com o método *multigrid* e o esquema FAS.

Pela Tab. 2, pode-se notar que, tanto para os métodos *singlegrid*, quanto para os métodos *multigrid* com o uso do esquema FAS, quanto mais fortemente implícito for o método, mais rápido ele é. Esta análise sofre uma pequena variação no caso dos métodos *multigrid* com o uso do esquema CS quando se trata dos métodos GS e ADI, mas mesmo assim, o método mais fortemente implícito (MSI) continua sendo o mais rápido.

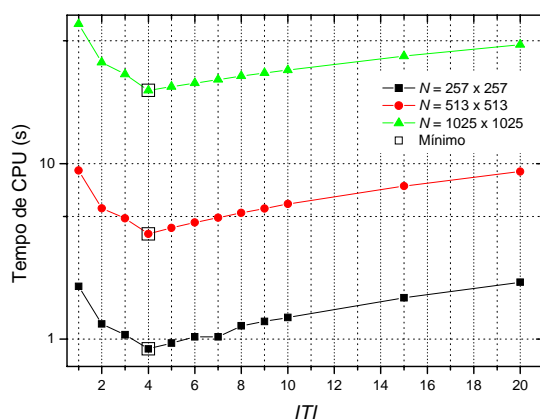
Tabela 2. Ordem ( $p$ ) dos métodos e *solvers*.

<i>Solver</i>	Tipo de <i>solver</i>	<i>Singlegrid</i>	Esquema CS	Esquema FAS
Elim.Gauss	direto	4.88	---	---
GS	iterativo	2.50	1.35	1.40
ADI	iterativo	2.50	1.36	1.39
MSI	iterativo	2.18	1.19	1.16

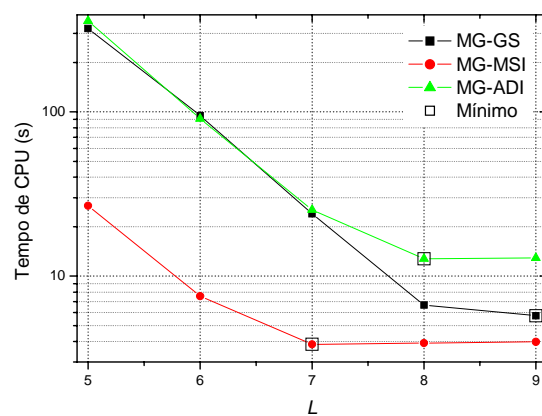
## 4.2. Esquema de aproximação completa (FAS)

### 4.2.1. Iterações internas (ITI)

A Fig. 3a mostra a influência do número de iterações internas (*ITI*) sobre o tempo de CPU para alguns valores de  $N$  e o *solver* MSI. Verificou-se que, para cada malha, o menor tempo de CPU ocorre com  $ITI_{\text{ótimo}} = 4$ . Diminuindo-se ou aumentando-se o valor de *ITI* em relação ao ótimo, pode-se aumentar significativamente o tempo de CPU. Este comportamento da Fig. 3a também ocorre com os outros *solvers* testados: GS e ADI. A Tab. 3 mostra o número ótimo de iterações internas ( $ITI_{\text{ótimo}}$ ) para os três *solvers* testados. Observa-se que  $N$  e o *solver* influenciam muito pouco o valor de  $ITI_{\text{ótimo}}$ . Como na prática procura-se resolver problemas de grande porte, recomenda-se utilizar  $ITI = 3, 4$  ou  $5$ , respectivamente, para os *solvers* GS, MSI e ADI.



a) Número de iterações internas (*ITI*) com *solver* MSI.



b) Número de níveis de malhas ( $L$ ) para  $N = 513 \times 513$ .

Figura 3. Método *multigrid* (MG) com esquema FAS.

Tabela 3. Número ótimo de iterações internas ( $ITI_{\text{ótimo}}$ ) para o esquema FAS.

$N$	GS	MSI	ADI
257 x 257	4	4	5
513 x 513	3	4	5
1025 x 1025	3	4	5

Na seção anterior, verificou-se que para o esquema CS,  $ITI_{\text{ótimo}} = 1$  com qualquer *solver*. Já nesta seção, verificou-se que para o esquema FAS,  $ITI_{\text{ótimo}} = 3$  a 5, dependendo do *solver*. Portanto, o esquema utilizado (CS ou FAS) influencia o número ótimo de iterações internas do método *multigrid*.

#### 4.2.2. Níveis de malhas ( $L$ )

A Fig. 3b mostra a influência do número de níveis de malhas ( $L$ ) sobre o tempo de CPU, para  $N = 513 \times 513$  nós e os três *solvers* testados. Verificou-se que, para cada *solver*, o número ótimo de níveis de malha ( $L_{\text{ótimo}}$ ) ocorre na média para  $L_{\text{ótimo}} = L_{\text{máximo}} - 1$ . Diminuindo-se o valor de  $L$ , em geral aumenta-se significativamente o tempo de CPU. Portanto, recomenda-se utilizar  $L_{\text{ótimo}} = L_{\text{máximo}}$  com o esquema FAS e os *solvers* GS, MSI e ADI. Na seção anterior, verificou-se que para o esquema CS,  $L_{\text{ótimo}} = L_{\text{máximo}} - 1$ , com qualquer *solver*. Já nesta seção, ocorre o mesmo para o esquema FAS. Portanto o esquema utilizado (CS ou FAS) não tem grande influência sobre o número ótimo de níveis para o método *multigrid*.

#### 4.2.3. Número de incógnitas ( $N$ )

A Fig. 4 mostra a influência do número de incógnitas ( $N$ ) sobre o tempo de CPU para os três *solvers* (GS, MSI e ADI) usados com o método *multigrid* (MG). Também são mostrados resultados para os mesmos três *solvers* mas usando-se o método *singlegrid* (SG) e eliminação de Gauss (Elim.Gauss) também baseado em malha única. As malhas usadas são:  $5 \times 5, 9 \times 9, \dots$  até  $65 \times 65$  (Elim.Gauss) ou  $513 \times 513$  (SG) ou  $2049 \times 2049$  (MG) nós.

Na Fig. 4 são mostrados apenas os pontos cujo tempo de CPU não é muito influenciado pela incerteza de sua medição. Verificou-se que, para malhas com  $N > 10^4$ , o tempo de CPU do método *multigrid* com qualquer *solver* é significativamente menor do que o método *singlegrid* iterativo, que por sua vez é significativamente menor do que o método *singlegrid* direto (Elim.Gauss). Por exemplo, para a malha  $513 \times 513$  nós, o tempo de CPU do MG-MSI e SG-MSI é de 3,8 s e 7006 s, respectivamente, ou seja, o MG-MSI é cerca de 1844 vezes mais rápido do que o SG-MSI. Estas diferenças ficam cada vez maiores à medida que  $N$  aumenta, pois as inclinações das curvas dos *solvers* com MG são menores do que com SG. Entre os métodos iterativos, em geral, para o esquema FAS também vale a Eq. (9). Para os pontos da Fig. 4, a Tab. 2 apresenta a inclinação ( $p$ ) das curvas, obtida por ajuste de mínimos quadrados considerando a função dada na Eq. (10).

#### 4.3. Esquema CS versus FAS

Na Fig. 5 se faz uma comparação entre os esquemas *multigrid* CS e FAS em função de diversos valores de  $N$ . Pode-se notar que o *solver* MSI que é o mais eficiente com qualquer esquema. Pode-se notar ainda que, o esquema FAS é mais rápido do que o CS em qualquer  $N$ , entre 3,2 e 4,0 vezes. Isso também ocorre para os *solvers* GS (entre 2,1 e 2,9 vezes) e ADI (entre 1,7 e 2,6 vezes). Por exemplo, para a malha  $2049 \times 2049$  nós, o tempo de CPU do MG-MSI-FAS e MG-MSI-CS é de 124,7 s e 494,3 s, respectivamente, ou seja, o MG-MSI-FAS é cerca de 4,0 vezes mais rápido do que o MG-MSI-CS.

Portanto, recomenda-se o uso do esquema FAS com o *solver* MSI para a equação de Laplace bidimensional. Esta constatação é inédita e inesperada pois o esquema CS é indicado para resolver equações lineares, caso deste trabalho, e o esquema FAS, para equações não-lineares. Aqui vemos que o esquema CS perde a hegemonia perante o esquema FAS, mesmo que para um problema linear.

Os resultados do presente trabalho diferem daqueles comentados em Brandt (1977). Ele fez análises teóricas e experimentais (numéricas) entre as razões de engrossamento  $r = 2, 3$  e  $3/2$  para diversos problemas, mas não se referiu à Eq. (5) exatamente. Brandt mostra preferência pelo esquema CS em relação ao esquema FAS no caso de problemas lineares. Segundo Brandt, cada ciclo iterativo do esquema FAS é mais caro computacionalmente se comparado ao esquema CS devido à transferência de informações a respeito do resíduo e da solução para as malhas mais grossas.

Seja o problema de minimização do tempo de CPU em função do uso dos diferentes esquemas CS e FAS. Para este problema, constatou-se que o esquema ótimo tem uma forte influência da qualidade das informações transferidas pela restrição e pela prolongação e, principalmente, pela transferência de informações a respeito do resíduo e da solução para as malhas mais grossas. Para o esquema FAS, por exemplo, a estimativa inicial para as malhas mais grossas é sempre a restrição da solução suavizada da malha imediatamente mais fina, enquanto que para o esquema CS, a estimativa inicial é sempre a estimativa inicial nula. Ou seja, o esquema FAS tem sempre estimativas iniciais melhores para os problemas

nas malhas grossas. Com isto, constatou-se no presente trabalho que o número de ciclos ou iterações do esquema FAS é significativamente menor do que o esquema CS. Por exemplo, para a malha 2049x2049 nós, o número de ciclos ou iterações do MG-MSI-FAS é de apenas 3 contra 22 do MG-MSI-CS.

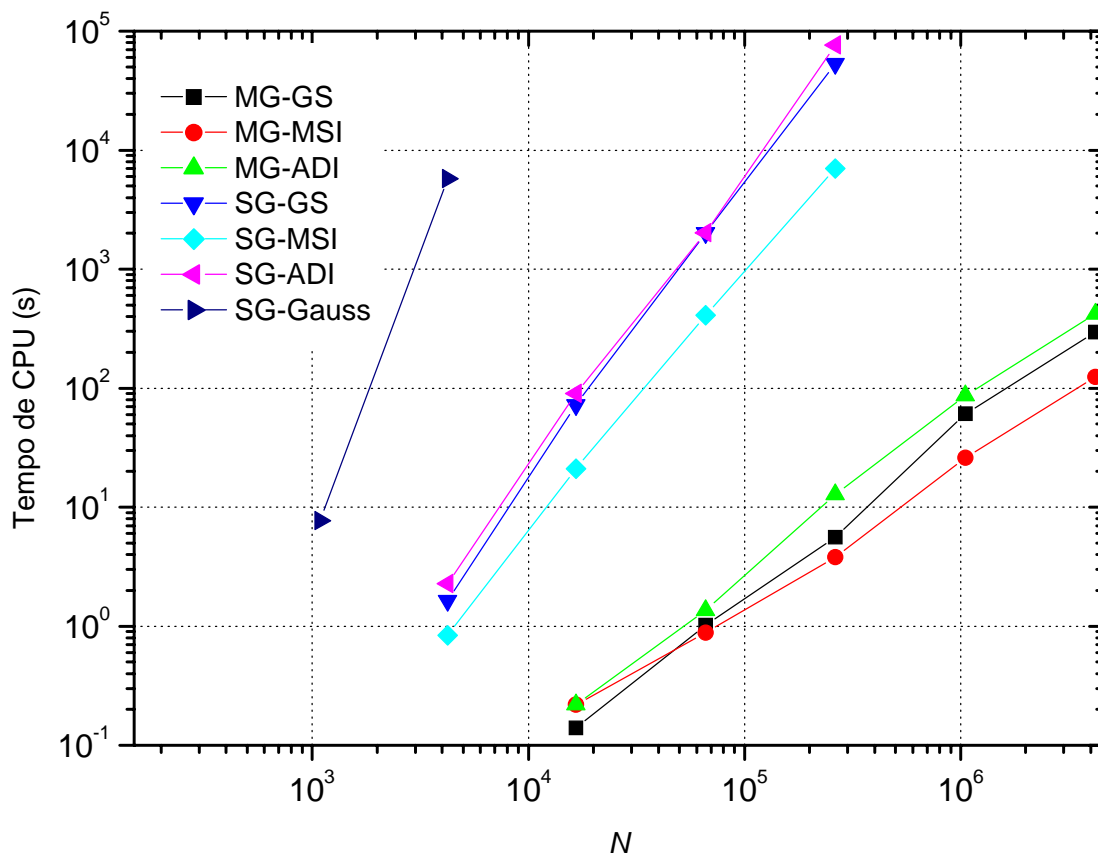


Figura 4. Tempo de CPU versus número de incógnitas ( $N$ ) com esquema FAS.

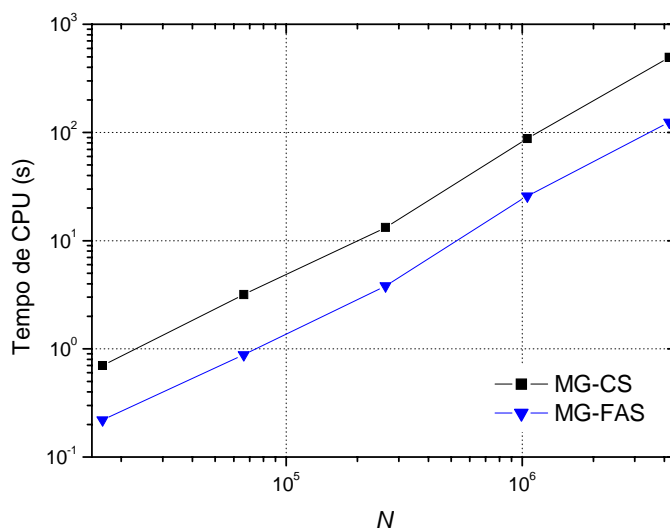


Figura 5. Tempo de CPU versus número de incógnitas ( $N$ ) para o método *multigrid* (MG) com o *solver* MSI e os esquemas CS e FAS.

## 5. Conclusão

Neste trabalho verificou-se o efeito de diversos parâmetros sobre o tempo de CPU necessário para resolver um problema com o método *multigrid* (MG) geométrico. Os parâmetros considerados foram: número de nós ( $N$ ), número de iterações internas ( $ITI$ ), número de níveis de malhas ( $L$ ), *solvers* (GS, MSI e ADI), e esquemas de correção (CS) e de aproximação completa (FAS). O modelo matemático considerado é um problema bidimensional linear, governado pela equação de Laplace com condições de contorno de Dirichlet. Esta equação foi discretizada com o método de diferenças finitas.

Com base nos resultados deste trabalho, verificou-se que:

- 1) O esquema FAS é mais rápido do que o CS.
- 2) Para os esquemas CS e FAS, o *solver* MSI é mais rápido do que o GS e o ADI. Portanto, mesmo com o método *multigrid*, quanto mais fortemente implícito é o *solver*, mais rápido ele é.
- 3) O esquema utilizado influencia o número ótimo de iterações internas: para o esquema CS,  $ITI_{ótimo} = 1$  com qualquer *solver*; e, para o esquema FAS,  $ITI_{ótimo} = 3$  a 5, dependendo do *solver*.
- 4) O esquema utilizado não tem grande influência sobre o número ótimo de níveis de malha:  $L_{ótimo} \approx L_{máximo}$  para os esquemas CS e FAS com os *solvers* GS, MSI e ADI.

Segundo o conhecimento dos autores, estas constatações são inéditas na literatura disponível.

## 6. Agradecimentos

O primeiro autor agradece o apoio do Laboratório de Experimentação Numérica (LENA), do Departamento de Engenharia Mecânica da UFPR, por disponibilizar sua infra-estrutura, da Universidade Estadual de Ponta Grossa e da CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) pelo suporte financeiro e dos amigos do LENA. O segundo autor é bolsista do CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico).

## 7. Referências

- Brandt, A., "Multi-Level adaptive solutions to boundary-value problems, *Mathematics of Computation*", vol. 31, pp. 333-390, 1977.
- Briggs, W. L. and Henson, V.E., McCormick, S.F., "A Multigrid Tutorial", 2ª ed., SIAM, 2000.
- Burden, R. L. and Faires, J. D., "Numerical Analysis", 6ª ed., Brooks/Cole Publishing Company, 1997.
- Dennis, J. E. and Schnabel, R., "Numerical Methods for Unconstrained Optimization and Nonlinear Equations". Prentice Hall, 1983.
- Fedorenko, R. P., "On the Speed of Convergence of an Iteration Process", *USSR Comput. Math. And Math. Phys.*, vol. 4 (3), 1964.
- Ferziger, J. H. and Peric, M., "Computational Methods for Fluid Dynamics", 2ª ed., Springer, 1999.
- Fortuna, A. O., "Técnicas Computacionais para Dinâmica dos Fluidos", Edusp, 2000.
- Golub, G. H. and Ortega, J. M., "Scientific Computing and Differential Equations: An Introduction to Numerical Methods", Academic Press, Inc., 1992.
- Golub, G. H. and Van Loan, C., "Matrix Computations", 2ª ed., Johns Hopkins Press, Baltimore, 1989.
- Hestenes, M. and Stiefel, E., "Methods of Conjugate Gradients for Solving Linear Systems", *Journal of Research of the National Bureau of Standards*, vol. 49, pp. 409-436, 1952.
- Hirsch, C., "Numerical Computational of Internal and External Flows", vol.1, Wiley, 1988.
- Maliska, C. R., "Transferência de calor e mecânica dos fluidos computacional", LTC, 2ª ed., 2004.
- Moro Filho, R. C., "Aplicação da Técnica Multigrid em Transferência de Calor Computacional", XXV Iberian Latin American Congress on Computational Methods in Engineering, 2004.
- Pinto, M. A. V.; Santiago, C. D.; Marchi, C. H., "Effect of Parameters of a Multigrid Method on the CPU Time for One-dimensional Problems", *Proceedings of the International Congress of Mechanical Engineering (COBEM)*, Paper 0619, 2005a.
- Pinto, M. A. V.; Santiago, C. D.; Marchi, C. H., "Efeito de Parâmetros do Método Multigrid sobre o Tempo de CPU para a Equação de Burgers Unidimensional", *Proceedings of the Iberian Latin-American Congress on Computational Methods in Engineering (CILAMCE)*, Paper Cil 05-0098, 2005b.
- Roache, P. J., "Fundamentals of Computational Fluid Dynamics", Hermosa Publishers, 1998.
- Stüben, K., "Algebraic Multigrid (AMG): an introduction with applications", in: GMD-Report 70, November 1999.
- Stüben, K., "A Review of Algebraic Multigrid", *Journal of Computation and Applied Mathematics*, vol. 128, pp. 281-309, 2001.
- Tannehill, J. C., Anderson, D. A. and Pletcher, R. H., "Computational Fluid Mechanics and Heat Transfer", 2ª ed., Washington: Taylor & Francis, 1997.
- Wesseling, P., "An Introduction to *Multigrid* Methods", John Wiley & Sons, 1992.
- Wesseling, P. and Oosterlee, C. W., "Geometric Multigrid with Applications to Computational Fluid Dynamics", *Journal of Computation and Applied Mathematics*, vol. 128, pp. 311-334, 2001.

**EFFECT OF CS AND FAS MULTIGRID PARAMETERS ON THE CPU TIME FOR TWO-DIMENSIONAL LAPLACE'S EQUATION**

Márcio Augusto Villela Pinto

Department of Mathematics and Statistics, State University of Ponta Grossa, Ponta Grossa, PR, Brazil  
marciovp@demec.ufpr.br

Carlos Henrique Marchi

Department of Mechanical Engineering, Federal University of Paraná, Curitiba, PR, Brazil  
marchi@demec.ufpr.br

**Abstract**

On the necessary CPU time to solve a problem, one verifies the effect considered by: number of nodes, number of inner iterations, number of grid levels, solvers (GS, MSI and ADI) and Correction (CS) and Full Approximation Schemes (FAS). The considered problem involves a two-dimensional linear problem: Laplace's equation with Dirichlet boundary conditions. The finite difference method is used to discretize the differential equation. The algebraic linear systems are solved with the three solvers associated to geometric multigrid with V-cycle. Some literature results are confirmed and some new ones are presented. The main conclusions of this work are that FAS is faster than CS and MSI solver is faster than GS and ADI solvers for both CS and FAS schemes.

Keywords: solver, CFD, finite difference, heat transfer, numerical methods.

## NUMERICAL SOLUTION OF ONE-DIMENSIONAL REACTIVE FLOWS IN ROCKET ENGINES WITH REGENERATIVE COOLING

**Luciano Kiyoshi Araki**

Programa de Pós-graduação em Métodos Numéricos em Engenharia, UFPR, Curitiba, PR, Brazil  
lucaraki@demec.ufpr.br

**Carlos Henrique Marchi**

Departamento de Engenharia Mecânica, Universidade Federal do Paraná, Curitiba, PR, Brazil  
marchi@demec.ufpr.br

**Abstract.** *The maximum temperature in the internal structure and the pressure drop on the coolant flow along the channels are two of the most important design parameters for rocket engines with regenerative cooling. A one-dimensional mathematical model for the reactive flows in a LOX/LH<sub>2</sub>-rocket engine with regenerative cooling is presented. This problem includes the combustion gases flow in the rocket diverging-converging nozzle, the coolant flow in channels and the heat transfer through the walls (from combustion gases to coolant). For the combustion gases flow, three different physical flow models are used: a frozen, an equilibrium and a non-equilibrium one. Several chemical reaction schemes are also taken into account for each study, for the evaluation of effects of the species and reaction equation systems on the final numerical solution. Besides the influence of physical and chemical models on the answers, the effects of the adopted grids over the final results are also discussed. Therefore numerical errors estimative are also made and the physical and chemical models results are compared. Numerical results are obtained with a finite volume software, implemented with second order scheme for variables, co-located grid arrangement for all speed flows and Fortran 95 language.*

**Keywords.** *Finite volume method, CFD, equilibrium flow, non-equilibrium flow, numerical error estimate.*

### 1. Introduction

The thrust chamber is the key subassembly of a rocket engine, where the liquid propellants are metered, injected, atomized, vaporized, mixed and burnt, to form hot reaction gas products (Sutton and Biblarz, 2001). By speeding up these gases, thrust force is obtained for orbit and payload definition. The knowledge of chemical and physical phenomena concerned about such gases flow is, therefore, important for both rocket design (associated to material definition and cooling system design) and mission goals achievement.

Heat is transmitted to all internal hardware surfaces exposed to hot gases, namely the injector face, the chamber and nozzle walls. Only 0.5 to 5% of the total energy generated in the gas is transferred as heat to the chamber walls (Sutton and Biblarz, 2001). This amount of energy, however, can increase the wall temperatures until material failure. Because of this, most of rocket engines have cooling systems, which allow a longer lifetime for the entire device. One of the most used cooling systems for large rocket engines is the regenerative one. The prediction of heat transfer characteristics in a regeneratively cooled rocket combustion engine is one of the most important and most challenging tasks in the design work of a high performance engine (Fröhlich *et al.*, 1993).

The problem involving reaction gas products flow and heat transfer to cooling system may be divided up into three sub-problems, namely: the reactive combustion gases flow through the thrust chamber (which includes the combustion chamber and the nozzle engine); the heat conduction from hot gases to the coolant, through the wall structure; and the turbulent coolant flow, in the regenerative cooling system. The solution for the whole problem can be obtained by solving, iteratively and in sequence, the three sub-problems, allowing them to interact, until its convergence.

Although two and three-dimensional models are commonly used, one-dimensional models are still employed in rocket engines projects, being corrected by empirical coefficients (Fröhlich *et al.*, 1993; Sutton and Biblarz, 2001). Studies involving one-dimensional softwares are, because of this, still useful. Based on this, in this work an one-dimensional model, which includes different physical and chemical schemes, is purposed for solving the coupled problem of reaction gas products flow and heat transfer to the thrust chamber structure and coolant.

Four physical models (one-species with variable properties, frozen, local equilibrium and non-equilibrium flows) are studied. Both frozen and local equilibrium flows are ideal models; they differ, however, about how the chemical reactions take place during the flow. For both frozen and local equilibrium physical models, nine different chemical reaction schemes are used. These schemes are the same presented before by Marchi *et al.* (2005), and include from three to eight chemical species and take into account from null to eighteen chemical reaction equations. For non-equilibrium model, five different chemical schemes are used, including six or eight chemical species, which totalise five chemical schemes. At this work, however, only two chemical models are studied (one six-species and one eight-species for frozen and equilibrium flows and two six-species for non-equilibrium flow). According to Marchi *et al.* (2005) and Araki and Marchi (2006), six and eight-species models present good accuracy with CEA results (Glenn Research



Center/NASA, 2005), for frozen and equilibrium flows with adiabatic walls, reason for considering only six and eight-species models in this work.

All the models have been implemented using Compaq Visual Fortran 6.6.0. The implemented software, called RHG1D 3.0, was run at a PC Pentium IV, 3.4 GHz, with 4 GB RAM. Results are compared to those ones obtained for adiabatic walls simulations and numerical error estimates, related to all the results and based on GCI estimative (Roache, 1994), are also presented.

## 2. Mathematical model

The basic principles of the rocket propulsion are essentially those of mechanics, thermodynamics and chemistry (Sutton and Biblarz, 2001). The flow and heat transfer problem in rocket thrust chambers can be divided up into three different (but coupled) problems, for which there are independent mathematical (and numerical) models: (1) the reaction gas products (combustion gases mixture) flow through the thrust chamber; (2) the heat conduction from hot gases to the coolant; and (3) the coolant flow through the regenerative cooling system. Figure 1 shows, schematically, both the reaction gas products and the coolant flows.

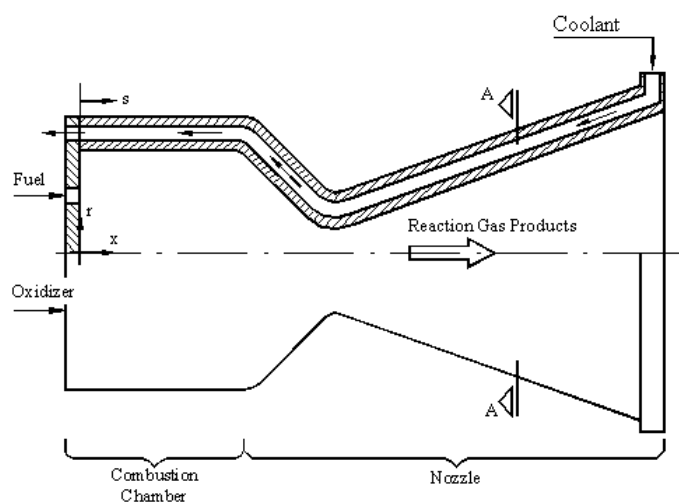


Figure 1. Thrust chamber engine with regenerative cooling system (transversal section).

### 2.1. Reaction gas products flow

The mathematical formulation for a one-dimensional, single-species flow (or a multi-species frozen or equilibrium flow) through the nozzle engine is based only on four equations: conservation of mass, conservation of momentum, conservation of energy and state (the one-species gas and/or the multi-species gas are treated by the perfect gases law), given in the sequence:

$$\frac{d}{dx}(\rho u S) = 0, \quad (1)$$

$$\frac{d}{dx}(\rho u S u) = -S \frac{dP}{dx} + F', \quad (2)$$

$$c_p \frac{d}{dx}(\rho u S T) = u S \frac{dP}{dx} + q' + S_{eq/ne}, \quad (3)$$

$$P = \rho R T, \quad (4)$$

where:  $\rho$ ,  $u$ ,  $P$  and  $T$  are four dependent variables, related to density, velocity, pressure and temperature (in this order);  $x$  is the axial coordinate along the gases flow (Fig. 1);  $S$  is the internal cross-sectional area of thrust chamber structure;  $R$  is the gas constant (or gas mixture constant, for frozen and equilibrium flows);  $c_p$  is the constant-pressure specific heat;  $F'$ ,  $q'$  and  $S_{eq/ne}$  are related to the viscous forces, the heat loss to the walls and the source term correspondent to equilibrium or non-equilibrium conditions (this term is necessary only for equilibrium or non-equilibrium flows), respectively, and are evaluated by the following relations:

$$F' = -\frac{\pi}{8} f \rho u |u| D, \quad (5)$$

$$q' = |u F'| + S'_{wall} (q''_h + q''_r), \quad (6)$$

$$S'_{eq/ne} = \begin{cases} -\sum_{i=1}^N h_i \frac{d}{dx} (\rho u S Y_i), & \text{for local equilibrium flow;} \\ -S \sum_{i=1}^N h_i \dot{w}_i, & \text{for non-equilibrium flow;} \end{cases} \quad (7)$$

where:  $f$  is the Darcy friction factor;  $D$  is the diameter of the cross-section;  $S'_{wall}$  is the internal wall surface by length unity (along  $x$ -axis, Fig. 2);  $N$  is the total number of species existent in the flow;  $h_i$  is the enthalpy for each chemical species  $i$ ;  $Y_i$  is the mass fraction for each chemical species  $i$ ;  $\dot{w}_i$  corresponds to mass rate generation of species  $i$ ; and  $q''_h$  and  $q''_r$  are the convection and radiation heat fluxes to internal walls.

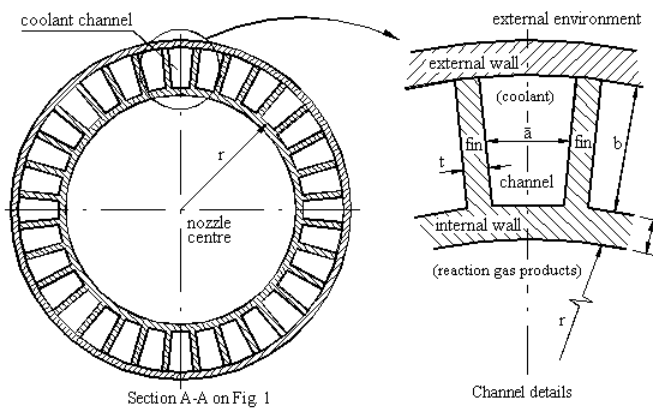


Figure 2. Cooling channels and their geometrical parameters.

As can be seen previously, the conservation of energy, Eq.(3), has the temperature as unknown and not the enthalpy nor the internal energy, as normally used. The major advantage of such variation is about the temperature determination, which can be obtained directly from the numerical model, not depending on the enthalpy (or internal energy) values. Besides the gas products flow, chemical reaction schemes have an important role on final results. Frozen, equilibrium and non-equilibrium models, in different levels, take into account chemical reactions. Depending on the adopted chemical reaction scheme, different chemical composition and physical properties are obtained or, at least, larger or smaller CPU time efforts are verified. For the frozen model, by hypothesis, the flow from combustion chamber to thrust chamber exit is so fast that there is no time for changes in chemical composition of reaction gas products. For the local equilibrium model, in counterpart, chemical equilibrium composition must be evaluated for every single cross-section of the thrust chamber. An intermediate (and more realistic) model between local equilibrium and frozen ones is the non-equilibrium. For this model, chemical composition must be evaluated for every single cross-section of the thrust chamber as done for local equilibrium; the equilibrium composition, however, is not reached and, for each single chemical species, mass formation rates are evaluated.

For frozen and equilibrium models, nine different chemical reaction schemes were implemented, taking into account from 3 to 8 chemical species and from null to eighteen chemical reaction equations, as can be seen at Table 1 where  $N$  is the number of chemical species and  $L$  is the number of chemical reaction equations. Chemical reaction schemes implemented in software Mach1D 5.0, for both frozen and local equilibrium flows, are the same presented by Marchi *et al.* (2005). For non-equilibrium flow, only six and eight-species models were considered, totalising five. For both frozen and equilibrium flows, different chemical schemes mean at least one different chemical reaction equation (even when the same number of chemical species and the same number of chemical reaction equations are considered). For non-equilibrium, otherwise, different chemical schemes mean, at least, different forward reaction constant: based on this, model 3 could be split up into two ones (the other non-equilibrium models are 5, 7 and 10).

At non-equilibrium flow model, forward and backward reaction constants ( $kf_j$  and  $kb_j$ , respectively) have an important role for mass generation rates evaluation, being related to each other by the progression rate of reaction  $j$  ( $\gamma$ ):

$$\gamma_j = \left\{ kf_j \prod_{i=1}^N C_i^{\nu_{ij}'} - kb_j \prod_{i=1}^N C_i^{\nu_{ij}''} \right\}, \quad (8)$$

where  $C_i$  is the molar concentration of species  $i$  in gases mixture and  $\nu_{ij}'$  and  $\nu_{ij}''$  are, in this order, the stoichiometric coefficients of chemical species  $i$  in reaction  $j$  existent in reagents and products. The species generation rate ( $\theta_j$ ) is obtained from the product between Eq. (8) and the third body effective concentration and can be estimated by:

$$\theta_j = \sum_{i=1}^N \alpha_{ij} \cdot C_i \cdot \gamma_j, \quad (9)$$

in which  $\alpha_{ij}$  is the efficiency of chemical species  $i$  at a dissociation reaction  $j$ . Unfortunately, there is a lack of information about the efficiency of the molecules existent on a dissociation reaction model. Because of this, it is commonly adopted that the efficiencies of all chemical species are equal to unit and, in this case, the concentration of third body species corresponds to the whole number of existent species (Barros *et al.*, 1990).

Table 1. Chemical reaction schemes implemented in software Mach1D 5.0.

Model	L	N	Species	Observations
0	0	3	H <sub>2</sub> O, O <sub>2</sub> , H <sub>2</sub>	Ideal model
1	1	3	H <sub>2</sub> O, O <sub>2</sub> , H <sub>2</sub>	–
2	2	4	H <sub>2</sub> O, O <sub>2</sub> , H <sub>2</sub> , OH	–
3	4	6	H <sub>2</sub> O, O <sub>2</sub> , H <sub>2</sub> , OH, O, H	4 reactions with 3 <sup>rd</sup> body – Barros <i>et al.</i> (1990) and Smith <i>et al.</i> (1987)
4	4	6	H <sub>2</sub> O, O <sub>2</sub> , H <sub>2</sub> , OH, O, H	4 reactions – Svehla (1964)
5	8	6	H <sub>2</sub> O, O <sub>2</sub> , H <sub>2</sub> , OH, O, H	8 reactions (4 with 3 <sup>rd</sup> body) – Barros <i>et al.</i> (1990)
7	8	6	H <sub>2</sub> O, O <sub>2</sub> , H <sub>2</sub> , OH, O, H	8 reactions (4 with 3 <sup>rd</sup> body) – Smith <i>et al.</i> (1987)
10	6	8	H <sub>2</sub> O, O <sub>2</sub> , H <sub>2</sub> , OH, O, H, HO <sub>2</sub> , H <sub>2</sub> O <sub>2</sub>	4 reactions from model 3 and 2 from Kee <i>et al.</i> (1990) – all the reactions including 3 <sup>rd</sup> body
9	18	8	H <sub>2</sub> O, O <sub>2</sub> , H <sub>2</sub> , OH, O, H, HO <sub>2</sub> , H <sub>2</sub> O <sub>2</sub>	18 reactions (5 with 3 <sup>rd</sup> body) – Kee <i>et al.</i> (1990)

The determination of mass generation rates is obtained by the product of Eq. (9) and molecular weight ( $M_i$ ) of the species, resulting in

$$\dot{w}_i = M_i \sum_{j=1}^L (\Delta \nu_{ij} \cdot \theta_j), \quad (10)$$

where  $\Delta \nu_{ij}$  is calculated by  $\Delta \nu_{ij} = \nu_{ij}'' - \nu_{ij}'$  and represents the difference between formed and consumed number of moles for a reaction  $j$ . For non-equilibrium model, mass generation rates determination is fundamental for the species conservation of mass. Equations (1) to (4) are enough for one-species, frozen and equilibrium flows mathematical formulation; for non-equilibrium flow, however, associated to these four equations must be taken into account the species continuity equation:

$$\frac{d}{dx} (\rho u S Y_i) = S \dot{w}_i, \quad (11)$$

which corresponds to the conservation of mass for each species separately.

## 2.2. Coolant flow

Only four equations are needed for the one-dimensional study of coolant flow through the entire regenerative cooling system of a thrust chamber: the mass conservation equation, the momentum conservation equation, the energy conservation equation and a polynomial constitutive relation for density:

$$\frac{d}{ds} (\rho_c u_c S_c) = 0, \quad (12)$$

$$\frac{d}{ds} (\rho_c u_c S_c u_c) = -S_c \frac{dP_c}{ds} + F_c', \quad (13)$$

$$c_{pc} \frac{d}{ds} (\rho_c u_c S_c T_c) = \beta T_c u_c S_c \frac{dP_c}{ds} + q'_c, \quad (14)$$

$$\rho_c = \rho_1 + \rho_2 T_c + \rho_3 T_c^2, \quad (15)$$

where: the index  $c$  corresponds to coolant properties;  $\rho$ ,  $u$ ,  $P$  and  $T$  are four dependent variables, related to density, velocity, pressure and temperature (respectively);  $s$  is the length of the flow along the center of a channel (Fig. 1);  $S_c$  is the internal cross-sectional area of coolant channels;  $F'_c$ ,  $\beta$  and  $q'_c$  are related to the viscous forces, volumetric thermal expansion coefficient and the heat loss to the coolant channel walls, in this order; and  $\rho_1$ ,  $\rho_2$  and  $\rho_3$  are constants related to the chosen coolant.

### 2.3. Heat conduction through the wall

As reaction gas products have high temperature values, heat is transferred from them to thrust chamber walls by convection and radiation mechanisms. This energy is then conducted through walls to the coolant and transmitted to it by convection. The whole process can be modelled by:

$$q = (q''_h + q''_r) S_{wh} = q''_w S_{wh} = q''_c S_{wc}, \quad (16)$$

where:  $q$  is the heat transfer rate through the wall;  $S_{wh}$  is the internal wall area in contact with reaction gas products (Fig. 2);  $q''_w$  is the heat flux through the wall;  $q''_c$  is the heat flux to the coolant; and  $S_{wc}$  is the effective heat transfer area between the wall and the coolant.

More details about the employed mathematical model can be found in Marchi *et al.* (2000; 2004).

### 3. Numerical model

The mathematical models for reaction gas products and coolant flows (assembly to the heat conduction through the wall) in the thrust chamber structure are discretized using finite volume method. The domain is divided into  $N_{vol}$  control volumes, in axial directions  $x$  and  $s$ , in which the differential equations are integrated. For reaction gas products, a co-located grid arrangement, appropriated for all speed flows is used (Marchi and Maliska, 1994), associated with a second-order discretization scheme (CDS), with deferred correction (Ferziger and Perić, 2001); similar treatment is given to coolant flow model, except for the appliance for all speed flows, once the coolant flow is always subsonic. The systems of algebraic equations obtained are solved by TDMA method (Ferziger and Perić, 2001).

Pressure and velocity are coupled by SIMPLEC algorithm (Van Doormaal and Raithby, 1984), in order to convert the mass equation in a pressure (or better, in a pressure-correction) one. So, the mass conservation equation, Eq. (1) or Eq. (12), is used for determination of a pressure-correction ( $P'$ ), while velocity ( $u$ ) is obtained from the momentum equation, Eq. (2) or Eq. (13), and the energy equation, Eq. (3) or Eq. (14), is taken for temperature ( $T$ ) determination. Density ( $\rho$ ) is gotten from the state equation, Eq. (4), or from Eq. (15). Equation (11) is also needed for non-equilibrium flow. It must be noted that velocity ( $u$ ) is evaluated from very reduced values (near null values) until supersonic ones, and not only for supersonic values, as commonly found in literature (Barros *et al.*, 1990).

The iterative process for solving the reaction gas products mathematical model provided by Eqs. (1) to (4) or Eqs. (12) to (15) consists only on nine steps:

1. Definition of data (temperature, pressure, density, velocity) in an instant of time  $t$ .
2. Estimation of all variables in an instant  $t+\Delta t$  (time is used as a relaxation parameter).
3. Definition of thermophysical properties (such as the constant-pressure specific heat, Darcy friction factor and heat transfer coefficients by convection).
4. Coefficients calculation for the algebraic system (by discretization) of the momentum equation and solution of this system by TDMA for the velocity  $u$ .
5. Coefficients calculation for the algebraic system (by discretization) of the energy equation and solution by TDMA for temperature  $T$ .
6. Calculation of density  $\rho$ .
7. Coefficients calculation for the algebraic system (by discretization) of the mass equation and solution by TDMA for pressure correction  $P'$ .
8. Correction of  $\rho$ ,  $P$  and  $u$  with  $P'$ .
9. Return to item 2, until the achievement of the desired number of iterations.

Over this basic algorithm, some modifications are necessary for reaction gas products flow, depending on the adopted physical model. For equilibrium and non-equilibrium models, step 3 includes also the chemical composition determination for all the domain control volumes and, also here, the determination of thermoproperties must be done for every single control volume. Non-equilibrium chemical composition determination includes, also, the use of Eq. (10), for mass generation rate evaluation, and Eq. (11), in order to estimate the chemical composition, respecting the mass conservation law for each species separately.

The adopted boundary conditions to reaction gas products flow include: definition of inlet temperature ( $T$ ) and pressure ( $P$ ) as functions of stagnation parameters ( $T_0$  and  $P_0$ , in this order); the chemical mixture composition, given by mass fractions ( $Y_i$ ), is obtained from local data (temperature and pressure); and the entrance velocity ( $u$ ) is obtained from a linear extrapolation from the values obtained for internal flow. About exit conditions, for supersonic flows in nozzles, no exit boundary conditions are required; for the implementation of a numerical model, however, exit boundary conditions are needed. Because of this, temperature ( $T$ ), velocity ( $u$ ), pressure ( $P$ ) and mass fractions ( $Y_i$ ) are obtained by linear extrapolation from internal control volumes.

For coolant flow, some other boundary conditions are employed, as follows: for channels entrance, both temperature and velocity are fixed ( $T_{in}$  and  $u_{in}$ , in this order) and inlet pressure is obtained from linear extrapolation of values obtained for internal flow; for density, both inlet and outlet values are estimated by Eq. (15); exit pressure is defined as null and both, exit temperature and exit velocity, are obtained from linear extrapolation of values obtained for internal flow.

The coupling between reaction gas products and coolant flows is made by the heat conduction through the thrust chamber wall, according to the procedures presented by Marchi *et al.* (2004), which consists on seven steps:

1. First estimative of wall temperatures distribution, for the wall in contact with the reaction gas products.
2. Solution of reaction gas products flow.
3. Solution of coolant flow.
4. Evaluation of heat transfer rate between the reaction gas products and the coolant.
5. Evaluation of both wall temperatures: the temperature of the wall in contact with the reaction gas products and the wall in contact with the coolant.
6. Evaluation of heat transfer rate error, which consists on the difference between the reaction gas products-to-wall heat transfer rate and the wall-to-coolant heat transfer.
7. Return to item 2, until the achievement of the desired tolerance or the desired number of iterations.

#### 4. Definition of the problem

The thrust chamber geometry used in this work is the same one presented by Marchi *et al.* (2000; 2004), which consists on a cylindrical section, called combustion chamber (with radius  $r_{in}$  and length  $L_c$ ) assembled to a nozzle device, whose longitudinal section is defined by a cosine curve (with throat nozzle radius  $r_g$  and length  $L_n$ ). The radius  $r$ , for  $x > L_c$ , is evaluated by the following equation:

$$r = r_g + \frac{(r_{in} - r_g)}{2} \left\{ 1 + \cos \left[ 2\pi \frac{(x - L_c)}{L_n} \right] \right\}, \quad (17)$$

where  $x$  corresponds to the position where the radius is evaluated. Despite the cylindrical section is called combustion chamber, it does not correspond to a real combustion chamber; the effects of fuel and oxidant injection, mixture and burning are not considered. Figure 3 shows all geometrical parameters of the thrust chamber studied in this work.

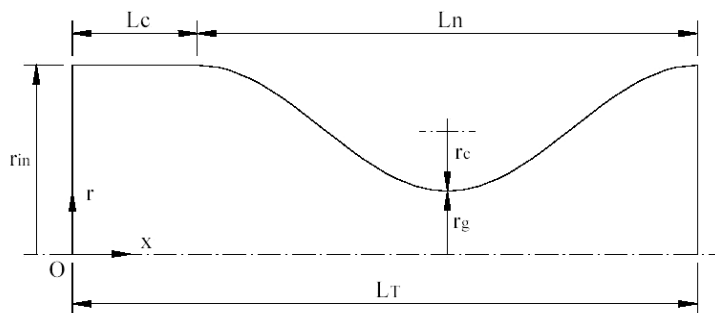


Figure 3. Geometrical parameters of the thrust chamber engine.

About the cooling system, the number of channels ( $m$ ) placed around thrust chamber structure. The parameters, which define the geometrical the channels (Fig. 2) are: the thickness ( $e$ ) of thrust chamber internal wall; the thickness

( $t$ ) of wall that separates two neighbour cooling channels; and the channels height ( $b$ ). The average width of each channel ( $\bar{a}$ ) is obtained as a function of previous parameters by

$$\bar{a} = \frac{\pi}{mb} [(r+e+b)^2 - (r+e)^2] - t, \quad (18)$$

when  $m > 1$ . While the parameters  $b$ ,  $e$  and  $t$  remain constant,  $\bar{a}$  does not have the same behaviour, due to radius variations.

Copper-made channels cover all the length of the thrust chamber engine, accompanying the variable radius of the engine. And, while the reaction gas products flows along the  $x$ -axis in a positive direction, the coolant flows in counter flow, namely, in  $s$ -axis negative direction. For all the studies presented in this work, the values used for geometrical parameters are:  $r_g = 0.1$  m;  $r_{in} = 0.3$  m;  $L_C = 0.1$  m;  $L_n = 0.4$  m;  $m = 200$ ;  $b = 5.0$  mm;  $e = 2.0$  mm; and  $t = 1.5$  mm. Based on these data, the ratio between the height and the average width of each channel ( $b/\bar{a}$ ) varies between 0.62 and 2.8; the engine internal wall area is  $\approx 9.242 \times 10^{-1}$  m<sup>2</sup>; the base channel area in contact with the coolant is  $\approx 7.272 \times 10^{-1}$  m<sup>2</sup> and the fins surface in contact with the coolant is  $\approx 1.371$  m<sup>2</sup>. Water is taken as coolant and its mass flow rate is 1 kg/s in each channel, totalising 200 kg/s, with an inlet temperature of 300 K. Inside the combustion chamber, the stagnation temperature ( $T_0$ ) is taken as 3420.33 K, while the stagnation pressure value ( $P_0$ ) is 20 MPa and liquid oxygen and liquid hydrogen are injected at the stoichiometric composition ( $OF = 7.936682739$ ). For isentropic analytical solution, the gas constant ( $R$ ) is taken as 526.97 J/kg·K and the ratio between specific heats ( $\gamma$ ) is 1.1956.

The aim of numerical solutions is given by the evaluation of some parameters of interest, cited in the following:

1. Nozzle discharge coefficient ( $C_d$ ): defined as the ratio between numerical mass flux ( $\dot{m}_{num}$ ) and the theoretical one ( $\dot{m}_{th}$ ), obtained from isentropic analytical solution:

$$C_d = \frac{\dot{m}_{num}}{\dot{m}_{th}}. \quad (19)$$

2. Non-dimensional momentum thrust ( $F^*$ ): defined by the ratio between numerical and theoretical thrusts ( $F_{num}$  and  $F_{th}$ , in this order), given by

$$F^* = \frac{F_{num}}{F_{th}}, \quad (20)$$

where  $F$  corresponds to the thrust values and can be obtained by the following relation between mass flux ( $\dot{m}$ ) and exit velocity ( $u_{ex}$ ):

$$F = \dot{m} \cdot u_{ex}. \quad (21)$$

3. Pressure ( $P_{ex}$ ), temperature ( $T_{ex}$ ), velocity ( $u_{ex}$ ) and Mach's number ( $M_{ex}$ ) for gas products, at the nozzle exit.
4. Maximum wall temperature ( $T_{max}$ ): obtained by wall temperature distribution.
5. Coolant pressure drop ( $\Delta P$ ): evaluated from channels entrance to its exit.
6. Exit coolant temperature ( $T_{cool}$ ).

## 5. Numerical results and discussion

Four different physical models, including heat transfer effects, are studied in this work: a one-species gas with variable properties; a frozen gases mixture flow; an equilibrium gases mixture flow; and non-equilibrium gases mixture flow. All the analyses results were obtained employing a PC, with Pentium IV 3.4 GHz processor, 4 GB RAM, Windows XP. The results are then compared with those ones obtained by the same physical models (provided by both Mach1D 5.0 and CEA, this last one a numerical software provided by NASA), but without coolant flow effects (adiabatic walls).

For all the studies, the simulations were made for 3 different grids, to allow the determination of apparent and effective convergence orders (Marchi and Silva, 2002). Numerical error estimates, also, based on GCI estimator (Roache, 1994), were taken for all the physical and chemical models. Some pieces of information about numerical errors and error estimators can be found in Tannehill et al. (1997), Marchi (2001), and Ferziger and Perić (2001). Error estimates are very important for evaluating if two different physical (or chemical) models have the same results or the differences can be assigned to numerical errors. The GCI estimator is evaluated by

$$GCI(\varphi_1, p) = 3 \frac{|\varphi_1 - \varphi_2|}{(q^p - 1)}, \quad (22)$$

where:  $\varphi_1$  and  $\varphi_2$  are, respectively, the numerical solutions for the refined ( $h_1$ ) and coarse ( $h_2$ ) grids;  $q$  is the grid refinement ratio ( $q = h_2 / h_1$ );  $h$  is the grid spacing or distance between two successive grid points; and  $p$  is related to the asymptotic ( $p_L$ ) or apparent ( $p_U$ ) order (the lowest value between the two ones). Asymptotic error order depends on the chosen discretization model, while apparent error, for constant refinement ratio, is evaluated by

$$p_U(h_1) = \frac{\log\left(\frac{\varphi_2 - \varphi_3}{\varphi_1 - \varphi_2}\right)}{\log(q)}, \quad (23)$$

where  $\varphi_3$  corresponds to the numerical solution for a supercoarse grid.

Although nine chemical schemes are available at RGH1D 3.0 software for frozen and equilibrium flows and another six schemes are implemented for non-equilibrium flow, only models 3 and 10 (with six and eight species, respectively) for frozen and equilibrium flows and models 31 and 32 (both with six species) for non-equilibrium flow are studied in this work. According to results obtained by Marchi *et al.* (2005) and Araki and Marchi (2006), six and eight-species models present good accuracy with CEA results; because of this, only six and eight-species models results are reported in this work. Tables 2, 3 and 4 present the results for the variables of interest, including numerical error estimates, for an 80-volumes grid. The reason for using such a grid is based on other studies (Marchi *et al.*, 2004; Araki and Marchi, 2006), for which numerical error estimates are about the same magnitude of experimental error; besides, CPU time consumption for an 80-volumes grid is much lower than the one required for a 10240 control volumes, or even a 2560 one (Araki and Marchi, 2006).

As can be seen from Table 2 results, heat transfer effects are not important for non-dimensional momentum thrust determination: there is no modification on values, when the simulations including or not such effects are compared. For nozzle discharge coefficient, there is a tiny variation on values, which can be bigger than the error estimates (like for frozen flow), however, which is no greater than 1%. Larger differences are observed (Tables 2 and 3) for local variables of interest, such as exit pressure, temperature and velocity. The heat transfer effects assembled to the chemical reactive flow provides a reduction on exit pressure, exit temperature and exit velocity values. This drop is about 400-750 Pa for exit pressure (which is equal, at maximum, to a reduction of 2.6%), 35-70 K for exit temperature (at maximum, a 4.5% reduction) and 15-35 m/s for exit velocity (at maximum, a 1.0% reduction). Such variations are motivated by the heat conduction to the coolant: once this energy is transferred, the temperature of the reaction gas mixture drops and, consequently, all the other thermophysical parameters are affected, including the chemical composition (for equilibrium and non-equilibrium models), as can be seen at Table 5.

Table 2. Comparison for  $C_d$ ,  $F^*$  and  $P_{ex}$  (80 control volumes).

Results without heat transfer effects			
Model	$C_d$ [adim.]	$F^*$ [adim.]	$P_{ex}$ [Pa]
<b>Analytical (R1)</b>	<b>1.0</b>	<b>1.0</b>	<b>2.917342x10<sup>4</sup></b>
One-species, variable Properties (R2)	1.060 ± 3x10 <sup>-3</sup>	1.004 ± 4x10 <sup>-3</sup>	3.005x10 <sup>4</sup> ± 4x10 <sup>1</sup>
Frozen Flow – mod. 3, 4, 5 and 7	1.001 ± 3x10 <sup>-3</sup>	1.000 ± 4x10 <sup>-3</sup>	2.74x10 <sup>4</sup> ± 1x10 <sup>2</sup>
Frozen Flow – mod. 9 and 10	1.001 ± 3x10 <sup>-3</sup>	1.000 ± 4x10 <sup>-3</sup>	2.74x10 <sup>4</sup> ± 1x10 <sup>2</sup>
<b>CEA (frozen flow)</b>	<b>1.000580</b>	<b>0.998992</b>	<b>2.7448x10<sup>4</sup></b>
Equilibrium Flow – mod. 3, 4, 5 and 7	0.98 ± 1x10 <sup>-2</sup>	1.01 ± 1x10 <sup>-2</sup>	3.63x10 <sup>4</sup> ± 5x10 <sup>2</sup>
Equilibrium Flow – mod. 9 and 10	0.98 ± 1x10 <sup>-2</sup>	1.01 ± 1x10 <sup>-2</sup>	3.63x10 <sup>4</sup> ± 5x10 <sup>2</sup>
<b>CEA (local equilibrium flow)</b>	<b>0.977372</b>	<b>1.011553</b>	<b>3.6178x10<sup>4</sup></b>
Non-equilibrium Flow – mod. 31	1.008 ± 3x10 <sup>-3</sup>	1.012 ± 5x10 <sup>-3</sup>	3.175x10 <sup>4</sup> ± 7x10 <sup>1</sup>
Non-equilibrium Flow – mod. 32	1.007 ± 3x10 <sup>-3</sup>	1.014 ± 5x10 <sup>-3</sup>	3.254x10 <sup>4</sup> ± 6x10 <sup>1</sup>
Results taking into account heat transfer effects			
Model	$C_d$ [adim.]	$F^*$ [adim.]	$P_{ex}$ [Pa]
One-species, variable Properties (R2)	1.070 ± 3x10 <sup>-3</sup>	1.004 ± 4x10 <sup>-3</sup>	2.942x10 <sup>4</sup> ± 8x10 <sup>1</sup>
Frozen Flow – mod. 3	1.011 ± 3x10 <sup>-3</sup>	0.999 ± 4x10 <sup>-3</sup>	2.67x10 <sup>4</sup> ± 1x10 <sup>2</sup>
Frozen Flow – mod. 10	1.011 ± 3x10 <sup>-3</sup>	0.999 ± 4x10 <sup>-3</sup>	2.67x10 <sup>4</sup> ± 1x10 <sup>2</sup>
Equilibrium Flow – mod. 3	0.98 ± 1x10 <sup>-2</sup>	1.01 ± 1x10 <sup>-2</sup>	3.59x10 <sup>4</sup> ± 5x10 <sup>2</sup>
Equilibrium Flow – mod. 10	0.98 ± 1x10 <sup>-2</sup>	1.01 ± 1x10 <sup>-2</sup>	3.59x10 <sup>4</sup> ± 5x10 <sup>2</sup>
Non-equilibrium Flow – mod. 31	1.012 ± 3x10 <sup>-3</sup>	1.012 ± 5x10 <sup>-3</sup>	3.10x10 <sup>4</sup> ± 2x10 <sup>2</sup>
Non-equilibrium Flow – mod. 32	1.011 ± 3x10 <sup>-3</sup>	1.013 ± 3x10 <sup>-3</sup>	3.18x10 <sup>4</sup> ± 2x10 <sup>2</sup>

(R1):  $R_g = 526.97$  J/kg·K; (R2):  $R_g \approx 461.53$  J/kg·K (equivalent to combustion gases mixture for the ideal model)

Table 4 provides the numerical results for coolant properties (as well as for heat transfer parameters). As can be seen, numerical results are quite independent from the physical model choice; even for exit coolant temperature, for

which the results are different (based on numerical error estimates), the maximum difference among models is only 0.54 K. Larger differences are found for the maximum heat flux ( $q_{max}$ ) (from the reaction gas products to the coolant): it achieves  $8.3 \times 10^5$  W/m<sup>2</sup> (comparing the one-species gas and frozen flow models); the effect of such variation on the maximum wall temperature, however, is quite small: only 8.5 K (for the same physical models) and, if frozen and equilibrium results are compared, this difference is even smaller: 5.5 K. Based on these results and on CPU time consumption, which is presented at Table 6, for coolant and wall properties definition, the frozen flow model is a good choice (even for an initial analysis): its results are quite the same of the other physical models and its CPU time consumption is the lowest one (even smaller than the monogas with variable properties).

Table 3. Comparison for  $T_{ex}$ ,  $u_{ex}$  and  $M_{ex}$  (80 control volumes).

Results without heat transfer effects			
Model	$T_{ex}$ [K]	$u_{ex}$ [m/s]	$M_{ex}$ [adim.]
<b>Analytical (R1)</b>	<b>1712.7409</b>	<b>3316.7150</b>	<b>3.1928346</b>
One-species, variable Properties (R2)	1800 ± 7	3142 ± 6	3.15 ± 1x10 <sup>-2</sup>
Frozen Flow – mod. 3, 4, 5 and 7	1606 ± 9	3312 ± 7	3.24 ± 1x10 <sup>-2</sup>
Frozen Flow – mod. 9	1606 ± 9	3312 ± 7	3.24 ± 1x10 <sup>-2</sup>
<b>CEA (frozen flow)</b>	<b>1607.91</b>	<b>3311.4519</b>	<b>3.231</b>
Equilibrium Flow – mod. 3, 4, 5 and 7	2461.2 ± 3x10 <sup>-1</sup>	3427 ± 2	2.911 ± 2x10 <sup>-3</sup>
Equilibrium Flow – mod. 9 and 10	2461.4 ± 3x10 <sup>-1</sup>	3427 ± 2	2.911 ± 2x10 <sup>-3</sup>
<b>CEA (local equilibrium flow)</b>	<b>2462.41</b>	<b>3432.7056</b>	<b>2.986</b>
Non-equilibrium Flow – mod. 31	1910 ± 1x10 <sup>1</sup>	3332 ± 6	3.05 ± 1x10 <sup>-2</sup>
Non-equilibrium Flow – mod. 32	1980 ± 1x10 <sup>1</sup>	3338 ± 6	3.02 ± 1x10 <sup>-2</sup>
Results taking into account heat transfer effects			
Model	$T_{ex}$ [K]	$u_{ex}$ [m/s]	$M_{ex}$ [adim.]
One-species, variable Properties (R2)	1730 ± 7	3112 ± 6	3.18 ± 1x10 <sup>-2</sup>
Frozen Flow – mod. 3	1534 ± 9	3278 ± 7	3.27 ± 2x10 <sup>-2</sup>
Frozen Flow – mod. 10	1534 ± 9	3278 ± 7	3.27 ± 2x10 <sup>-2</sup>
Equilibrium Flow – mod. 3	2425.4 ± 8x10 <sup>-1</sup>	3409 ± 2	2.922 ± 2x10 <sup>-3</sup>
Equilibrium Flow – mod. 10	2425.6 ± 8x10 <sup>-1</sup>	3409 ± 2	2.922 ± 2x10 <sup>-3</sup>
Non-equilibrium Flow – mod. 31	1860 ± 1x10 <sup>1</sup>	3315 ± 6	3.08 ± 1x10 <sup>-2</sup>
Non-equilibrium Flow – mod. 32	1924 ± 9	3320 ± 2x10 <sup>1</sup>	3.05 ± 1x10 <sup>-2</sup>

(R1):  $R_g = 526.97$  J/kg·K; (R2):  $R_g \approx 461.53$  J/kg·K (equivalent to combustion gases mixture for the ideal model)

Table 4. Comparison for  $\Delta P$ ,  $T_{cool}$ ,  $q_{max}$  and  $T_{max}$  with heat transfer effects (80 control volumes).

Model	$\Delta P$ [Pa]	$T_{cool}$ [K]	$q_{max}$ [W/m <sup>2</sup> ]	$T_{max}$ [K]
One-species, variable Properties (R2)	$8.4 \times 10^5 \pm 3 \times 10^4$	$311.36 \pm 2 \times 10^{-2}$	$3.197 \times 10^7 \pm 5 \times 10^4$	$620.0 \pm 3 \times 10^{-1}$
Frozen Flow – mod. 3	$8.4 \times 10^5 \pm 3 \times 10^4$	$311.6 \pm 2 \times 10^{-1}$	$3.28 \times 10^7 \pm 1 \times 10^5$	$628.5 \pm 7 \times 10^{-1}$
Frozen Flow – mod. 10	$8.4 \times 10^5 \pm 3 \times 10^4$	$311.6 \pm 2 \times 10^{-1}$	$3.28 \times 10^7 \pm 1 \times 10^5$	$628.5 \pm 7 \times 10^{-1}$
Equilibrium Flow – mod. 3	$8.4 \times 10^5 \pm 3 \times 10^4$	$311.9 \pm 2 \times 10^{-1}$	$3.23 \times 10^7 \pm 4 \times 10^5$	623 ± 3
Equilibrium Flow – mod. 10	$8.4 \times 10^5 \pm 3 \times 10^4$	$311.9 \pm 2 \times 10^{-1}$	$3.23 \times 10^7 \pm 4 \times 10^5$	623 ± 3
Non-equilibrium Flow – mod. 31	$8.4 \times 10^5 \pm 3 \times 10^4$	$311.40 \pm 2 \times 10^{-2}$	$3.235 \times 10^7 \pm 9 \times 10^4$	$624.0 \pm 4 \times 10^{-1}$
Non-equilibrium Flow – mod 32	$8.4 \times 10^5 \pm 3 \times 10^4$	$311.46 \pm 2 \times 10^{-2}$	$3.240 \times 10^7 \pm 9 \times 10^4$	$624.5 \pm 4 \times 10^{-1}$

(R2):  $R_g \approx 461.53$  J/kg·K (equivalent to combustion gases mixture for the ideal model)

Table 5. Reaction gas products composition at nozzle engine exit (80 control volumes).

Model		H <sub>2</sub> O	O <sub>2</sub>	H <sub>2</sub>	OH	O	H	HO <sub>2</sub>	H <sub>2</sub> O <sub>2</sub>	O <sub>3</sub>
Frozen Flow, without heat transfer effects	3	0.78369	0.07754	0.01565	0.10276	0.01790	0.00247	---	---	---
	10	0.78354	0.07743	0.01565	0.10272	0.01789	0.00247	0.00027	0.00004	---
	<b>CEA</b>	<b>0.77987</b>	<b>0.07515</b>	<b>0.01570</b>	<b>0.10900</b>	<b>0.01751</b>	<b>0.00246</b>	<b>0.00027</b>	<b>0.00004</b>	<b>&lt;0.00001</b>
Equilibrium Flow, without heat transfer effects	3	0.92742	0.03659	0.00606	0.02687	0.00259	0.00047	---	---	---
	10	0.92736	0.03661	0.00606	0.02689	0.00260	0.00047	0.00001	9.79x10 <sup>-7</sup>	---
	<b>CEA</b>	<b>0.92548</b>	<b>0.03579</b>	<b>0.00611</b>	<b>0.02956</b>	<b>0.00257</b>	<b>0.00047</b>	<b>0.00001</b>	<b>&lt;0.00001</b>	<b>&lt;0.00001</b>
Non-equilibrium Flow, without heat transfer effects	31	0.81253	0.10023	0.01709	0.05351	0.01592	0.00072	---	---	---
	32	0.82375	0.09475	0.01600	0.05132	0.01349	0.00068	---	---	---
Frozen flow, taking into account heat transfer effects	3	0.78369	0.07754	0.01565	0.10276	0.01790	0.00247	---	---	---
	10	0.78354	0.07743	0.01565	0.10272	0.01789	0.00247	0.00027	0.00004	---
Equilibrium flow, taking into account heat transfer effects	3	0.93600	0.03279	0.00540	0.02339	0.00205	0.00038	---	---	---
	10	0.93595	0.03281	0.00540	0.02340	0.00205	0.00038	0.00001	<0.00001	---
Non-equilibrium flow, taking into account heat transfer effects	31	0.81681	0.09883	0.01680	0.05189	0.01504	0.00062	---	---	---
	32	0.82832	0.09342	0.01570	0.04929	0.01269	0.00059	---	---	---



Table 6. CPU time physical models; simulations including heat transfer effects (80 control volumes).

Model	Iterations for:			Global iterations	CPU time
	Physical	Chemical	Reaction products flow Coolant flow		
Monogas, variable properties	---	6,000	1,000	20	23.4 s
Frozen Flow	3	5,000	1,000	20	10.6 s
	10	5,000	1,000	20	12.2 s
Equilibrium Flow	3	15,000	1,000	20	1.79 h
	10	15,000	1,000	20	3.49 h
Non-Equilibrium Flow	31	5,000,000	1,000	5	1.06 day
	32	4,000,000	1,000	5	20.0 h

From Table 5, it can be noted that the reaction gas products compositions, for both equilibrium and non-equilibrium flows, taking into account heat transfer effects, are only a little different from those obtained for adiabatic walls. It is related to the small drop at the temperature profile through the entire nozzle engine, considering heat transfer effects, as can be seen at Fig. 4. The lower the temperature, the smaller the species dissociation effects; because of this, all the species mass fractions drop when the heat transfer effects are considered, but the H<sub>2</sub>O one.

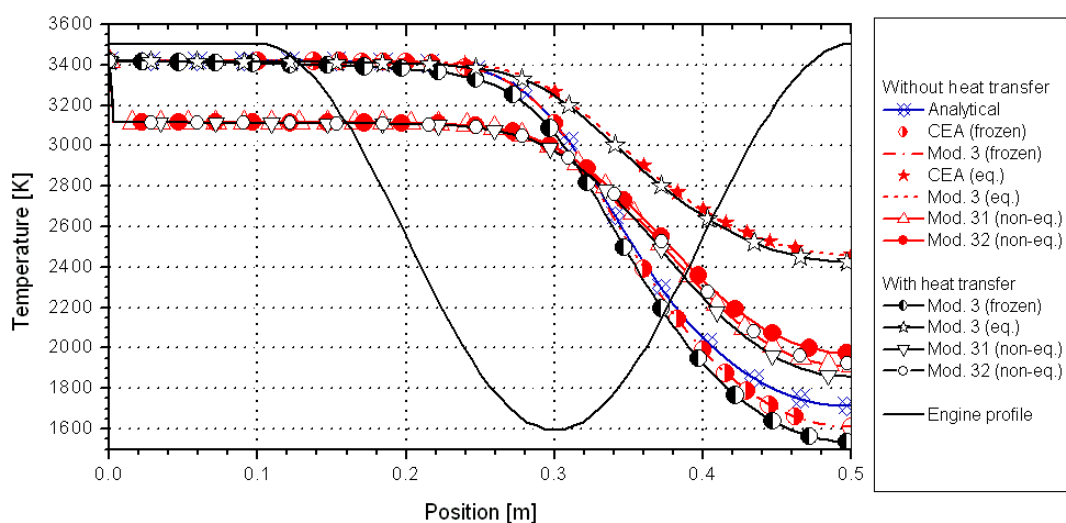


Figure 4. Temperature distribution of reaction gas products through the nozzle engine.

Beside the CPU time consumption for physical and chemical models studied in this work, Table 6 provides the number of iterations for solving each problem (reaction gas products and coolant flows) separately and, also, the number of global iterations. These amounts of iterations were chosen in order to allow the convergence of each sub-problem separately (reaction gas products and coolant flows), at each global iteration. The number of total global iterations was chosen in order to achieve the round-off error (except for non-equilibrium flow, for which a tolerance of  $10^{-5}$  was specified). This proceeding was necessary to minimize other types of numerical errors than discretization ones and, in this way, to guarantee that numerical errors are made, essentially, of discretization ones.

While Fig. 4 presents the temperature distribution of reaction gas products through the nozzle engine, Fig. 5 provides the wall temperature (in contact with reaction gas products) along the whole engine. Despite the differences on temperature values at thrust chamber entrance and exit, all the physical models have similar solutions for maximum wall temperature (actually, the difference at numerical results is smaller than 10K, as previously commented).

Although frozen flow presents lower temperatures for reaction gas products than equilibrium flow ones, wall temperatures are, at least from thrust chamber entrance to nozzle throat, higher than those obtained for equilibrium flow studies. This phenomenon is explained by the higher values for convection heat transfer coefficient verified for frozen flow, which achieves, at maximum,  $11,691 \text{ W/m}^2\cdot\text{K}$  at nozzle throat region (against  $10,861 \text{ W/m}^2\cdot\text{K}$  for equilibrium flow model). Thus, even though the recombination reactions, which take place along all the thrust chamber length and forms H<sub>2</sub>O species, are exothermic and contribute to the higher reaction gas products temperatures, the higher values for convection heat transfer coefficient, verified for frozen flow model, overtake these recombination effects and its associated higher temperature, at least until nozzle throat region, where the highest wall temperatures are found. Otherwise, the difference between frozen flow and equilibrium flow reaction gas temperatures after the nozzle throat region becomes so expressive that the effects of the higher values for convection heat transfer coefficient (verified for frozen flow model) are overtaken by the higher equilibrium flow temperature values for reaction gas products. Because of this, the wall temperatures, from the nozzle throat to the thrust chamber exit, are higher for the equilibrium flow model.

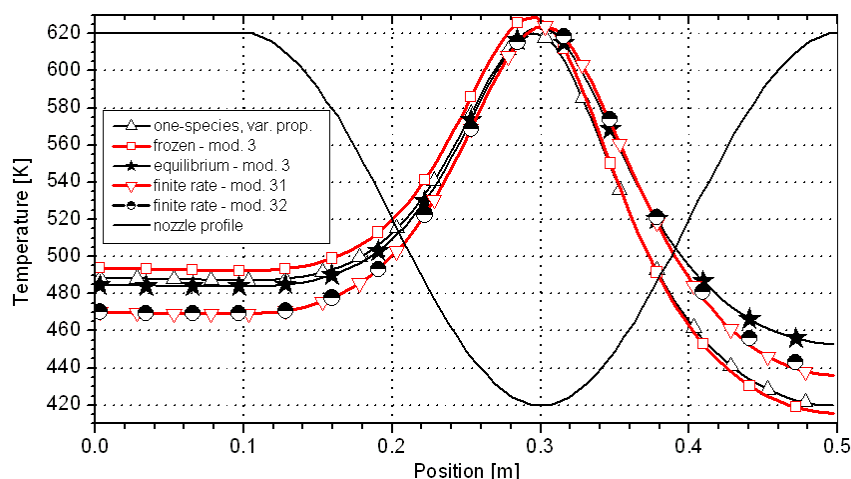


Figure 5. Wall temperature distribution (in contact with reaction gas products) through the nozzle engine.

## 6. Conclusion

A software for one-dimensional chemical reactive flow with regenerative cooling was implemented, using FORTRAN 95 language and Visual Compaq 6.6.0 compiler. Four physical models were studied: a one-species flow (with variable properties); and the other three for reaction gas products (frozen, local equilibrium and non-equilibrium flows). For frozen and equilibrium models, nine different chemical reaction schemes were implemented, the same chemical reaction schemes discussed by Marchi *et al.* (2005), but only two were studied (a six and a eight-species models) and the results were compared to those ones obtained for adiabatic wall studies, including those ones from the NASA's software CEA. For non-equilibrium, only six and eight-species reaction schemes were employed, totalising six models, comparing the results to frozen and local equilibrium ones. Unfortunately, because of time restrictions, only two from these six models were studied.

Differently as usually found in literature, the energy equation has the temperature as unknown (and not the enthalpy nor the internal energy, as commonly used). The major advantage of such variation is about the temperature determination, which is done directly from the numerical model. Also, the velocity determination is done from the combustion chamber (where the velocity is almost null) to the nozzle exit (supersonic flow), covering all the velocity regimes in a real engine (subsonic, transonic and supersonic ones), and not only the supersonic flow, as commonly done.

Assembling refrigeration effects to reaction gas products flow has only small influence on global parameters of interest: for nozzle discharge coefficient, there is only a tiny variation on values, which can be bigger than the error estimates (like for frozen flow), however, which is no greater than 1%; for non-dimensional momentum thrust, no appreciable variation on numerical results was verified. For local parameters of interest, otherwise, larger differences were observed, especially for exit temperature values, for which a drop between 35 and 70 K at exit temperature (at maximum, a 4.5% reduction, for frozen flow model) was verified. At exit velocity and exit pressure, such variations are smaller: only a 1.0% reduction for velocity (15-35 m/s) and a 2.6% reduction for pressure (about 400-750 Pa). The reaction gas products compositions, for both equilibrium and non-equilibrium flows, taking into account heat transfer effects, are only a little different from those obtained for adiabatic walls.

Comparing the physical models results for coolant parameters of interest, it was observed only a small variation on coolant exit temperature: 0.54 K, while the pressure drop values are the same for all the models studied. For maximum heat flux (from the reaction gas products to the coolant), the difference more expressive: it achieves  $8.3 \times 10^5$  W/m<sup>2</sup> (comparing the one-species gas and frozen flow models); the effect of such variation on the maximum wall temperature, however, is quite small: only 8.5 K (for the same physical models).

Frozen flow model presents higher values for wall temperatures from thrust chamber entrance until its throat region, although equilibrium flow model shows greater reaction gas products temperatures. This phenomenon is consequence of the higher values for convection heat transfer coefficient verified for frozen flow model and, because of this, the highest value for maximum wall temperature was 628.5 K, for frozen flow model. From the nozzle throat region to the exit thrust chamber, the effects of the higher values for convection heat transfer coefficient are overtaken by those ones caused by the greater reaction gas products temperature, what explains the higher wall temperature values for equilibrium flow after the nozzle throat region.

For all the studies, 80-volumes grids were employed, once they provide numerical error estimates at the same magnitude of experimental ones (Marchi *et al.*, 2004; Araki and Marchi, 2006). Comparing numerical results (for coolant and wall properties definition) and also CPU time requirements, the frozen flow model is a good choice (even for an initial analysis): its results are quite the same of the other physical models and its CPU time consumption is the

lowest one (even smaller than the monogas with variable properties, which demands at least 90% more time): it is at least more than 500 times faster than the equilibrium flow model and 5500 times faster than the non-equilibrium one.

## 7. Acknowledgement

The authors would acknowledge Federal University of Paraná (UFPR), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) and The “UNIESPAÇO Program” of The Brazilian Space Agency (AEB) by physical and financial support given for this work. The first author would, also, acknowledge his professors and friends, by discussions and other forms of support.

## 8. References

- Araki, L. K., Marchi, C. H., 2006, “Effects of Chemical Reaction Schemes and Physical Models on a One-Dimensional Flow in a Rocket Engine Nozzle”, Proceedings of the 27th Iberian Latin-American Congress on Computational Methods in Engineering, Belém, Brazil. Paper CIL04-500. (Paper submitted.)
- Barros, J. E. M., Alvim Filho, G. F., Paglione, P., 1990, “Estudo de Escoamento Reativo em Desequilíbrio Químico através de Bocais Convergente Divergente”, Proceedings of the III Encontro Nacional de Ciências Térmicas, Itapema, Brazil.
- Ferziger, J. H. and Perić, M., 2001, “Computational Methods for Fluid Dynamics”, 3ed., Berlin: Springer-Verlag.
- Fröhlich, A., Popp, M., Schmidt, G., Thelemann, D., 1993, “Heat Transfer Characteristics of H<sub>2</sub>/O<sub>2</sub> Combustion Chambers”, Proceedings of the 29<sup>th</sup> Joint Propulsion Conference, Monterrey, USA, AIAA 93-1826.
- Glenn Research Center, 2005, “CEA – Chemical Equilibrium with Applications”, available at: [http://www.grc.nasa.gov/WWW/CEA\\_Web/ceaHome.htm](http://www.grc.nasa.gov/WWW/CEA_Web/ceaHome.htm), access in: Feb 16, 2005.
- Kee, R. J., GrCar, J. F., Smooke, M. D., Miller, J. A., 1990, “A Fortran Program for Modeling Steady Laminar One-Dimensional Premixed Flames”, Albuquerque: Sandia National Laboratories, SAND85-8240•UC-401.
- Marchi, C. H., 2001, “Verificação de Soluções Numéricas Unidimensionais em Dinâmica dos Fluidos”, Thesis (Mechanical Engineering PhD). Universidade Federal de Santa Catarina, Florianópolis, SC.
- Marchi, C. H., Araki, L. K., Laroca, F., 2005, “Evaluation of Thermochemical Properties and Combustion Temperatures for LOX/LH<sub>2</sub> Reaction Schemes”, Proceedings of the 26<sup>th</sup> Iberian Latin-American Congress on Computational Methods in Engineering, Guarapari, Brazil. Paper CIL06-0095.
- Marchi, C. H., Laroca, F., Silva, A. F. C., Hinckel, J. N., 2000, “Solução Numérica de Escoamentos em Motor-Foguete com Refrigeração Regenerativa”, Proceedings of the 21<sup>st</sup> Iberian Latin-American Congress on Computational Methods in Engineering, Rio de Janeiro, Brazil.
- Marchi, C. H., Laroca, F., Silva, A. F. C., Hinckel, J. N., 2004, “Numerical Solutions of Flows in Rocket Engines with Regenerative Cooling”, Numerical Heat Transfer, *Part A*, Vol.45, pp. 699-717.
- Marchi, C. H., Maliska, C. R., 1994, “A nonorthogonal finite volume method for the solution of all speed flows using co-located variables”, Numerical Heat Transfer, *Part B*, v. 26, pp. 293 – 311.
- Marchi, C. H., Silva, A. F. C., 2002, “Unidimensional numerical solution error estimation for convergent apparent order”, Numerical Heat Transfer, *Part B*, Vol. 42, pp. 167-188.
- Roache, P. J., 1994, “Perspective: A method for uniform reporting of grid refinement studies”, Journal of Fluids Engineering, Vol. 116, pp. 405-413.
- Smith, T. A., Pavli, A. J., Kacynski, K. J., 1987, “Comparison of Theoretical and Experimental Thrust Performance of a 1030:1 Area Ratio Rocket Nozzle at a Chamber Pressure of 2413 kN/m<sup>2</sup> (350 psia)”, Cleveland, NASA Lewis Research Center, NASA Technical Paper 2725.
- Sutton, G. P., Biblarz, O., 2001, “Rocket Propulsion Elements”, 7 ed., New York, John Wiley & Sons.
- Svehla, R. A., 1964, “Thermodynamic and Transport Properties for the Hydrogen-Oxygen System”, Cleveland, NASA Lewis Center, NASA SP-3011.
- Tannehill, J. C., Anderson, D., Pletcher, R. H., 1997, “Computational Fluid Mechanics and Heat Transfer”, 2ed., Philadelphia: Taylor & Francis.
- Van Doormal, J. P., Raithby, G. D., 1984, “Enhancements of the SIMPLE method for predicting incompressible fluid flow”, Numerical Heat Transfer, v. 7, pp. 147-163.

## INVERSE DESIGN OF A ILLUMINATION SYSTEM HAVING NON-GRAY WALLS

### Alexandre Seewald

Department of Mechanical Engineering  
Federal University of Rio Grande do Sul  
Porto Alegre, RS, Brazil  
seewald@mecanica.ufrgs.br

### Paulo Smith Schneider

Department of Mechanical Engineering  
Federal University of Rio Grande do Sul  
Porto Alegre, RS, Brazil  
pss@mecanica.ufrgs.br

### Francis Henrique Ramos França

Department of Mechanical Engineering  
Federal University of Rio Grande do Sul  
Porto Alegre, RS, Brazil  
frfranca@mecanica.ufrgs.br

*Abstract. This work applies the inverse analysis to an illumination design of a three-dimensional rectangular enclosure having non-gray walls. The problem consists of finding the luminous fluxes input on the light source elements, located on the top of the enclosure, to satisfy a prescribed uniform luminous flux on the design surface, located on the bottom surface. All the enclosure surfaces emit and reflect diffusively, but the hemispheric spectral emissivities are wavelength dependent in the visible region of the spectrum. The illumination design is inherently an inverse problem, in which the design surface is subjected to two conditions while the light sources are left unconstrained. Such problems are known to be ill-conditioned, and requires regularization methods to deal with the system of equations. In addition, the dependence of the surfaces emissivity on the wavelength lead to a system of non-linear equations. The ill-conditioned characteristics of the system of equations are tackled by the TSVD (Truncated Singular Value Decomposition) regularization method, while an iterative procedure is proposed to deal with the non-linearity of the equations. This work considers two design cases: one in which the light source elements cover the entire top surface; and one in which a reduced number of light sources are considered, a more practical solution. In both cases, the proposed inverse design is capable of providing a solution that satisfies the prescribed luminous flux on the design surface within average and maximum errors less than 2,0 and 10%, respectively.*

*Keywords. Inverse analysis, illumination design, non-gray surfaces, radiation exchanges, regularization methods*

### 1. Introduction

A large variety of environmental control and industrial processes require the specification of the illumination level on a certain working area. Examples include plant growth as well as human and industrial activities. The designer must then specify the locations and power inputs of the light sources so that the required illumination flux is achieved. This problem usually involves a trial-and-error approach, in which the light sources are specified and calculations are run to verify if the required illumination is achieved. If not, another configuration is attempted, until the resulting illumination is satisfactory. This trial-and-error procedure may require many steps, since when an unsatisfactory result is obtained it is not clear which direction to take next, due to the complexity and non-linearity of the illumination relations. The illumination inverse design is an alternative procedure, from which the light sources configurations are directly found from the specified illumination flux.

The methods for the design of artificial lighting of environments were first established at the beginning of the 20th century. The luminous flux on a given working area is not only dependent on the power of the light sources, but also on the absorbing and reflecting effect of the remaining surfaces of the enclosure. The methods allowed the calculation of light radiation exchanges as well as for the characterization of the light sources behavior. In addition, many studies have been carried out to provide recommending lighting for the many possible applications, from human and industrial activities to plants growth (Boast, 1953; Mark, 2000). In general, not only the intensity of light (luminous flux intensity) is specified, but its uniformity.

The first works to deal with the illumination design were presented by Harrison and Anderson (1916 and 1920), who proposed an experimental procedure, the lumen method, in which the luminous flux on a working plane was determined from a combination of a series of proposed assembling of punctual and continuous light sources. In the 1940's, Moon (1941) and Moon and Spencer (1946a, 1946b) proposed the interreflection method for the design of

three-dimensional rectangular enclosures having any aspect ratio and being formed by diffuse surfaces. The method presented the advantage of allowing the calculation of the brightness of a surface, accounting for the reflection of light.

The advances in the field thermal radiation in the past century led to many analytical and computational tools that can be readily applied to illumination analysis, since light itself and thermal radiation in the visible region of the spectrum are essentially the same phenomenon. The sole difference is that the thermal radiation flux must be converted into luminous flux, accounting for the photopic luminous efficacy of the human eye. Schneider and França (2004) presented the application of the inverse analysis to the illumination design of an enclosure in which an uniform luminous flux was imposed on a working surface of the enclosure, the *design surface*. The work showed that such surfaces were constrained by two boundary conditions: the specified luminous flux and the luminous emissive power, which is zero. In the analysis, the luminous power required on the light sources were found directly from the two conditions imposed on the design surfaces, without involving the trial-and-error solution of the conventional techniques. However, as typical of inverse analysis, the formulation resulted in an ill-conditioned system of equations, which requires special solution techniques, such as the regularization methods (Hansen, 1998). A comprehensive review of inverse design involving radiative exchanges in enclosures was presented in França et al. (2003).

This paper aims at extending the analysis presented in Schneider and França (2004) to include surfaces that are non-gray. In such a case, the system of equations is non-linear in addition to the ill-posed characteristics of the inverse problem. The objective is to find the luminous power on the light sources located on the top surfaces of a three-dimensional enclosure that satisfy the specified uniform luminous flux on the design surface, located on the bottom of the enclosure. All the surfaces that form the enclosure are assumed to be diffuse. A zonal type formulation is applied for the discretization of the radiation exchanges. The ill-conditioned nature of the system is treated by means of the truncated singular value decomposition (TSVD). Results are presented for two cases: when the light sources cover the entire top surface; and when a reduced amount of light sources are considered, a more practical design proposition.

## 2. Physical and mathematical modeling

### 2.1 Luminous flux and thermal radiation

Incandescent lamps are common sources of light. Their main component is a resistance that reach temperatures around 3000 K under the passage of electric current. At such temperatures, a considerable amount of thermal radiation is emitted in the visible region of the wavelength spectrum,  $0.4 \mu\text{m} \leq \lambda \leq 0.7 \mu\text{m}$ . The luminous flux, in units of lumens/m<sup>2</sup> or lux, can be related to the thermal radiation flux, in units of W/m<sup>2</sup>, by means of the following relation:

$$q^{(l)} = CV_{\lambda}q^{(w)} \quad (1)$$

where  $q^{(l)}$  and  $q^{(w)}$  correspond respectively to the luminous flux (lumens/m<sup>2</sup>) and to the thermal radiation flux (W/m<sup>2</sup>) for a specific wavelength  $\lambda$ ,  $C$  is a conversion factor constant, equal to 683 lumens/W, and  $V_{\lambda}$  is the so called photopic spectral luminous efficacy of the human eye, which takes into account the human eye sensitivity to the thermal radiation comprehended in the visible region of the spectrum, and is obtained by the following curve fitting:

$$V_{\lambda} = c_1 \exp(c_2 + c_3 / \lambda + c_4 \ln(\lambda)) \quad (2)$$

where  $c_1 = 9.77517 \times 10^{-1}$ ,  $c_2 = 1.31695 \times 10^3$ ,  $c_3 = -1.00004 \times 10^5$ ,  $c_4 = -1.7989 \times 10^2$ , and  $\lambda$  is in nm. According to the above relation, the spectral luminous efficacy peaks with a value of 1.0 for a thermal radiation in the wavelength of 0.555  $\mu\text{m}$ , and then decay monotonically to zero as the lower and upper limits of the visible region, 0.4  $\mu\text{m}$  and 0.7  $\mu\text{m}$ , are approached. In general, a source of light is composed by radiation covering the entire range of the visible region. In such a case, Eq. (1) must be applied to each infinitesimal amount of the spectral energy and then be integrated in the visible spectrum. Such procedure will be demonstrated in the next section.

### 2.2 Radiation Exchanges in an Enclosure

Methods for the evaluation of radiation exchanges in an enclosure is well established in the literature (Siegel and Howell, 2002). To allow acceptable resolution in the variation of the radiative quantities, the surfaces are subdivided into sufficiently small elements, in which all thermal quantities can be assumed uniform. Next, the energy balance will be applied to each element  $j$  of the enclosure. The energy balance can take different forms as shown below:

$$q_{\lambda,o,j}^{(w)} = \varepsilon_{\lambda,j} e_{\lambda,b,j}^{(w)} + (1 - \varepsilon_{\lambda,j}) q_{\lambda,i,j}^{(w)} \quad (3)$$

$$q_{\lambda,i,j}^{(w)} = \sum_k F_{j-k} q_{\lambda,o,k}^{(w)} \quad (4)$$

In the above equations,  $q_{\lambda,o}^{(w)}$  is the spectral radiosity, in  $W/(m^2\mu m)$ , which takes into account both emission and reflection;  $q_{\lambda,i}^{(w)}$  is the spectral irradiation, in  $W/(m^2\mu m)$ , corresponding to all radiation that arrives on the surface element;  $e_{\lambda,b}^{(w)}$  is the spectral blackbody emissive power, in  $W/(m^2\mu m)$ , which is given by Planck's blackbody distribution;  $\varepsilon_\lambda$  is the spectral emissivity; and  $F_{j-k}$  is the view factor between surface elements  $j$  and  $k$ . In the above equations the superscript  $(w)$  was used to indicate thermal energy. In a next step, the above equations are converted into luminous quantities. To accomplish this, it should be first noted that the thermal energy flux ( $W/m^2$ ) in the wavelength  $\lambda$  centered in the interval  $d\lambda$  can be related to the spectral energy flux  $W/(m^2\mu m)$  by:

$$dq^{(w)} = q_\lambda^{(w)} d\lambda \quad (5)$$

The corresponding luminous flux (lumens) in the wavelength  $\lambda$  centered in the interval  $d\lambda$  follows from Eq. (1):

$$dq^{(l)} = CV_\lambda q_\lambda^{(w)} d\lambda = q_\lambda^{(l)} d\lambda \quad (6)$$

where  $q_\lambda^{(l)}$  is the spectral luminous flux, in lumens/( $m^2\mu m$ ). Multiplying Eqs. (3) and (4) by  $CV_\lambda$  leads to:

$$q_{\lambda,o,j}^{(l)} = \varepsilon_{\lambda,j} e_{\lambda,b,j}^{(l)} + (1 - \varepsilon_{\lambda,j}) q_{\lambda,i,j}^{(l)} \quad (7)$$

$$q_{\lambda,i,j}^{(l)} = \sum_k F_{j-k} q_{\lambda,o,k}^{(l)} \quad (8)$$

where  $e_{\lambda,b,j}^{(l)}$  is the blackbody spectral luminous power, given by  $e_{\lambda,b,j}^{(l)} = CV_\lambda e_{\lambda,b,j}^{(w)}$ .

The above relations present the luminous energy balance for a given spectral wavelength  $\lambda$ . The complete solution requires the integration of the above relation for the entire spectral region that contains the visible region, that is  $0.4 \mu m \leq \lambda \leq 0.7 \mu m$ . For gray surfaces, the spectral hemispherical emissivities  $\varepsilon_{\lambda,j}$  can be readily removed from the integral, and the integration can be performed by simply introducing the total luminous quantities (Schneider and França, 2004). For non-gray surfaces, the spectrum domain can be divided into bands  $\Delta\lambda$  where the spectral hemispherical emissivities  $\varepsilon_{\lambda,j}$  for the surfaces can be approximated by a constant value, represented by  $\varepsilon_{\Delta\lambda,j}$ . It follows that:

$$q_{\Delta\lambda,o,j}^{(l)} = \varepsilon_{\Delta\lambda,j} e_{\Delta\lambda,b,j}^{(l)} + (1 - \varepsilon_{\Delta\lambda,j}) q_{\Delta\lambda,i,j}^{(l)} \quad (9)$$

$$q_{\Delta\lambda,i,j}^{(l)} = \sum_k F_{j-k} q_{\Delta\lambda,o,k}^{(l)} \quad (10)$$

where  $q_{\Delta\lambda}^{(l)}$  is given by integration of  $q_\lambda^{(l)}$  over band  $\Delta\lambda$ . Finally, the total luminous fluxes are obtained by the summation of the contribution of all bands, that is:

$$q_{o,j}^{(l)} = \sum_{\Delta\lambda} q_{\Delta\lambda,o,j}^{(l)} \quad (11)$$

$$q_{i,j}^{(l)} = \sum_{\Delta\lambda} q_{\Delta\lambda,i,j}^{(l)} \quad (12)$$

$$e_{b,j}^{(l)} = \sum_{\Delta\lambda} e_{\Delta\lambda,b,j}^{(l)} \quad (13)$$

where  $q_o^{(l)}$  is the outgoing luminous flux, in lumens/ $m^2$  or lux, which takes into account both emission and reflection;  $q_i^{(l)}$  is the incident luminous flux, in lumens/ $m^2$ , the luminous energy that arrives from the light sources as well as from reflections from the side walls.

### 3. Problem definition

Figure 1 presents a schematic view of a three-dimensional enclosure, which is formed by surfaces that are diffuse but non-gray. The design surface is located on the bottom of the enclosure; the light sources are located on the top surface. The remaining of the enclosure is formed by walls that do not emit but reflect incident light. The length, width and height of the enclosure are designated by  $L$ ,  $W$  and  $H$ , respectively. As depicted in Fig. 2, the enclosure is divided into finite-size square elements,  $\Delta x = \Delta y = \Delta z$ , in which the luminous energy balance will be applied. For designation of elements in the design surface, lamps and wall, indices  $jd$ ,  $jl$  and  $jw$  will be used throughout this analysis.

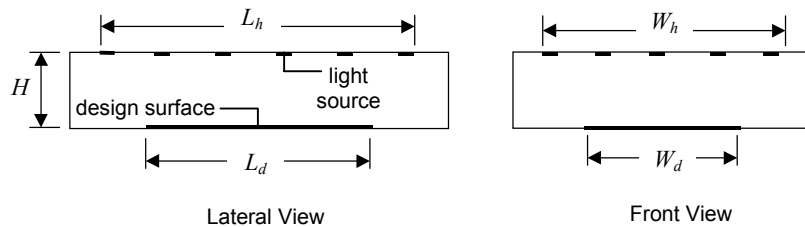


Figure 1. Three-dimensional rectangular enclosure.

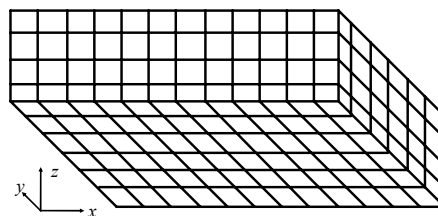


Figure 2. Division of the bottom and two side surfaces of the enclosure into finite size elements.

An uniform incident luminous flux, designated by  $q_{specified}^{(l)}$ , is specified on the design surface. In addition, no light is emitted from the design surface, so  $e_{b,jd}^{(l)} = 0$ . Therefore, two conditions are imposed on the design surface elements. For the elements on the wall, one single conditions is known,  $e_{b,jw}^{(l)} = 0$ , since they do not emit light. On the other hand, no condition is known for the light source elements. The conditions on the light sources are to be found from the two specifications ( $q_{i,jd}^{(l)} = q_{specified}^{(l)}$  and  $e_{b,jd}^{(l)} = 0$ ) on the design surface.

The conventional solution techniques deal with problems where only one condition (either the incident luminous flux or the blackbody luminous power) is imposed on each surface element. This leads to a system of equations on the unknown outgoing luminous flux of each element. This system is typically well-behaved and can be solved by any standard matrix inversion technique. Once the system is solved for the outgoing luminous fluxes, the conservation equations can be again applied to find the unknown condition (the incident luminous flux or the blackbody luminous power) of each surface element. Since the knowledge of one condition is required on each surface element, the conventional technique can tackle the illumination design only through a trial-and-error solution, in which the power luminous flux is guessed on each light source, and the solution is run to verify if the imposed incident luminous flux on the design surface is attained. Due to the nature of the problem, one should expect to make a large number of guesses to obtain an approximate solution.

The inverse design is an alternative technique to tackle the above problem without relying on a trial-and-error solution. Note that for a design surface element  $jd$ , since  $e_{\Delta\lambda,b,jd}^{(l)} = 0$ , the outgoing luminous flux in band  $\Delta\lambda$  can be found directly from the specified incident luminous flux according to Eq. (9):

$$q_{\Delta\lambda,o,jd}^{(l)} = (1 - \epsilon_{\Delta\lambda,jd}) q_{\Delta\lambda,i,jd}^{(l)} \quad (14)$$

Writing the above equation in dimensionless form, one obtains:

$$Q_{\Delta\lambda,o,jd} = (1 - \varepsilon_{\Delta\lambda,jd}) Q_{\Delta\lambda,i,jd} \quad (15)$$

The dimensionless luminous flux is given by  $Q = q^{(l)} / q_{i,specified}^{(l)}$ , so that  $Q_{i,jd} = 1.0$ , where:

$$Q_{i,jd} = \sum_{\Delta\lambda} Q_{\Delta\lambda,i,jd} \quad (16)$$

Next, Eqs. (9) and (10) are combined and applied to each design surface element  $jd$  (with  $e_{\Delta\lambda,b,jd}^{(l)} = 0$ ) and rearranged to form a system of equations for the dimensionless outgoing luminous flux on the light source elements  $Q_{\Delta\lambda,o,jl}$ :

$$\sum_{jl} F_{jd-jl} Q_{\Delta\lambda,o,jl} = \frac{Q_{\Delta\lambda,o,jd}}{(1 - \varepsilon_{\Delta\lambda,j})} - \sum_{jw} F_{jd-jw} Q_{\Delta\lambda,o,jw} \quad (17)$$

Note that in the above equation, the first term on the right-hand side is known. The outgoing luminous fluxes on the wall elements,  $Q_{\Delta\lambda,o,jw}$ , are unknown, so an additional relation is imposed by applying Eqs. (9) and (10) to these elements with the condition that  $e_{\Delta\lambda,b,jw}^{(l)} = 0$ . It follows that:

$$Q_{\Delta\lambda,o,jw} = (1 - \varepsilon_{\Delta\lambda,jw}) \left( \sum_{jd} F_{jw-jd} Q_{\Delta\lambda,o,jd} + \sum_{jl} F_{jw-jl} Q_{\Delta\lambda,o,jl} + \sum_{jw'-jw''} F_{jw'-jw''} Q_{\Delta\lambda,o,jw''} \right) \quad (18)$$

The last term arises from the fact that, as seen in Fig. 1, two walls element  $jw$  and  $jw'$  can “see” each other, in which case the view factor is not zero. Finally, the set of equations become complete arranging Eqs. (9) and (10) for each light source element  $jl$ :

$$E_{\Delta\lambda,b,jl} = \frac{1}{\varepsilon_{\Delta\lambda,jl}} \left( Q_{\Delta\lambda,o,jl} - \sum_{jd} F_{jl-jd} Q_{\Delta\lambda,o,jd} - \sum_{jw} F_{jl-jw} Q_{\Delta\lambda,o,jw} \right) \quad (19)$$

where  $E_{\Delta\lambda,b,jl} = e_{\Delta\lambda,b,jl} / q_{i,specified}^{(l)}$  is the dimensionless emissive luminous power of light source element  $jl$  in band  $\Delta\lambda$ .

#### 4. Solution procedure

For the illumination inverse design of enclosure having non-gray surfaces, an important difficulty arises in that only the total incident luminous flux on the design surface is known, but not its value on each band  $\Delta\lambda$ . To allow the application of the inverse design technique, the incident luminous in each band is first guessed by assuming that the same amount of luminous energy is distributed in each band  $\Delta\lambda$ , that is:

$$Q_{\Delta\lambda,i,jd} = \frac{Q_{i,jd}}{N_{\Delta\lambda}} \quad (20)$$

where  $N_{\Delta\lambda}$  is the number of bands.

At this point, the approach proposed in Schneider and França (2004) for the entire visible wavelength region can be applied for each band  $\Delta\lambda$ . The outgoing luminous flux on the elements located on the walls are initially neglected in Eq. (17). Therefore, the unknowns in Eq. (17) are only the outgoing luminous fluxes on the light source elements,  $Q_{\Delta\lambda,o,jl}$ . Once Eq. (17) is written for each of the  $M$  elements that form the design surface, a system with  $M$  equations will be formed. The unknowns are the outgoing radiative heat fluxes on the  $N$  light source elements. Therefore, one aspect of the inverse design arises: the number of equations and the number of unknowns are not necessarily the same, unless  $M = N$ . In addition, since the problem corresponds to a discrete form of an integral equation of the first-kind, the system of equations is ill-conditioned, which require the application regularization methods, like the Tikhonov method, the conjugate gradient, the truncated singular value decomposition (TSVD), the modified TSVD, among others. In this solution, the TSVD method is chosen, as will be discussed later on.

Although a system of equations can be assembled in such a way to include the equations for the elements on the walls, an aspect of interest in the aforementioned procedure is that the elements in the design surface, where two conditions are imposed, are directly linked to the light source elements, where no condition is imposed. Thus, the ill-



posed part of the problem is separated from the remaining of the problem, allowing an isolated application of the appropriate methods of solution of ill-posed system of equations.

The solution is achieved by means of the following procedure. First, the outgoing luminous flux on each design surface element  $jd$  is computed by means of Eq. (15). The outgoing luminous flux on each wall element,  $Q_{\Delta\lambda,o,jw}$ , is initially set equal to zero. Then, the system of equations formed by Eqs. (17) is solved for the outgoing radiative heat fluxes on the light sources,  $Q_{\Delta\lambda,o,jl}$ . Next, Eq. (18) is applied to each wall element  $jw$  to form a (well-conditioned) system of equations on the outgoing luminous flux on the wall elements,  $Q_{\Delta\lambda,o,jw}$ . Once this system is solved, the newly computed  $Q_{\Delta\lambda,o,jw}$  are inserted into the system of equations formed by Eqs. (17), and the procedure is repeated until convergence is achieved. Finally, with the converged values of the outgoing luminous fluxes, Eq. (19) is applied to solve for  $E_{\Delta\lambda,b,jl}$ , the dimensionless emissive luminous power of the light source element  $jl$  in band  $\Delta\lambda$ .

However, the above solution is valid only for the guessed distribution of the luminous incident flux on the design surface for each band  $\Delta\lambda$ , as given by Eq. (20), but there is no guarantee that the distribution is correct. To improve the guess, the first step is to note that, for each light source element  $jl$ ,  $N_{\Delta\lambda}$  values of  $E_{\Delta\lambda,b,jl}$  are obtained, one for each band  $\Delta\lambda$ . Solving for the temperature associated with each band, the temperatures will not be necessarily the same, which is clearly not consistent, since a single temperature has to be associated with each light source element. To correct this result, the total emissive luminous power of each light source element  $jl$  is computed from:

$$E_{b,jl} = \sum_{\Delta\lambda} E_{\Delta\lambda,b,jl} \quad (21)$$

and a single temperature of the light source element can be computed by finding the temperature that will lead to the total emissive luminous power given by the summation of Eq. (21). This temperature will be an energy-based average of the temperatures obtained for each band  $\Delta\lambda$ .

At this point of the solution, an average temperature for each light source is obtained. A forward solution can be run where the light sources temperatures is known and the condition of null emissive luminous power is imposed to the design surface and to the walls. The solution of this problem leads to the incident luminous flux on each design surface element for each band  $\Delta\lambda$ , indicated by  $Q'_{i,jd}$ , so that the total incident luminous flux obtained by this procedure is:

$$Q'_{i,jd} = \sum_{\Delta\lambda} Q'_{\Delta\lambda,i,jd} \quad (22)$$

Note that  $Q'_{i,jd}$  is not necessarily equal to the imposed value, equal to 1.0, since it resulted from the distribution guessed by Eq. (20). So a new distribution of incident luminous energy in the bands can be proposed so that:

$$Q_{\Delta\lambda,i,jd} = \left( \frac{Q'_{\Delta\lambda,i,jd}}{Q'_{i,jd}} \right) Q_{i,jd} \quad (23)$$

The procedure is repeated until the distribution of the incident luminous flux in the bands for each design surface element does not vary any more.

#### 4.1 Regularization of the system of equations

The procedure discussed above involves the solution of a system of linear equations on the outgoing luminous fluxes on the light sources, as formed by Eqs. (17), which can be represented by:

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{b}(\mathbf{x}) \quad (24)$$

where matrix  $\mathbf{A}$  is formed by the view factors between the design surface and the light source elements,  $F_{jd-jl}$ ; vector  $\mathbf{x}$  represents the unknown outgoing luminous fluxes on the light sources,  $Q_{\Delta\lambda,o,jl}$ ; and vector  $\mathbf{b}$  contains the terms on the right-hand side of Eq. (17), which are dependent on  $\mathbf{x}$ .

The system formed by Eqs. (17) is expected to present the characteristics of ill-posed problems. In general, the components of the exact solution vector  $\mathbf{x}$  present steep oscillations between very large positive and negative numbers, and small perturbations cause a much amplified change in the solution. One solving an ill-posed problems should not aim at an exact solution, but rather to impose additional constraints to reduce the size (norm) of  $\mathbf{x}$ , and achieve a smooth

solution. However, the greater the smoothness imposed on the solution is, the greater will be the residual. This is the basic idea behind any regularization method for the solution of ill-posed problems.

Among the regularization procedures, the truncated singular values decomposition (TSVD) is employed here. First, matrix  $\mathbf{A}$  is decomposed into three matrices:

$$\mathbf{A} = \mathbf{U} \cdot \mathbf{W} \cdot \mathbf{V}^T \quad (25)$$

where  $\mathbf{U}$  and  $\mathbf{V}$  are orthogonal matrices, and  $\mathbf{W}$  is a diagonal matrix formed by the singular values  $w_j$ . As a consequence, the solution vector  $\mathbf{x}$  can be computed by:

$$\mathbf{x} = \sum_{j=1}^N \left( \frac{b_k \cdot u_{kj}}{w_j} \right) \mathbf{v}_j \quad (26)$$

where  $N$  stands for the number of unknowns (in this case, the number of light source elements).

In ill-posed problems, the singular values  $w_j$  decay continuously to very small values. Since they appear in the denominator of Eq. (26), the components of  $\mathbf{x}$  can present very large absolute numbers. However, the smaller the singular value  $w_j$  is, the closer the corresponding vector  $\mathbf{v}_j$  is to the null-space of  $\mathbf{A}$ . In other words, the terms related to the smaller singular values can be eliminated from Eq. (26) without introducing a large error into the solution. This is the main idea of the TSVD: only the terms related to the  $p$ -th largest singular values are kept in Eq. (26), instead of all  $N$  terms. The solution is the vector  $\mathbf{x}$  with the smallest norm subjected to minimum deviation  $|\mathbf{A} \cdot \mathbf{x} - \mathbf{b}|$ .

Another aspect of the inverse solution concerns the number of unknowns and the number of equations. The number of equations equals the number of unknowns only when the number of light source elements and the number of design surface elements are the same. In general, this is not always verified, as in the example cases to be shown later in this work. An important feature of the TSVD method is that it can also be applied to the situation where the numbers of unknowns and equations are not the same, as will be shown in the results section.

## 4.2 Verification of the solution

Due to the need of regularizing the system of equations, an exact solution is not expected. The following procedure is used for the verification of the solution. After the inverse solution is run to convergence, a forward problem is solved where the dimensionless emissive luminous powers on the light source elements are known, from the inverse solution, and only the condition null emissive luminous power is imposed on the design surface and on the wall elements. The incident luminous flux on each element design surface  $jd$  is then calculated and compared to the specified heat flux by:

$$\gamma_{jd} = \left| \frac{Q_{specified} - Q_{i,jd}}{Q_{specified}} \right| \quad (27)$$

where  $Q_{specified}$  is the specified dimensionless incident luminous flux, equal to 1.0, and  $Q_{i,jd}$  is the incident luminous flux resulting from the emissive luminous powers on the light sources that were obtained from the inverse solution. Once  $\gamma_{jd}$  is calculated for each element  $jd$  in the design surface, the arithmetic average and the maximum errors  $\gamma_{avg}$  and  $\gamma_{max}$  can be readily found.

## 5. Results and discussion

The case considered in this work consists of a three-dimensional enclosure as shown in the schematic of Fig. 1. The aspect ratio of the enclosure is  $W/L = 0.8$ , and the dimensionless height is  $H/L = 0.2$ . The selection of the other dimensions of the enclosure will require a few considerations. First, the design surface ought not to cover the entire extension of the base, since the portions close to the corners would be mainly affected by the reflections from the side walls, not from the luminous radiation from the light source elements on the top surface. Therefore, the design surface dimensions are taken as  $L_d/L = 0.8$  and  $W_d/L = 0.6$ . The locations of the light sources will be proposed later on.

For the design surface elements, the luminous emissive power is zero and the dimensionless incident luminous flux is  $Q_{i,jd} = 1.0$ ; for the wall elements, the luminous emissive power is also zero; the light source elements are unconstrained. As an example case of the application of the methodology, the wavelength spectrum in the visible region is divided into three bands. The hemispherical emissivities and its dependence to the wavelength are shown in Fig. 3 for design, wall and light source elements. Considering that the light sources are formed by incandescent tungsten filaments, they can be modeled as gray surfaces having hemispherical emissivity  $\epsilon_{jl} = 0.36$ . The problem is at this point completely defined unless for the location of the light source elements, which will be discussed next.

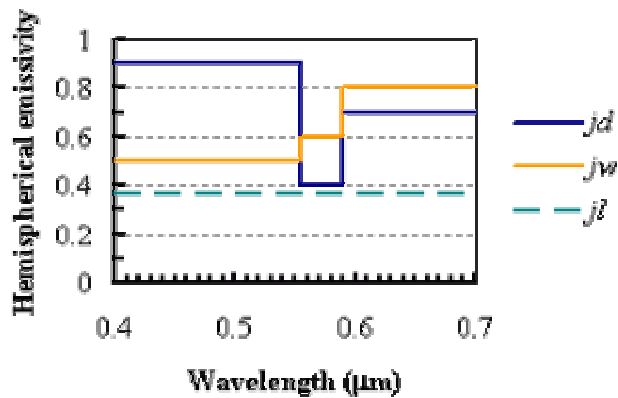


Figure 3. Spectral distribution of the hemispherical emissivity for the three types of surface elements. band b1 ( $0.40 \mu\text{m} < \lambda < 0.55 \mu\text{m}$ ), band b2 ( $0.55 \mu\text{m} < \lambda < 0.59 \mu\text{m}$ ), band b3 ( $0.59 \mu\text{m} < \lambda < 0.70 \mu\text{m}$ ).

### 5.1 Case 1: light sources covering the entire top surface

To illustrate the inverse design, it is first considered that the light source elements cover the entire top surface. Figure 4 shows the locations of both the light source elements (circular dots) and the design surface elements (shaded area). Due to symmetry, indicated by the dashed lines, only a quarter of the domain is solved:  $0 \leq x/L \leq 0.5$ ,  $0 \leq y/L \leq 0.4$ . One consequence of having the light sources covering the entire top surface is that the numbers of design surface and light source elements (indicated by  $M$  and  $N$ , respectively) are not the same. Selecting a grid size of  $\Delta x/L = 1/30$ , one finds  $M = 108$  and  $N = 180$ . Thus, the system of equations formed by Eq. (17) is composed by  $M = 108$  equations and  $N = 180$  unknowns.

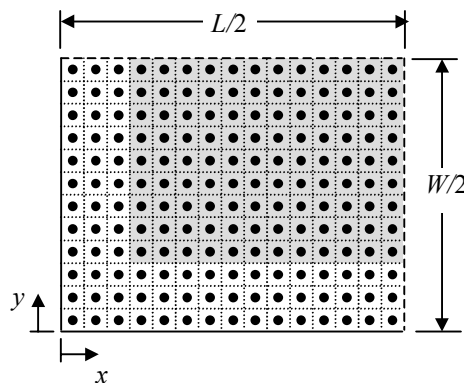


Figure 4. Locations of the design surface (shaded area) and light source elements (circular dots) in one quarter of the bottom and top surfaces for case 1: light sources covering the entire top surface. Dashed lines indicate symmetry.

The procedure presented in Section 4 is then applied to find the required net luminous flux and temperature of the light source elements. Figure 5 shows the singular values of matrix  $\mathbf{A}$ , under the label Case 1. As typical of inverse design problems, the singular values  $w_j$  decay steadily to values that are as small as  $10^{-10}$ . It should be noted that in the figure only the non-zero singular values are shown; for  $j > M = 108$ , all the singular values are zero. This indicates that the problem presents infinite exact solutions. To select the exact solution with the smallest norm, one needs only to apply the TSVD method to retain only the  $M$  first terms of the series of Eq. (26). In this case, it was observed that the components of vector  $\mathbf{x}$  had very large absolute values, with alternating positive and negative signals, due the small singular values that arise in the denominator of the series terms. Since the components of vector  $\mathbf{x}$  correspond to the outgoing luminous fluxes on the light sources, positive numbers, this exact solution has no physical meaning. However, the TSVD can be further explored to keep only the terms of the series corresponding to the  $p$  ( $< M$ ) largest singular values. In this case, the obtained regularized solution will be no longer exact, but can present an acceptable behavior.

Figure 6 presents examples of regularized solutions for three different regularization parameters  $p = 12, 10$  and  $8$ . Solutions for  $p \geq 13$  presented negative values for the emissive luminous power on some of the light sources, which is not physically acceptable. Solutions for  $p = 11, 9$  and  $6$  were also obtained, but are not presented here since they did not bring additional features to those shown in Fig. 6. As can be seen in all cases, all the solutions present an oscillatory

behavior that are in fact reminiscent of the steep oscillations of the smallest-norm exact solution (that is, for  $p = M = 108$ ). In general, the smaller the value  $p$  is, the smoother the solution will be, although at the expense of presenting a larger error. This can be verified in Table 1, which presents the maximum and average errors for different values of  $p$ , as defined in Section 4.2, as well the minimum singular value retained in the series of Eq. (26),  $w_{j,\min}$ .

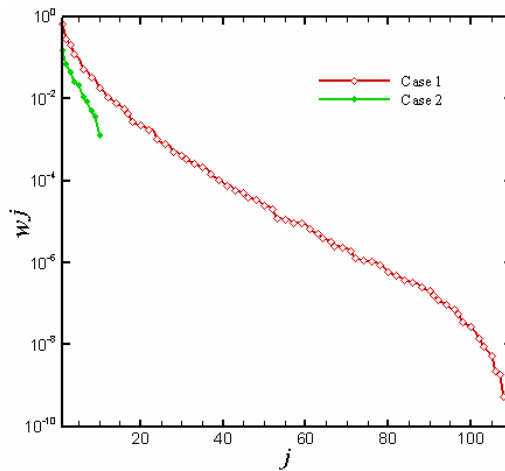


Fig. 5 – Singular values of matrix A: cases 1 and 2.

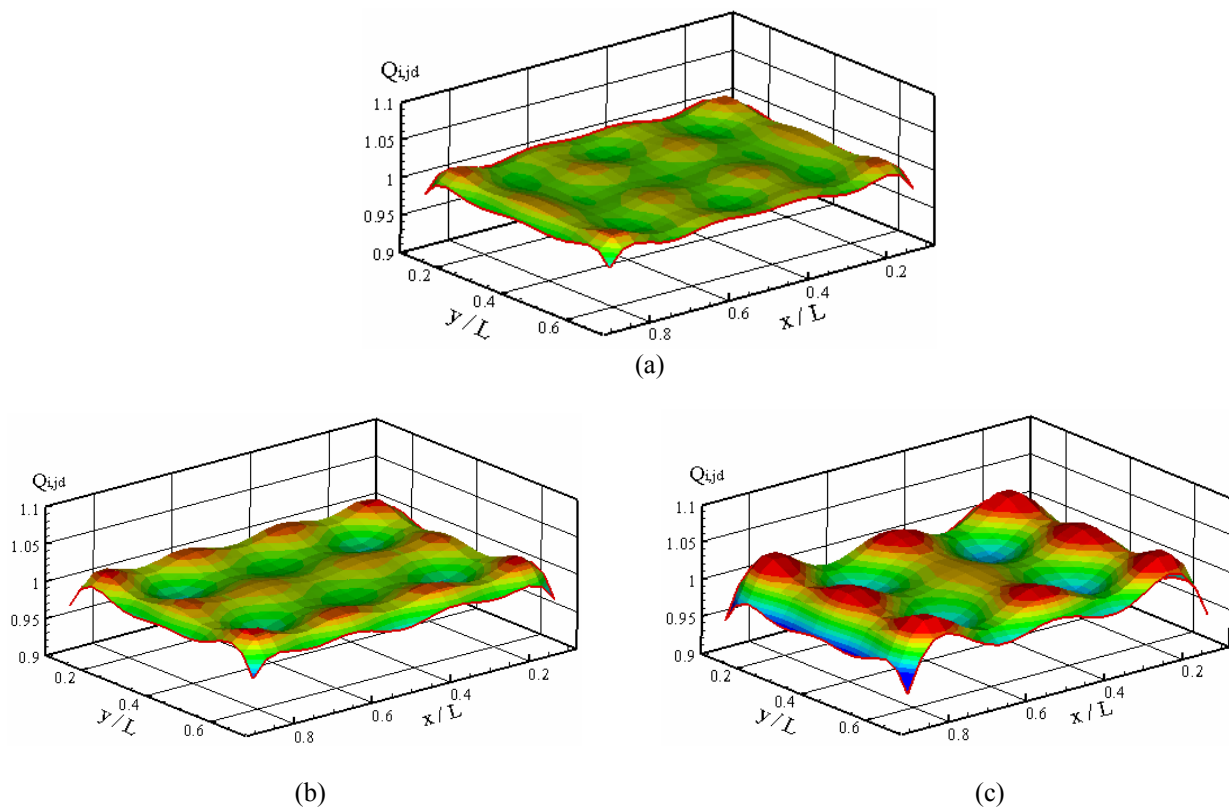


Fig. 6 – Incident luminous flux distribution on the design surface for different regularization parameters  $p$ : (a)  $p = 12$ ; (b)  $p = 10$ ; (c)  $p = 8$ . Case 1.

Table 1 – Minimum singular value and inverse solution errors for different regularization parameters  $p$ .

$p$	$w_{j,\min}$	$\gamma_{\text{avg}}$ (%)	$\gamma_{\text{max}}$ (%)
12	$1.1154 \times 10^{-2}$	0.4353	2.1692
10	$1.7734 \times 10^{-2}$	0.7173	3.2396
8	$3.3388 \times 10^{-2}$	1.3182	5.5417

### 5.2 Case 2: reduced number of light sources

The solutions obtained and discussed in the previous section illustrate the typical fact that the inverse design technique can lead to a number of approximate solutions, where the final selection can take into account the accuracy and practicality of the solutions. One aspect of the light source configuration of Fig. 4 is that a very large number of independent light sources are being required (that is,  $4 \times 180 = 720$  elements for the entire enclosure), which is not practical. The designer will probably be much more interested on having a smaller amount of heat sources. For that matter, the previously selected solution (for  $p = 12$ ) provide a hint towards this goal. Figure 7 illustrates a design where ten light sources are distributed on the entire top surface. Note that as expected the solution is symmetrical for each quarter of the domain. The locations of the light sources were proposed to approximately coincide with the points of maximum luminous fluxes for the solution with  $p = 12$ .

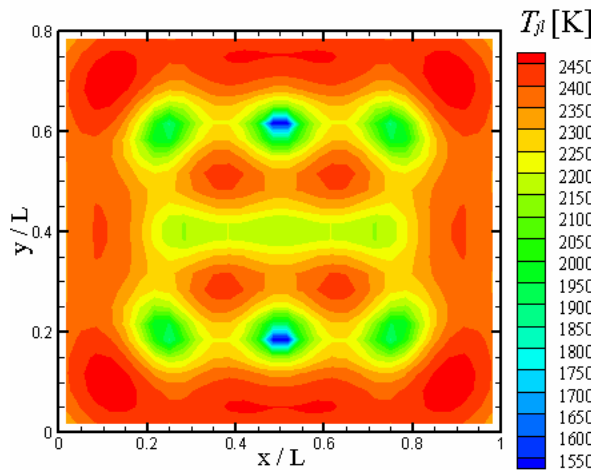


Fig. 7 – Required lamp filament temperature distribution on the top surface for the solution with  $p = 12$ . Case 1.

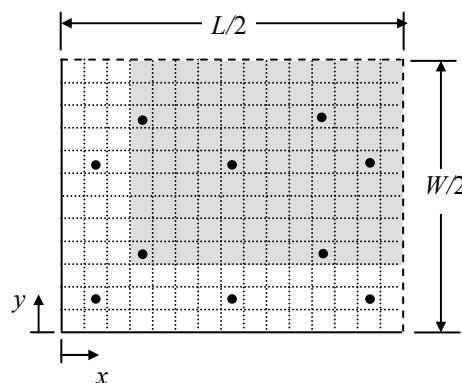


Fig. 7 – Locations of the design surface (shaded area) and light source elements (circular dots) in one quarter of the bottom and top surfaces for case 2: reduced number of light sources. Dashed lines indicate symmetry.

Since the number of elements on the design remains the same,  $M = 108$ , and the number of light sources is  $N = 10$ , the problem is overspecified: the number of equations  $M$  is larger than that of unknowns. It is well known that such a problem can only be solved with some approximation. Figure 5 presents the singular values of matrix  $\mathbf{A}$  for this case, under the label of Case 2. As seen, the ten singular values decay to a value in the order of  $10^{-3}$ . Keeping all the singular values in the series of Eq. (26) provides the smallest norm vector  $\mathbf{x}$  that leads to the smallest norm  $|\mathbf{A} \cdot \mathbf{x} - \mathbf{b}|$ . The solution for  $p = 4$  is shown in Table 2, which indicates the net luminous flux ( $Q_{r,jl} = Q_{o,jl} - Q_{i,jl}$ ) and the temperature on each element shown in Fig. 7. In the table, the location of the light sources are indicated by indices  $i$  and  $j$  to designate the position in  $x$  and  $y$  directions, respectively. The luminous flux on the design surface is shown in Fig. 8. As seen the solution is within the interval  $-0.90 < Q_{i,jd} < -1.10$ : the maximum and average error of the solution are 9.9783 % and 2.9458 %. If an error less than 10.0 % is acceptable, then the proposed solution is acceptable. Otherwise a larger number of singular values should be considered, until the solution start generating negative emissive power for light sources. Note that better solutions can be also achieved by the choice of other locations of the light sources.

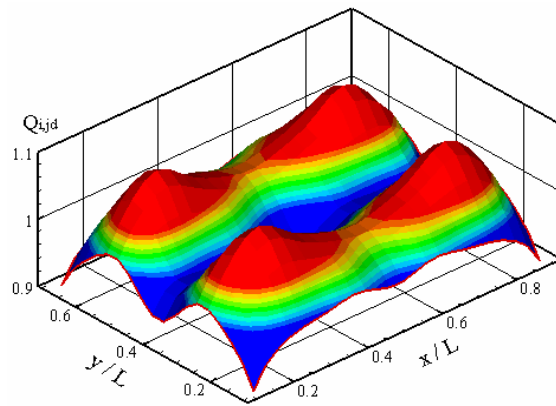


Fig. 8 – Dimensionless incident luminous flux on the design surface for the solution in case 2.

Table 2 – Required dimensionless net luminous on the light source elements for case 2.

$jl$	$i$	$j$	$Q_{r,jl}$	$T_{jl}$ (K)	$jl$	$i$	$j$	$Q_{r,jl}$	$T_{jl}$ (K)
1	2	2	12.8992	2978.17	6	8	8	35.7926	3378.41
2	2	8	18.6702	3111.76	7	12	4	22.2477	3179.28
3	4	4	26.7419	3253.13	8	12	10	17.0751	3078.80
4	4	10	23.2914	3197.36	9	14	2	13.5647	2996.33
5	8	2	23.2046	3195.99	10	14	8	14.0683	3009.22

Figure 9 shows the amount of energy in each of the three bands shown in Fig. 3. The sum of the energy in each band leads to the total incident irradiation of Fig. 8. Incandescent light sources filaments with temperatures in the order of 3000 K (see Table 2) have a luminous efficiency of about 5%. In this temperature level, most of the thermal radiation is concentrated in the infrared region ( $\lambda > 0.70\mu\text{m}$ ). Therefore, band b3 ( $0.59\mu\text{m} < \lambda < 0.70\mu\text{m}$ ), located in the larger wavelength region of the visible light interval, and closer to infrared region, contributes with the greater amount of energy. It follows that the contribution of band b2 ( $0.55\mu\text{m} < \lambda < 0.59\mu\text{m}$ ) is also more significant than band 1 ( $0.40\mu\text{m} < \lambda < 0.55\mu\text{m}$ ), despite occupying a smaller region of the spectrum.

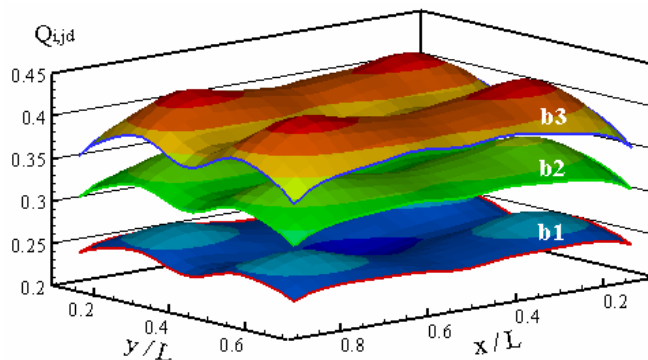


Fig. 9 – Dimensionless incident luminous flux for each spectral band considered for the design surface. Case 2.

All the above solutions involved a discretization of the spatial domain such that  $\Delta x/L = 1/30$ , which proved to be sufficiently small to guarantee grid independence. It should be pointed that, for greater generality, all the results were presented in dimensionless form, but conversion dimensional terms should be straightforward. For instance, the luminous power (in lumens) on each light source element of Table 2 can be determined as  $Q_{r,jl} \cdot q_{\text{specified}}^{(l)} \cdot (\Delta x)^2$ . Convergence with respect to the correction of the incident energy in the bands was achieved for all cases with four iterations, leading to a maximum variation of  $10^{-6}$  between the results of the two last steps.

## 6. Conclusions

This work considered an inverse design problem of a non-gray enclosure in which the luminous fluxes on the light source elements were determined to satisfy the specified uniform incident luminous flux on the design surface. The numerical discretization of the problem led to an ill-conditioned system of non-linear equations. Ill-conditioned systems of equations require the application of regularization methods to recover physically acceptable solutions. In this work, it was employed the truncated singular valued decomposition (TSVD), which eliminates the linear combinations related to the smallest singular values of the system, keeping only the  $p$  largest ones. The non-linearity of the problem arose from the surfaces being non-gray, in which case the amount of energy in the spectral bands that cover the visible wavelength region is unknown.

The problem was formulated from the application of the luminous energy conservation in all the bands where the spectral emissivities could be assumed uniform. The example case consisted of three bands. The proposed procedure set a system of equations relating the design surface elements directly to the light source elements, calculating the outgoing luminous fluxes on the wall elements from the conditions of the previous iterative step. The unknown distribution of the incident energy on the design surface in the spectral bands were initially guessed, and then the inverse solution was run to find an average temperature on the light sources, from which a new distribution of the incident energy were used as a new input, repeating the process until convergence was reached. In most cases, four steps were enough to assure convergence within a relative error of  $10^{-6}$ .

Two design cases were studied. In case 1, it was considered that the light source elements covered the entire top surface. The TSVD method was applied to find the solution for different regularization parameters, resulting in a few solutions that were evaluated according to physical requirements and the error of the solution. In case 2, it was proposed an inverse design in which only ten light source elements were set on the top surface (case 2). Despite the problem becoming overspecified, it was possible to obtain a solution more practical than having light sources covering the entire top surface, and still presenting an average and a maximum error less than 2.0 % and 10.0 %, respectively. It would be very difficult to arrive at such a solution by a trial-and-error procedure of the conventional forward technique.

## 7. Acknowledgments

The authors thank CAPES (Brazil) for the support under the program CAPES/UT-AUSTIN, No. 06/02.

## 8. References

- Boast W.B, 1953, *Illumination Engineering*, McGraw-Hill, New York..
- França, F., Howell, J., Ezekoye, O., and Morales, J. C., 2003, "Inverse Design of Thermal Systems," *Advances in Heat Transfer*, J. P. Hartnett and J. P. Irvine, eds., 36, Elsevier, pp. 1-110.
- Hansen, P. C., 1990, *Truncated SVD Solutions to Discrete Ill-Posed Problems With Ill-Determined Numerical Rank*, SIAM J. Sci. Statist. Comput., 11, pp. 503-518.
- Harrison, W. and Anderson, E. A., 1916, "Illumination Efficiencies as Determined in an Experimental Room," *Trans. Illum. Eng. Soc.*, **11**, pp 67-91.
- Harrison W. and Anderson E.A., 1920, Coefficients of Utilization, *Trans. Illum. Eng. Soc.*, **15**, pp 97-123.
- Mark, S., 2000, *The IESNA Lighting Handbook*, Illuminating Engineering Society of North America, New York.
- Moon P., 1941, "Interreflections in Rooms," *J. Optical Soc. Am.*, **31**, pp 374-382.
- Moon, P., and Spencer, D. E., 1946a, "Light Distribution in Rooms," *J. Franklin Inst.*, **242**, pp 111-141.
- Moon, P., and Spencer, D. E., 1946b, "Light Design by the Interreflection Method," *J. Franklin Inst.*, **242**, pp 465-501.
- Schneider, P. S., and França, F. H. R., 2004, "Inverse Analysis Applied to an Illumination Design", *Proceedings of the 10th Brazilian Congress of Thermal Sciences and Engineering*, Rio de Janeiro, Brazil.
- Siegel, R., and Howell, J. R., 2002, *Thermal Radiation Heat Transfer*, 4th Ed. Hemisphere Publishing Corporation, Washington.

## 9. Copyright notice

The authors are the only responsible for the material included in their paper.

# SOLUÇÃO NUMÉRICA DE PROBLEMA TERMOELÁSTICO COM ESTIMATIVA DO ERRO DE DISCRETIZAÇÃO

**Orestes Hacke**

Universidade do Contestado (UnC), Mafra, SC  
oresteshacke@brturbo.com

**Carlos Henrique Marchi**

Departamento de Engenharia Mecânica, Universidade Federal do Paraná (UFPR), Curitiba, PR  
marchi@demec.ufpr.br

**Resumo.** Neste trabalho, verifica-se o erro numérico da solução de um problema de condução de calor e estima-se o erro de um problema termoelástico; ambos bidimensionais. As soluções numéricas são obtidas empregando-se o método de diferenças finitas com aproximações numéricas de segunda ordem de acurácia. São avaliados o erro ( $E$ ) verdadeiro de discretização e sua estimativa ( $U$ ) além das ordens efetiva e aparente.  $U$  é obtido com os estimadores de Richardson e GCI. São utilizadas malhas uniformes com até 1024x1024 elementos. Verificou-se que o estimador GCI é confiável e as ordens efetiva e aparente tendem à ordem assintótica, prevista a priori, quando o tamanho dos elementos tende a zero.

*Palavras-chave:* diferenças finitas, GCI, extrapolação de Richardson, condução de calor, termoelasticidade.

## 1. Introdução

Com relação às estimativas dos erros numéricos envolvidos nos resultados de simulações numéricas, atualmente os trabalhos encontrados na literatura podem ser classificados em quatro conjuntos: (1) nenhuma estimativa é realizada e a solução numérica é obtida sobre uma única malha; (2) nenhuma estimativa é realizada mas são apresentadas soluções numéricas obtidas sobre duas ou mais malhas, geralmente fazendo-se comparações gráficas de perfis de variáveis de campo nas diversas malhas; (3) são feitas estimativas mas com base em estimadores de erro pouco confiáveis ou inadequados, como o estimador delta (Roache, 1998); e (4) são feitas estimativas com base no estado-da-arte, isto é, com os melhores estimadores de erro disponíveis, como o estimador GCI (Grid Convergence Index) (Roache, 1994).

A magnitude aceitável para o erro numérico é função, entre outros fatores, da finalidade da solução numérica, dos recursos financeiros envolvidos, do tempo permitido ou disponível para realizar as simulações e dos recursos computacionais existentes. Sabendo-se que as soluções numéricas contêm erros, entre outros motivos, é importante estimá-los porque quando o erro é maior do que o aceitável compromete-se a confiabilidade do uso da solução numérica.

Neste trabalho é resolvido um problema de condução de calor e um de termoelasticidade linear, ambos com o método de diferenças finitas, com malhas uniformes e aproximações numéricas de 2<sup>a</sup> ordem de acurácia. O modelo numérico segue basicamente o de Demirdzic e Martinovic (1993), que empregaram o método de volumes finitos em problemas termoelásticos, entre outros. Porém, eles não fizeram estimativa do erro de discretização. Assim, os objetivos do presente trabalho são: (1) verificar (Roache, 1998) as soluções numéricas para o problema térmico cuja solução analítica é conhecida; (2) obter soluções numéricas altamente acuradas, com malhas de até 1024x1024 elementos; (3) utilizar os estimadores de Richardson e GCI para estimar o erro de discretização das variáveis de interesse; (4) avaliar o desempenho destes estimadores para o caso em que a solução analítica é conhecida; e (5) comprovar a ordem de acurácia das soluções numéricas.

O trabalho está assim dividido: na seção 2 são apresentados os estimadores de erro; na seção 3 é definido o problema térmico, seu modelo numérico e apresentados os resultados; na seção 4, o mesmo para o problema termoelástico; e na seção 5, a conclusão do trabalho.

## 2. Erro de discretização

Erro numérico ( $E$ ) é a diferença entre a solução analítica exata ( $\Phi$ ) de uma variável de interesse e a sua solução numérica ( $\phi$ ), isto é, (Marchi e Silva, 2002)

$$E(\phi) = \Phi - \phi \quad (1)$$



onde  $E$  é causado por quatro fontes de erros: truncamento, iteração, arredondamento e programação. Quando as outras fontes são inexistentes ou muito pequenas em relação aos erros de truncamento,  $E$  também pode ser denominado de erro de discretização.

Em situações práticas, uma solução numérica é obtida porque a solução analítica é desconhecida. Por consequência, o valor verdadeiro do erro numérico também é desconhecido. Portanto, o erro numérico tem que ser estimado. Pelo número de citações e amplo uso que vem sendo feito dele, e segundo a experiência de um dos autores deste trabalho, o  $GCI$  (*Grid Convergence Index*) de Roache (1994) pode ser considerado o mais confiável dos estimadores atuais para erros de discretização. Segundo o  $GCI$ , o erro de discretização estimado ( $U$ ) é dado por

$$U_{GCI}(p, \phi_1) = F_s \frac{|\phi_1 - \phi_2|}{(r^p - 1)} \quad (2)$$

onde

$$p = \text{Min}(p_L, p_U > 0) \quad (3)$$

$$p_U = \frac{\log\left(\frac{\phi_2 - \phi_3}{\phi_1 - \phi_2}\right)}{\log(r)} \quad (4)$$

$$r = \frac{h_3}{h_2} = \frac{h_2}{h_1} \quad (5)$$

$\phi_1$ ,  $\phi_2$  e  $\phi_3$  = soluções numéricas obtidas respectivamente com malhas fina ( $h_1$ ), grossa ( $h_2$ ) e supergrossa ( $h_3$ ),  $h$  = tamanho dos elementos da malha, isto é, a distância entre dois nós consecutivos,  $r$  = razão de refino de malha,  $\text{Min}$  = valor mínimo entre os argumentos,  $F_s$  = fator de segurança (três, neste trabalho),  $p_L$  = ordem assintótica (Roache, 1998) do erro prevista para cada variável de interesse,  $p_U$  = ordem aparente (De Vahl Davis, 1983; Marchi e Silva, 2002) do erro calculada para cada variável de interesse.

A Eq. (2) resulta dos trabalhos de Roache (1994) e Marchi e Silva (2002). O cálculo da ordem  $p$  do erro, segundo a Eq. (3), aumenta a confiabilidade do erro estimado pela Eq. (2). Se  $p_U \leq 0$  ou indefinido, a Eq. (2) não deve ser aplicada (Marchi, 2001).

Teoricamente (Marchi, 2001), espera-se que  $p_E$  e  $p_U \rightarrow p_L$  para  $h \rightarrow 0$ . Isto é, espera-se que as ordens práticas ( $p_E$  e  $p_U$ ), que são calculadas com as soluções numéricas de cada variável de interesse, tendam à ordem teórica ( $p_L$ ), prevista *a priori*, quando o tamanho dos elementos da malha ( $h$ ) tende a zero.

A ordem efetiva ( $p_E$ ) do erro verdadeiro é definida por (Marchi, 2001)

$$p_E = \frac{\log\left[\frac{E(\phi_2)}{E(\phi_1)}\right]}{\log(r)} \quad (6)$$

Conforme a Eq. (6), a ordem efetiva ( $p_E$ ) é função do erro verdadeiro da variável de interesse. Assim, para os problemas cuja solução analítica é conhecida, ela pode ser usada para verificar *a posteriori* se, à medida que  $h \rightarrow 0$ ,  $p_E \rightarrow p_L$ .

O estimador de Richardson, baseado nas ordens assintótica ( $p_L$ ) e aparente ( $p_U$ ), é dado por

$$U_{Ri}(p_L, \phi_1) = \frac{\phi_1 - \phi_2}{r^{p_L} - 1} \quad (7)$$

$$U_{Ri}(p_U, \phi_1) = \frac{\phi_1 - \phi_2}{r^{p_U} - 1} \quad (8)$$

### 3. Condução de calor bidimensional

#### 3.1. Modelo matemático

O problema da condução de calor bidimensional em regime permanente é governado pela equação de Laplace, isto é,

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \tag{9}$$

onde  $x$  e  $y$  são as direções coordenadas e  $T$  é a temperatura. As condições de contorno são do tipo Dirichlet e dadas por:  $T(x,0) = T(0,y) = T(1,y) = 0$  e  $T(x,1) = \text{sen}(\pi x)$ .

As variáveis de interesse, isto é, as variáveis para as quais é obtida a solução numérica e verificado seu erro de discretização e suas ordens são:

- (a) Temperatura ( $T$ ) em  $x = y = 3/4$ : variável dependente, primária, local, incógnita principal do problema que é obtida a partir da solução da Eq. (9), que é uma equação diferencial parcial (EDP), linear, de segunda ordem. Sua solução analítica é

$$T(x, y) = \text{sen}(\pi x) \frac{\text{senh}(\pi y)}{\text{senh}(\pi)} \tag{10}$$

- (b) Taxa de transferência de calor ( $q$ ): variável secundária, obtida a partir da seguinte definição

$$q = -\int_0^1 k W \left( \frac{\partial T}{\partial y} \right)_{x,y=1} dx \tag{11}$$

onde  $k$  = condutividade térmica do material (constante),  $W$  = espessura do material (constante). Para a Eq. (10), a solução analítica da Eq. (11) é

$$q = -2kW \coth(\pi) \tag{12}$$

### 3.2. Modelo numérico

O modelo numérico é caracterizado pelo uso do método de diferenças finitas, malhas uniformes e o método de Gauss-Seidel (Kreyszig, 1999) para resolver o sistema de equações. A linguagem de programação usada é *C++ Builder 6.0*, com precisão dupla para todas as variáveis reais. Os dois termos da Eq. (9) foram aproximados com o esquema de diferença central (CDS) (Tannehill *et al.*, 1997). O processo iterativo foi levado até ser atingido o erro de máquina para minimizar o erro de iteração e utilizou-se estimativa inicial nula. A variável taxa de transferência de calor ( $q$ ) foi obtida com o esquema de diferença à montante de segunda ordem (UDS-2) (Tannehill *et al.*, 1997). Considerando-se as aproximações numéricas acima, a ordem assintótica prevista para cada variável de interesse é  $p_L = 2$ .

### 3.3. Resultados

Para obter as soluções numéricas deste trabalho foi utilizado um computador com processador Pentium 4, 2.8 GHz e 512 MB de memória RAM. Considerou-se  $k = 1$  W/m.K e  $W = 1$  m. As malhas utilizadas são uniformes com 2, 4, 8, ... até 1024 elementos em cada direção. Para a malha mais refinada, o número de iterações para atingir o erro de máquina foi de  $\approx 2,8 \times 10^6$ , que resultou no tempo de CPU de quase três dias (71 horas).

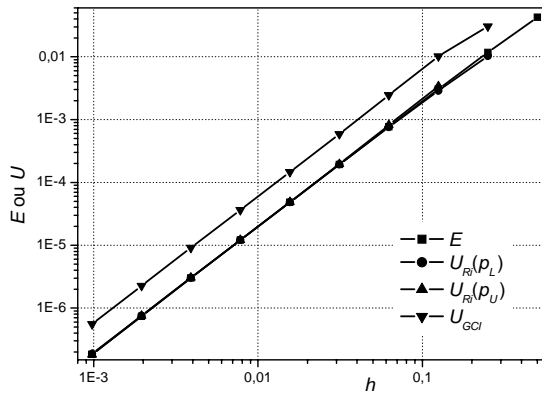
A Tab. 1 apresenta a solução analítica e numérica para as duas variáveis de interesse, para a malha 1024x1024 elementos. Nesta malha, para as duas variáveis, o estimador  $GCI$  é confiável. Isto é, a solução analítica está contida no intervalo compreendido pela solução numérica ( $\phi$ )  $\pm U_{GCI}$ .

Tabela 1. Solução numérica com malha 1024 x 1024 do problema térmico.

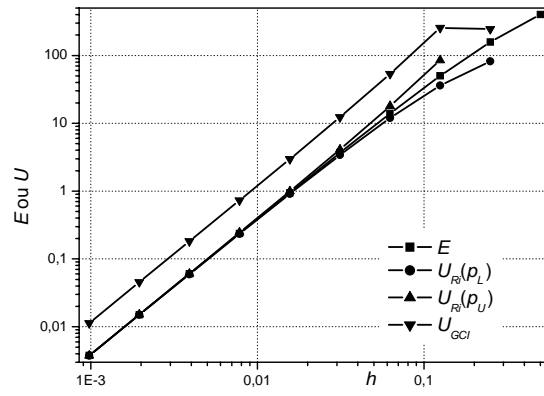
Variável	Solução analítica ( $\Phi$ )	Solução numérica ( $\phi$ ) e seu $U_{GCI}$
$T(3/4,3/4)$ [K]	0,32009852	$0,32009872 \pm 5,6 \times 10^{-7}$
$q$ [W]	-805,001	$-804,997 \pm 1,1 \times 10^{-2}$

Para as duas variáveis de interesse, a Fig. 1 apresenta o módulo do erro verdadeiro ( $E$ ), e a sua estimativa ( $U$ ) baseada nas Eqs. (2), (7) e (8). Percebe-se que o estimador  $GCI$  é confiável em qualquer malha, isto é,  $U/E \geq 1$ . Além disso,  $E$  se reduz monotonicamente à medida que  $h \rightarrow 0$ .

A Fig. 2 apresenta as ordens assintótica ( $p_L$ ), efetiva ( $p_E$ ) e aparente ( $p_U$ ) do erro. Nota-se que  $p_E$  e  $p_U \rightarrow p_L$  monotonicamente à medida que  $h \rightarrow 0$ . Para não comprometer a análise dos resultados, foram eliminados alguns pontos que, sabidamente, já sofrem efeito do erro de arredondamento; no caso em análise, isso ocorre em geral para  $h < 4 \times 10^{-2}$ . Todos os resultados de  $p_E$  e  $p_U$ , apresentados neste trabalho, foram obtidos com as Eqs. (4) e (6) com razão de refino ( $r$ ) igual a 2.

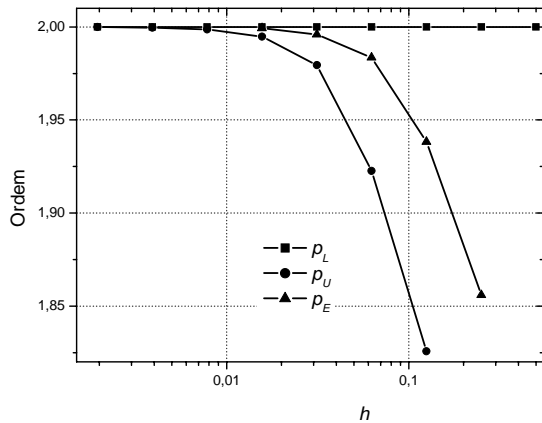


a) Temperatura  $T(3/4,3/4)$ .

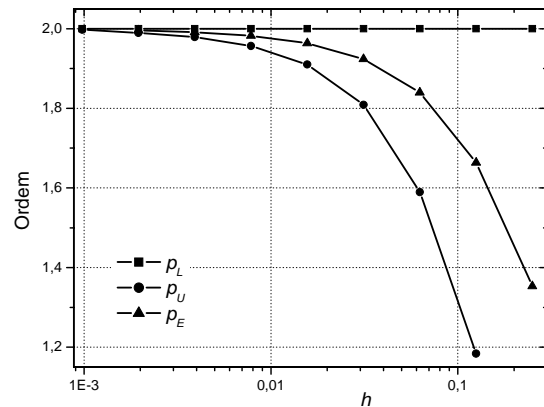


b) Taxa de transferência de calor  $q$ .

Figura 1. Condução de calor: módulo do erro ( $E$ ) e de sua estimativa ( $U$ ).



a) Temperatura  $T(3/4,3/4)$ .



b) Taxa de transferência de calor  $q$ .

Figura 2. Condução de calor: ordens assintótica ( $p_L$ ), efetiva ( $p_E$ ) e aparente ( $p_U$ ) do erro.

#### 4. Termoelasticidade bidimensional

##### 4.1. Modelo matemático

O problema termoelástico linear, bidimensional, em regime permanente é governado pelas seguintes equações (Timoshenko e Goodier, 1970):

$$C_u \cdot \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 2 \cdot C_u \cdot \beta \cdot \frac{\partial T}{\partial x} \quad (13)$$

$$C_u \cdot \frac{\partial}{\partial y} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} = 2 \cdot C_u \cdot \beta \cdot \frac{\partial T}{\partial y} \quad (14)$$

onde  $C_\mu = \frac{1+\mu}{1-\mu}$ ,  $\mu$  = razão de Poisson,  $x$  e  $y$  = direções coordenadas,  $T$  = temperatura,  $\beta$  = coeficiente de expansão térmica, e  $u$  e  $v$  = deslocamentos nas direções  $x$  e  $y$ . As condições de contorno são do tipo Dirichlet e dadas por  $u$  e  $v = 0$  em todos os quatro contornos. A parte térmica do problema é governada pela Eq. (9).

As variáveis de interesse são:

- (a) Deslocamento na direção  $x$  ( $u$ ) em  $x = y = 3/4$ : variável dependente, primária, local, incógnita principal do problema que é obtida a partir da solução das Eqs. (13) e (14), que é um sistema de equações diferenciais parciais, acopladas, de segunda ordem.
- (b) Força na direção  $x$  no contorno direito ( $F_x$ ): variável secundária, obtida a partir da seguinte definição

$$F_x = \int_0^1 W(\sigma_x)_{x=1,y} dy \tag{15}$$

onde a tensão normal  $\sigma_x$  é dada por

$$\sigma_x = \frac{E}{1+\mu} \cdot \left[ \frac{\mu}{1-\mu} \cdot \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + \frac{\partial u}{\partial x} - C\mu \cdot \beta \cdot (T - T_0) \right] \tag{16}$$

com  $E$  = módulo de Young e  $T_0$  = temperatura de referência, nula.

### 4.2. Modelo numérico

O modelo numérico é caracterizado pelo uso do método de diferenças finitas e malhas uniformes. O método de Gauss-Seidel (Kreyszig, 1999) é usado para resolver os três sistemas de equações originados da discretização das Eqs. (9), (13) e (14). A linguagem de programação usada é C++ Builder 6.0, com precisão dupla para todas as variáveis reais. Os termos das equações foram aproximados com o esquema de diferença central (CDS) (Tannehill *et al.*, 1997). O processo iterativo foi levado até ser atingido o erro de máquina para minimizar o erro de iteração e utilizou-se estimativa inicial nula para as três variáveis primitivas:  $T$ ,  $u$  e  $v$ . A variável força na direção  $x$  no contorno direito ( $F_x$ ) foi obtida com o esquema UDS-2, de segunda ordem (Tannehill *et al.*, 1997). Considerando-se as aproximações numéricas acima, a ordem assintótica prevista para cada variável de interesse é  $p_L = 2$ .

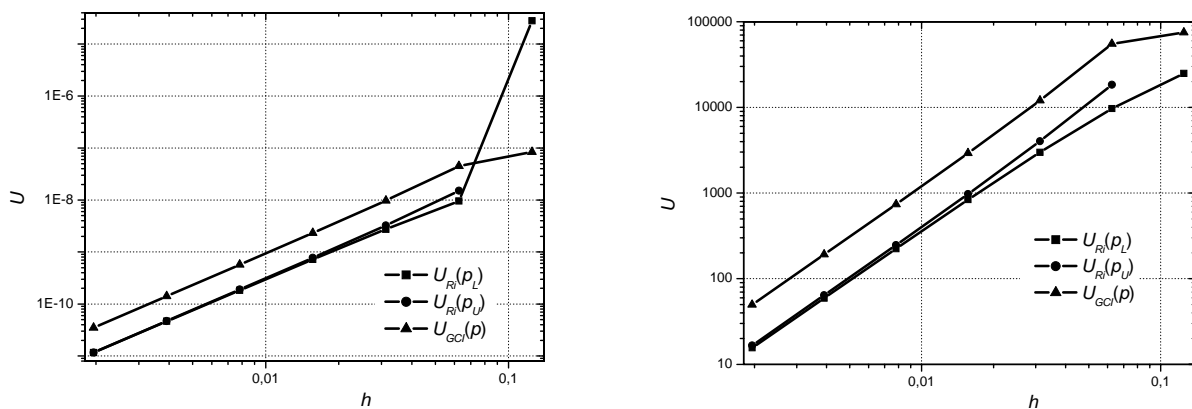
### 4.3. Resultados

As soluções numéricas foram obtidas para  $\beta = 16 \times 10^{-6} \text{ K}^{-1}$ ,  $E = 10^9 \text{ N/m}$ ,  $W = 1 \text{ m}$  e  $\mu = 0,32$ . As malhas utilizadas são uniformes com 2, 4, 8, ... até 512 elementos em cada direção. Para a malha mais refinada, o número de iterações para atingir o erro de máquina foi de  $\approx 6,4 \times 10^5$  que resultou no tempo de CPU de quase 18 horas. A Tab. 2 apresenta a solução numérica das duas variáveis de interesse, para a malha 512x512 elementos.

Tabela 2. Solução numérica com malha 512 x 512 do problema termoelástico.

Variável	Solução numérica ( $\phi$ ) e seu $U_{GCI}$
$u(3/4,3/4)$ [m]	$5,24863 \times 10^{-7} \pm 3,5 \times 10^{-11}$
$F_x$ [N]	$-404877 \pm 50$

Para as duas variáveis de interesse, a Fig. 3 apresenta a estimativa do erro de discretização ( $U$ ) baseada nas Eqs. (2), (7) e (8). A Fig. 4 apresenta as ordens assintótica ( $p_L$ ) e aparente ( $p_U$ ) do erro. Nota-se que  $p_U \rightarrow p_L$  monotonicamente à medida que  $h \rightarrow 0$ .



a) Deslocamento  $u(3/4,3/4)$ .

b) Força  $F_x$ .

Figura 3. Termoelasticidade: módulo da estimativa do erro ( $U$ ).

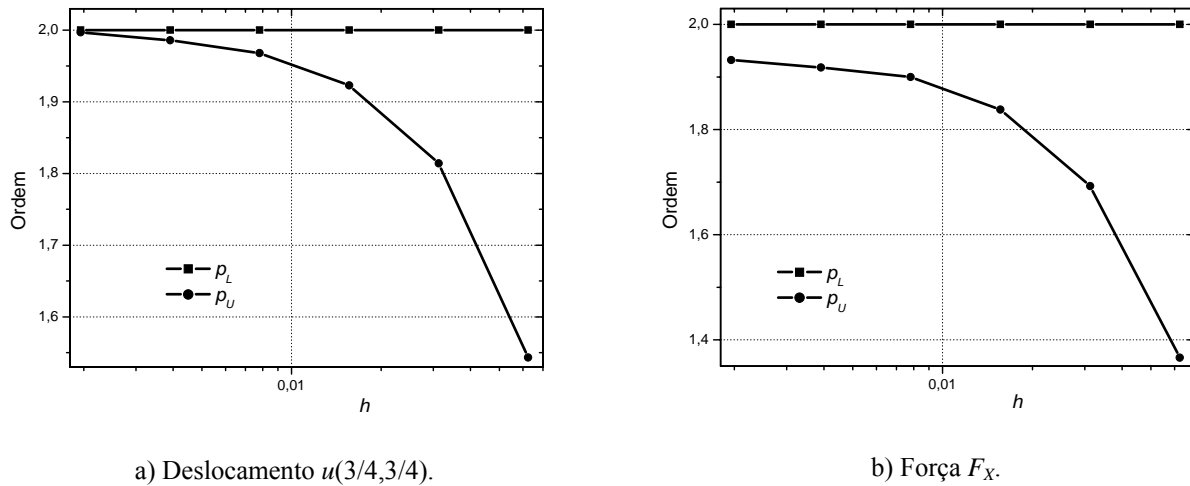


Figura 4. Termoelasticidade: ordens assintótica ( $p_L$ ) e aparente ( $p_U$ ) do erro.

## 5. Conclusão

Para o problema de condução de calor, cuja solução analítica é conhecida, verificou-se que:

- À medida que a malha é refinada, os valores das ordens efetiva ( $p_E$ ) e aparente ( $p_U$ ) tendem monotonicamente ao valor (2) teórico da ordem assintótica ( $p_L$ ) para as variáveis de interesse.
- A estimativa do erro ( $U_{GCI}$ ) é confiável em todos os pontos em que se comparou  $U$  com  $E$ , isto é, a solução analítica está contida no intervalo compreendido pela solução numérica ( $\phi \pm U_{GCI}$ ).

Para o problema termoelástico, cuja solução analítica não é conhecida, verificou-se que à medida que a malha é refinada, os valores de  $p_U$  tendem monotonicamente ao valor (2) teórico de  $p_L$  para as variáveis de interesse. Apresentou-se a solução numérica obtida com a malha 512x512 e a estimativa ( $U$ ) do seu erro de discretização.

## 6. Agradecimentos

O segundo autor é bolsista do CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico).

## 7. Referências

- Demirdzic, I. Martinovic, D., 1993, "Finite volume method for thermo-elasto-plastic stress analysis", Computer Methods in Applied Mechanics and Engineering, v. 109, p. 331-349.
- De Vahl Davis, G., 1983, "Natural convection of air in a square cavity: a bench mark numerical solution", International Journal for Numerical Methods in Fluids, v. 3, p. 249-264.
- Kreyszig, E., 1999, "Advanced Engineering Mathematics", 8<sup>th</sup> ed., New York: Wiley.
- Marchi, C. H., Silva, A. F. C., 2002, "Unidimensional Numerical Solution Error Estimation for Convergent Apparent Order", Numerical Heat Transfer, v. 42, p. 167-188.
- Marchi, C. H., 2001, "Verificação de Soluções Numéricas Unidimensionais em Dinâmica dos Fluidos", Tese Doutorado em Engenharia Mecânica, Universidade Federal de Santa Catarina, Florianópolis.
- Roache, P. J., 1994, "Perspective: a method for uniform reporting of grid refinement studies", ASME Journal of Fluids Engineering, v. 116, p. 405-413.
- Roache, P. J., 1998, "Verification and validation in computational science and engineering", Hermosa.
- Tannehill, J. C., Anderson, D. A., Pletcher, R. H., 1997, "Computational fluid mechanics and heat transfer", Taylor & Francis.
- Timoshenko, S. P., Goodier, J. N., 1970, "Theory of elasticity", Japan: McGraw-Hill.

# NUMERICAL SOLUTION OF THERMOELASTIC PROBLEM WITH DISCRETIZATION ERROR ESTIMATION

## **Orestes Hacke**

Universidade do Contestado – UnC  
Mafra, SC, Brazil  
oresteshacke@brturbo.com

## **Carlos Henrique Marchi**

Department of Mechanical Engineering  
Federal University of Paraná  
Curitiba, PR, Brazil  
marchi@demec.ufpr.br

**Abstract.** The main objective of this work is verify the error of numerical solution of a thermoelastic problem in two-dimensions. It is used the finite difference method with numerical approaches of second order. The true error ( $E$ ) of discretization and its estimative ( $U$ ) are evaluated. It is used the Richardson and GCI estimators. It were used uniform grids with 2 up to 1024 elements in each direction. It was verified that the GCI estimator supplies reliable uncertainty. The value of the effective and apparent orders approaches the asymptotic order monotonically as  $h \rightarrow 0$ .

Keywords: finite difference, GCI, Richardson extrapolation, heat conduction, thermoelasticity.

# MATHEMATICAL MODELING AND NUMERICAL SIMULATION OF A THREE-DIMENSIONAL FLOW OVER COMPLEX GEOMETRIES USING THE IMMERSED BOUNDARY METHOD

## João Marcelo Vedovoto

Faculty of Mechanical Engineering - Federal University of Uberlândia  
João Naves de Ávila Av. 2121, Uberlândia, MG, Brazil  
[jmvedovoto@mecanica.ufu.br](mailto:jmvedovoto@mecanica.ufu.br)

## Rubens Campregher

Department of Mechanical Engineering - Dalhousie University  
A1B 3X5 - Halifax, NS - Canada  
[rubenscamp@dal.ca](mailto:rubenscamp@dal.ca)

## Aristeu da Silveira Neto

Faculty of Mechanical Engineering - Federal University of Uberlândia  
João Naves de Ávila Av. 2121, Uberlândia, MG, Brazil  
[aristeus@mecanica.ufu.br](mailto:aristeus@mecanica.ufu.br)

**Abstract.** *The mathematical modeling and numerical simulation of real problems in engineering has a fundamental importance. However, the search for accurate and viable solutions constitutes in a great challenge.*

*The goal of this paper is to present the Immersed Boundary methodology for modeling and simulating flows over complex three-dimensional geometries, and also characterize the generation of wingtip vortices. This methodology is being developed in the Laboratory of Heat and Mass Transfer and Fluid Dynamics (LTCM).*

*The Immersed Boundary method uses two independent domains in the solution of the flows over complex geometries: An Eulerian domain, which is discretized using Finite Volume Method over a non-uniform mesh to integrate the Navier-Stokes equations, and a second-order approximation for time and space derivatives. The Lagrangian domain is represented by a superficial unstructured mesh, composed by triangles.*

*The in-house parallel code runs on a Beowulf-class cluster, a viable and reliable alternative to solve problems that demand very large computational resources. Three-dimensional flow over airfoils NACA-0012 were simulated aiming to accurately study wingtip vortices. The results present a good agreement with literature, and it is possible to understand the generation and development of the vortices.*

**Keywords.** *Immersed Boundary Method, Virtual Physical Model, complex geometries, wingtip vortices.*

## 1. Introduction

When an aircraft wing generates lift, it also produces horizontal, tornado-like vortices that create a potential wake-vortex hazard problem for other aircraft trailing. The powerful, high-velocity airflows contained in the wake behind the generating aircraft are long-lived, invisible, and a serious threat to aircraft encountering the system, especially small aviation aircraft. Thus the understanding of such phenomenon is fundamental.

In present work the Immersed Boundary method is used to simulate the conditions of generation and decay of those wingtip vortices. An in-house parallel code running on a Beowulf-class cluster is used.

In the Immersed Boundary (IB) methods, the presence of a solid or a gaseous interface inside a flow can be simulated by adding a source term into the Navier-Stokes equations, acting as a fluid body force. The way that this force is evaluated differentiates the methodologies among them. Furthermore, an important characteristic presented by the Immersed Boundary methodologies is that the immersed obstacle can be represented by a Lagrangian mesh while the flow domain can be discretized by an Eulerian grid such as Cartesian or cylindrical. There are, also, an interpolation function that promotes the information transfer from one domain to another. This domain independence allows one to promote the immersed body a displacement and/or a deformation relative to the flow grid.

The development of the Immersed Boundary method was credited to Charles Peskin and his collaborators, aiming to simulate the blood flow through cardiac valves. Accordingly to Peskin's work (Peskin, 1977), the source of the additional force term was due to the elastic boundary deformation rate, in which their constitutive points were tied by elastic membranes

More recently, Lima e Silva (Lima e Silva *et al.*, 2003) proposed a model that evaluates the force field by the momentum equation based on a three points scheme, similar to Mohd-Yusof's (Mohd-Yusof's, 1997) work, but using more simplified interpolation schemes that requires less computing resources. Since the model employs momentum equation and model the no-slip condition on the geometry wall in an indirect manner, the model has been called Virtual

Physical Model (VPM) (Lima e Silva *et al*, 2003). This current work proposal is to present an extension to three-dimensional domains of the VPM (Campregher, 2005) and apply it to the flow around a NACA- 0012 airfoil.

## 2. Mathematical and numerical modeling

The Immersed Boundary method uses two distinct domains to evaluate a flow over a complex geometry. An Eulerian domain is used to describe the behavior of the mean flow and covers the entire flow domain. For its turn, the Lagrangian domain is used to represent the interface fluid/fluid or fluid/solid.

This is one of the great advantage attributed to Immersed Boundary methods since it is possible to simulate flow around complex geometries using a more simplified Eulerian formulation for the fluid and a Lagrangian more versatile and simple grid for the interface fluid/solid. The coupling between Eulerian and the Lagrangian domains is done by Virtual Physical Model (Lima e Silva *et al* 2003).

In this work, Cartesian meshes were used to discretize the flow domain, configuring a simple and easy implementation, and at low computational cost. The following describes both domains in more details.

### 2.1. The Eulerian Domain

The domain was discretized by Finite Volume method over a structured non-uniform mesh. The flow is considered incompressible and isothermal. The integral form of the Navier-Stokes for such assumptions becomes:

$$\frac{\partial}{\partial t} \int_{\Omega} \rho \phi d\Omega + \int_S \rho \phi \mathbf{v} \cdot \mathbf{n} dS = \int_S \Gamma \nabla \phi \cdot \mathbf{n} dS + \int_{\Omega} q_{\phi} d\Omega, \quad (1)$$

where  $\phi$  is a property being transported,  $q_{\phi}$  is the term of generation or destruction of  $\phi$ , and  $\Gamma^{\phi}$  is the diffusivity of  $\phi$ .

The time derivative was approximated by a second-order three-time level (Ferziger & Peric, 2002), and the spatial derivatives by the Central-Difference Scheme.

The pressure-velocity coupling was done by the SIMPLEC method (Van Doormal e Raithby, 1984), with no relaxation in the velocity components equation. A co-located arrangement of variables was employed, and the Rhie-Chow (Rhie-Chow, 1983) interpolation method was used to avoid numerical oscillation due to pressure checkerboard fields.

The linear system originated from the velocity components discretization was solved by the SOR method. The SIP algorithm was used to solve the linear system generated by the discretization of the pressure correction equation.

The time and space integration of equation (1) over an elementary volume, shown in figure (1), after some mathematical arrangements leads to the following equation :

$$\begin{aligned} & \left( \frac{3\phi_p^n - 4\phi_p^{n-1} + \phi_p^{n-2}}{2\Delta t} \right) \Delta x \Delta y \Delta z + (\rho_e u_e \phi_e - \rho_w u_w \phi_w)^n \Delta y \Delta z + (\rho_n u_n \phi_n - \rho_s u_s \phi_s)^n \Delta x \Delta z + \\ & (\rho_t u_t \phi_t - \rho_b u_b \phi_b)^n \Delta x \Delta y = \left[ \left( \Gamma^{\phi} \frac{\partial \phi}{\partial x} \right)_e - \left( \Gamma^{\phi} \frac{\partial \phi}{\partial x} \right)_w \right]^n \Delta y \Delta z + \\ & \left[ \left( \Gamma^{\phi} \frac{\partial \phi}{\partial y} \right)_n - \left( \Gamma^{\phi} \frac{\partial \phi}{\partial y} \right)_s \right]^n \Delta x \Delta z + \left[ \left( \Gamma^{\phi} \frac{\partial \phi}{\partial z} \right)_t - \left( \Gamma^{\phi} \frac{\partial \phi}{\partial z} \right)_b \right]^n \Delta x \Delta y + q_{\phi} \Delta x \Delta y \Delta z \end{aligned} \quad (2)$$

The first term of the left-hand side of the equation (2) represents the discretization of the transient term by the *three-time level scheme* (Muzafferija and Peric, 1997). This scheme is a second order accurate in time.



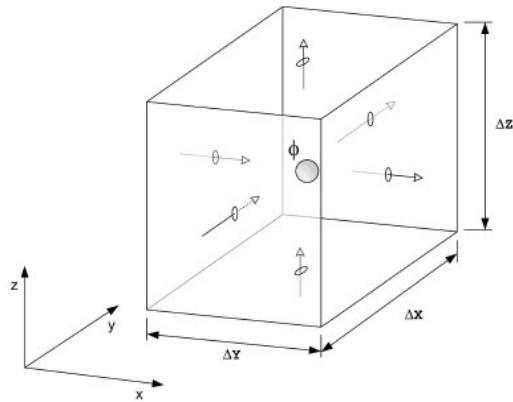


Figure 1: Elementary control volume, with variable  $\phi$  placed on centroid.

## 2.2. The Lagrangian domain

The Lagrangian approach for analyzing the movement of a particle constitutes of placing a system of coordinates at the particle and follow it individually. In other words, the system of coordinates moves through the flow following the particle. Thus, at each time step the particle keeps its own system of coordinates relatively to a global system of coordinates.

In the Virtual Physical model the geometry to be simulated is characterized by a Lagrangian set of points (see. Fig. (2)). This methodology permits to take advantage of the Lagrangian approximations like the ability to simulate moving bodies by just applying translation operations to the set of points.

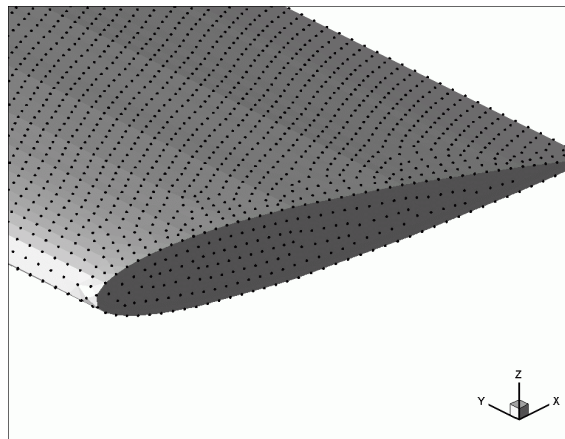


Figure 2: The surface of an airfoil NACA-0012 characterized by Lagrangian points.

The main characteristic of Immersed Boundary method is to simulate the presence of a fluid/solid or fluid/fluid interface inside a flow by adding a source term of force  $\vec{f}$  to the Navier-Stokes equations. In Fig. (3) an arbitrary Lagrangian point  $k$  is shown with coordinates  $\vec{x}_k$ , as well as an elementary volume of fluid with coordinates  $\vec{x}$ . The evaluation of  $\vec{f}$  differentiates the IB methods among them.

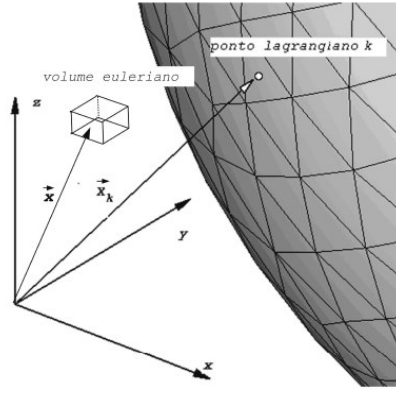


Figure 3: Schematic drawing of an arbitrary point  $k$  over a surface, placed on  $\vec{x}_k$ , and a element of fluid positioned in  $\vec{x}$

In the Virtual Physical Model, the Lagrangian force is obtained from a balance of momentum over a particle  $k$ , placed at  $\vec{x}_k$ . This particle also has properties pressure  $p_k$  and velocity  $\vec{V}_k$ . Thus, the force can be evaluated as:

$$\vec{F}_k = \frac{\partial(\rho \vec{V}_k)}{\partial t} + \vec{\nabla} \cdot (\rho \vec{V}_k \vec{V}_k) - \mu \nabla^2 \vec{V}_k + \vec{\nabla} p_k \quad (3)$$

The Eq. (3) can be interpreted as the necessary force so that a particle of fluid immediately adjacent to the Lagrangian point  $k$  reaches the velocity of this point, imposing a non-slip condition between the fluid and the immersed body.

Each term of the Eq. (3) has a particular meaning. The first term (the transient term) is responsible by the acceleration force ( $\vec{F}_{acc}$ ). The other terms, of spatial derivatives, are known as the advective term, the diffusive term and the pressure gradient term, respectively. These terms are responsible for inertial forces ( $\vec{F}_{inert}$ ), viscous forces ( $\vec{F}_{visc}$ ), and pressure forces ( $\vec{F}_{press}$ ). More details about this model and about each term evaluation can be found in Lima e Silva et al (Lima e Silva *et al*, 2003), and Campregher (2005).

The properties of the flow in the Eulerian mesh have to be interpolated to the Lagrangian mesh to calculate the Lagrangian forces. Once evaluated, the Lagrangian forces must be transferred back to the Eulerian domain. The connection between Lagrangian and Eulerian domains is promoted by the force distribution procedure.

### 2.2.1. The Virtual Physical Model

The discretization of Eq (3) is done by constructing a three-dimensional reference axis, with origin placed at the point  $k$ , as can be seen in the Fig (4). A Lagrangian polynomial is then used to obtain the space derivatives along each coordinate direction. Let  $m$  be a number of points employed to construct a polynomial interpolation of order  $m-1$ . Thus, the value of a property  $\phi$  along  $i$  direction, at any point  $p$ , is given by:

$$\phi_i(p) = \sum_m \psi_m(p) \phi_m, \quad (4)$$

where,

$$\psi_m(p) = \prod_{n, n \neq m} \left[ \frac{x_i(p) - x_i(n)}{x_i(p) - x_i(n)} \right]. \quad (5)$$

Substituting the  $m$  points, according to the stencil on Fig (4), the  $\phi$  property value along the  $x$  axis (where  $k, k_1$  and  $k_2$  points lay) can be obtained as:

$$\phi_p = \left[ \frac{(x_p - x_{k1})(x_p - x_{k2})}{(x_k - x_{k1})(x_k - x_{k2})} \right] \phi_k + \left[ \frac{(x_p - x_k)(x_p - x_{k2})}{(x_{k1} - x_k)(x_{k1} - x_{k2})} \right] \phi_{k1} + \left[ \frac{(x_p - x_k)(x_p - x_{k1})}{(x_{k2} - x_k)(x_{k2} - x_{k1})} \right] \phi_{k2}. \quad (6)$$

Deriving Eq (6) to  $x$  direction one has:

$$\frac{\partial \phi_p}{\partial x} = \left[ \frac{(x_p - x_{k1}) + (x_p - x_{k2})}{(x_k - x_{k1})(x_k - x_{k2})} \right] \phi_k + \left[ \frac{(x_p - x_k) + (x_p - x_{k2})}{(x_{k1} - x_k)(x_{k1} - x_{k2})} \right] \phi_{k1} + \left[ \frac{(x_p - x_k) + (x_p - x_{k1})}{(x_{k2} - x_k)(x_{k2} - x_{k1})} \right] \phi_{k2}, \quad (7)$$

and the second derivative results:

$$\frac{\partial^2 \phi_p}{\partial x^2} = \left[ \frac{2\phi_k}{(x_k - x_{k1})(x_k - x_{k2})} \right] + \left[ \frac{2\phi_{k1}}{(x_{k1} - x_k)(x_{k1} - x_{k2})} \right] + \left[ \frac{(x_p - x_k) + (x_p - x_{k1})}{(x_{k2} - x_k)(x_{k2} - x_{k1})} \right]. \quad (8)$$

From the equations above, it is possible to obtain every spatial derivative needed in Eq (3), just substituting the point  $p$  and the aimed variable  $\phi$ .

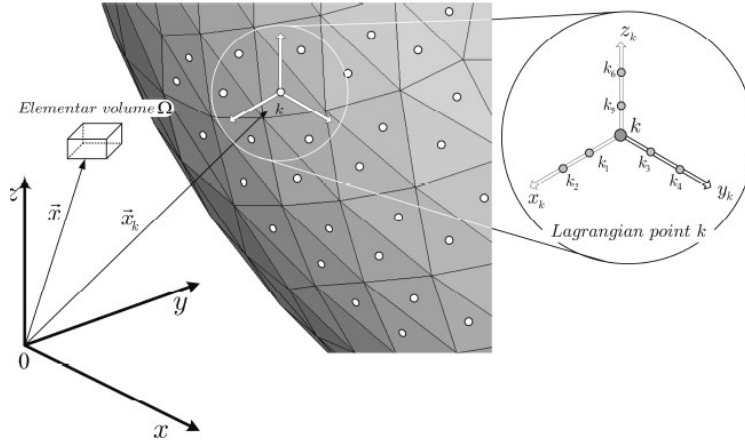


Figure 4: Position of the Lagrangian point  $\vec{x}_k$

A detailed view of a triangular element can be obtained in the Fig (5). The element sides are formed by line segments  $S_1, S_2,$  and  $S_3,$  between the vertex points  $P_1, P_2$  and  $P_3.$  Thus, one has  $S_1 = \overline{P_2 P_1}, S_2 = \overline{P_3 P_2}$  and  $S_3 = \overline{P_3 P_1}.$

The  $\Delta A_k$  is the triangular element surface area, which can be evaluated as:

$$\Delta A_k = \sqrt{S(S - S_1)(S - S_2)(S - S_3)} \quad (9)$$

where  $S = (1/2)(S_1 + S_2 + S_3).$  The  $\Delta S_k$  is the average length of the triangle sides. It worth noting that each of those geometric properties are associated to a Lagrangian point  $k.$

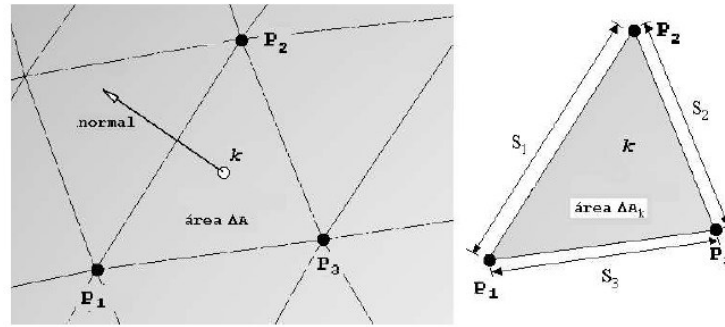


Figure 5: Detailed view of a triangular element

### 2.2.2. The distribution procedure

The Lagrangian force term  $\vec{F}$ , calculated at a Lagrangian point (denoted by  $\Omega_k$ ) is then distributed to Eulerian domain by means of the Dirac Delta Function. In a N-dimensional this function is defined as:

$$\vec{f}(\vec{x}) = \int_{R^n} \delta(\vec{x} - \vec{x}_k) \vec{F}(\vec{x}_k) d^n \vec{x}_k \quad (10)$$

Applying the Eq. (10) for a volume V of the Lagrangian domain,

$$\vec{f}(\vec{x}) = \int_{\Omega_n} \vec{F}(\vec{x}_k) \delta(\vec{x} - \vec{x}_k) d\vec{x}_k \quad (11)$$

The  $\delta$  function has the following property:

$$\int_{R^n} \delta(\vec{x} - \vec{x}_k) d\vec{x} = \begin{cases} 1 & \text{if } \vec{x}_k \in V \\ 0 & \text{if } \vec{x}_k \notin V \end{cases} \quad (12)$$

where  $V \in \Omega$ . This function acts as the core of a transformed integral (centered in  $\vec{x}_k$ ), which promotes the transposition between the Lagrangian and Eulerian domains (Griffith and Peskin, 2005).

In the Virtual Physical Model for three-dimensional domains, the Lagrangian force field ( $F_{i,k}$ ) is distributed over the Eulerian mesh using Eq (13).

$$f_i = \sum F_{i,k} D_i \Delta A_k \Delta S_k \quad (13)$$

The distribution function  $D_i$  is evaluated as:

$$D_i(x_k) = \prod_i \left\{ \frac{\varphi[(x_k - x_i)/\Delta x_i]}{\Delta x_i} \right\} \quad (14)$$

where the  $\varphi$  function is defined as:

$$\varphi(r) = \begin{cases} \tilde{\varphi}(r) & \text{if } \|r\| < 1 \\ \frac{1}{2} - \tilde{\varphi}(2-r) & \text{if } 1 < \|r\| < 2 \\ 0 & \text{if } \|r\| > 2 \end{cases} \quad (15)$$

$$\tilde{\varphi}(r) = \frac{3 - 2\|r\| + \sqrt{1 - 4\|r\| + 4\|r\|^2}}{8}. \quad (16)$$

The Distribution function is divided by a volume unit, that cancel out by multiplying for a characteristic area ( $\Delta A_k$ ) and for a characteristic length ( $\Delta S_k$ ). Thus, it remains the force density that is integrated over the volume  $\Omega$ .

The interface solid/fluid is managed by an indicator function  $I_i$ , built from:

$$\nabla^2 I_i = \nabla G_i, \quad (17)$$

where the  $G$  function is defined as:

$$G_i = \sum D_i \bar{n}_k \Delta A_k, \quad (18)$$

and the  $\bar{n}_k$  is the normal vector on the Lagrangian point  $k$ .

After the discretization of the Eq. (17), the algebraic equation system is evaluated by the MSI algorithm (Schneider and Zedan, 1981), a variation of the SIP procedure. By analyzing the Eq. (18), one can see that if the geometry is inserted into a non-uniform grid region, the interfacial region may become deformed, i.e., the geometry shell shape would be misrepresented.

Briefly describing, the force field evaluation procedure in the Virtual Physical Model can be stated as:

(1) With the flow field solved, the velocity components and the pressure are transferred, using the interpolation function given by Eq. (18), to the nearest Lagrangian points ( $k, k1...k6$ ) depicted in Fig. (4);

(2) Once having  $u_{i,k}$  and  $P_{i,k}$ , evaluates  $F_{i,k}$  by Eq. (3);

(3) Calculates the force field components, due to each Lagrangian point  $k$ , via Eq. (13);

(4) Advances in time;

(5) The force field are inserted into the source-term of Eq. (1);

(6) A new flow field is obtained and the procedure re-starts.

### 3. Parallel Programming Approach

One of the greatest problems involving complex physical problems simulations is the computational resources available. Thus, an in-house Beowulf cluster turns out to be very a viable solution due to its comparatively low cost when comparing to the supercomputers. Moreover, such clusters have good scalability, that is, it is possible to increase the cluster resource by adding more processors anytime. The cheaper assembly is ensured by employing on-the-shelf hardware and, mainly, due to the open source software, freely downloaded from internet.

Any group of ordinary machines connected by a local network may be enough to constitute a Beowulf cluster. This work ran on a homogeneous cluster of five machines with the following configurations: master node has an Intel® 865PERL mother-board, a Pentium4® 2.8GHz processor, with 1548 MB DDR RAM, 80 GB IDE hard drive, and a Radeon® 9200 128MB DDR AGP8x video card. The slave nodes have the same configuration, except by the video card, which is a Geforce® 64MB AGP4x. The postprocessing tasks are done by master node. All machines are connected by a Gigabit network via 3COM® 16 port switch.

The software uses for data interchange among processors the MPI (Message Passage Interface) library, more specifically a MPICH (MPI CHameleon). The MPI was chosen, mainly, due to its better performance on clusters of homogeneous machines. Furthermore, the MPI library has more than 120 functions permitting to write very efficient codes and, also, it is constantly updated by the MPI community.

The computational domain was split in 3 sub-domains with the Lagrangian mesh entirely in one of them, since it is not parallelized yet. Furthermore, as described above, it is imperative to use uniform spacing grids within and over the immersed geometry region. The other regions of the Eulerian domain was discretized in a non-uniform mesh to save computational resources. The way the problem was partitioned can be seen in fig. (6).

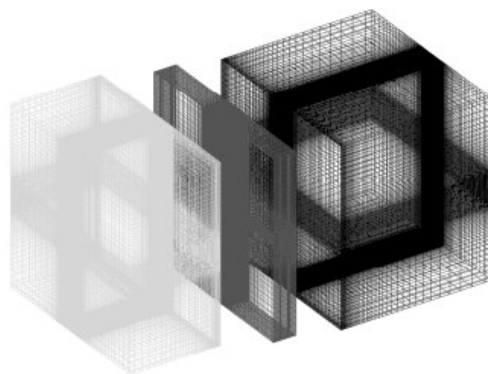


Figure 6: Eulerian sub-domains used in the simulations.

#### 4. Results and discussion

Craft (Craft et al, 2006) studied the formation and decay of wingtip vortices based on experimental results obtained by Chow (Chow et al, 1997). Those papers were used to qualitatively validate the present work.

The numerical Eulerian mesh used for every simulation is detailed in the Fig (7). The numerical domain has dimensions *length* ( $X$ ) = 0.88m, *width* ( $Y$ ) = 0.68 m and *height* ( $Z$ ) = 0.336 m. Such domain was discretized by, 148x268x76 grids in  $X$ ,  $Y$  and  $Z$  axis, respectively.

The NACA 0012 airfoil is represented by a triangular element mesh, as seen in Fig. (8), composed by 15446 nodes and 30888 elements. Again, the Immersed Boundary methodology requires only the discretization of the surface that represents the fluid/solid interface, i.e. the body shell. The airfoil is centered at  $(X_c, Y_c, Z_c) = (0.28, 0.34, 0.165)$  m, having a chord  $c=0.04m$  and a width of  $4c$ . For every simulation the attack angle  $\alpha = 10^\circ$  and the Reynolds number, based on chord, is  $Re = 10000$ .

The boundary condition for velocity components on the Eulerian domain side walls were set as free-slip conditions, the inflow at  $x=0$  m had a flat profile with values of  $u=U_I$  (inlet velocity),  $v=w=0m/s$ . The outflow was set of Neumann conditions.

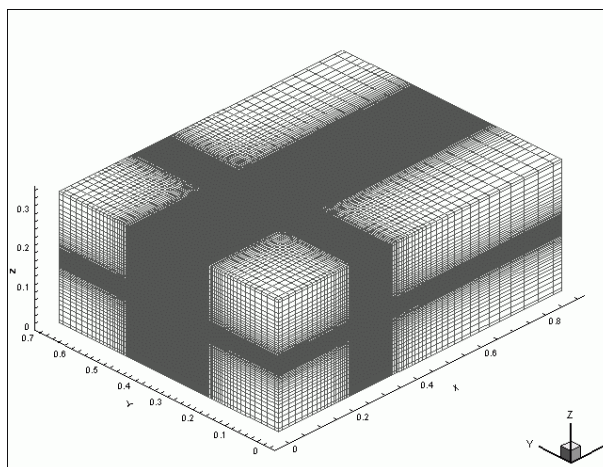


Figure 7: Eulerian and Lagrangian domain.

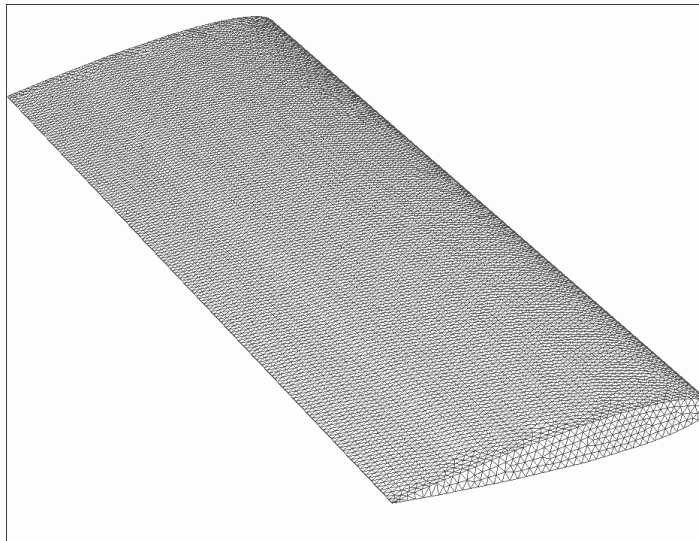
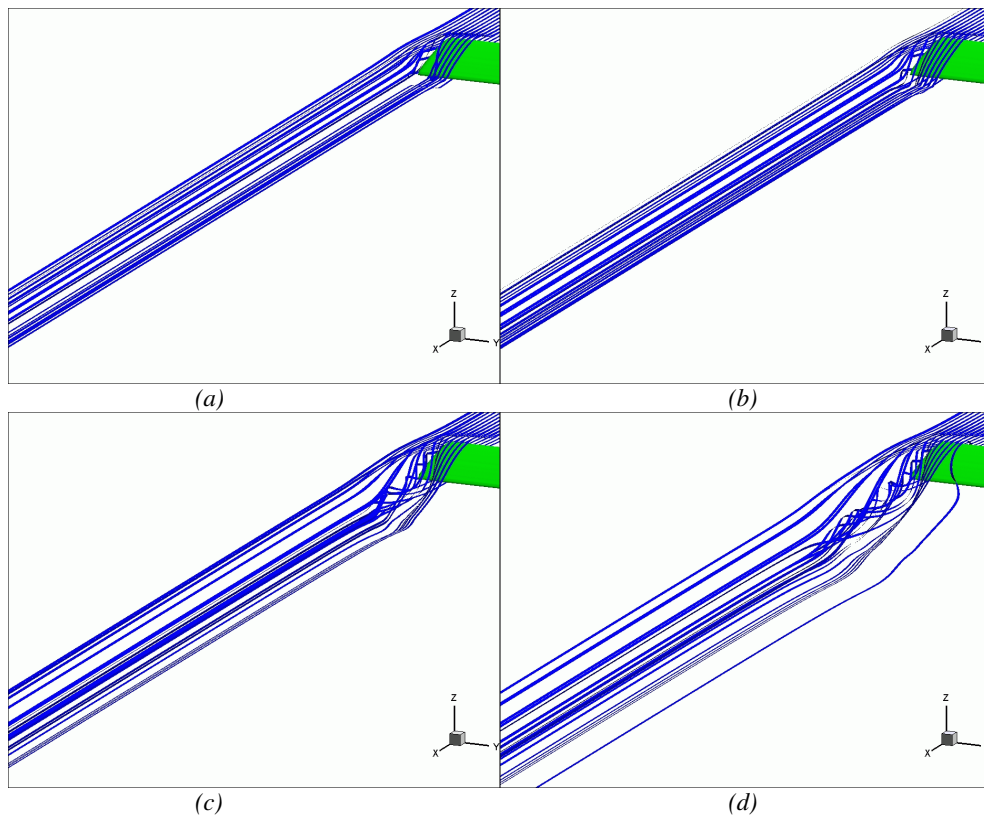


Figure 8: Airfoil discretized using a triangular elements mesh.

Computations are here reported as the flow over a NACA 0012 airfoil with a square wing tip. The generation and the evolution on time of the trailing vortices are well shown for a low Reynolds number ( $Re = 10000$ ).

In Figs (9a-h) it is possible to see the formation and evolution of the streamlines over the NACA 0012 airfoil of Fig (8). Since the overall flow pattern is symmetric at this flow regime, only one side of the airfoil is shown.



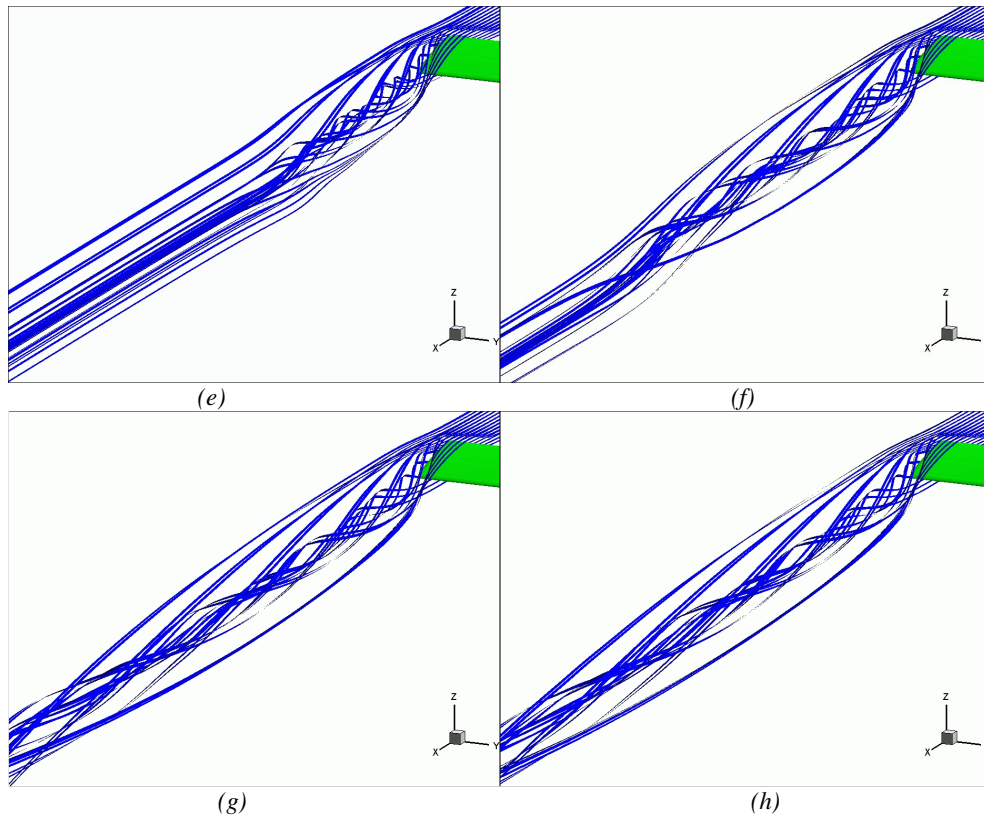


Figure 9: Evolution of streamlines over an NACA 0012 airfoil on times: (a)  $t=0.01s$ , (b)  $t=0.1s$  (c)  $t=0.5s$  (d)  $t=1.0s$  (e)  $t=1.5s$  (f)  $t=3.0s$  (g)  $t=5.0s$  (h)  $t=10.0s$ .

The value of the L2 norm was about  $10^{-3}$ , which is acceptable, once that the code has a second order in time-space accuracy.

Figure (10) shows three perpendicular planes to the airfoil. It's notable the influence of the tip of wing on the flow. We see that the flow is symmetric, and at the airfoil middle a Kelvin-Helmholtz like instability is formed and it is being transported. Figure (11) and Fig (12), show the top view and the back view of Fig (10). At Fig (12) the helicoidal wingtip vortices are clearly shown.

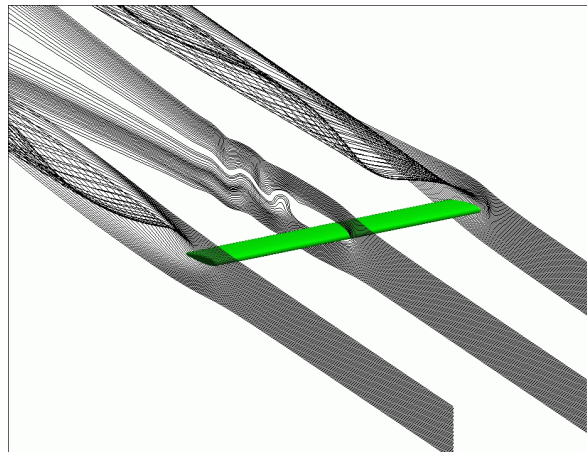


Figure 10: Perpendicular planes at  $t=10.0s$ .



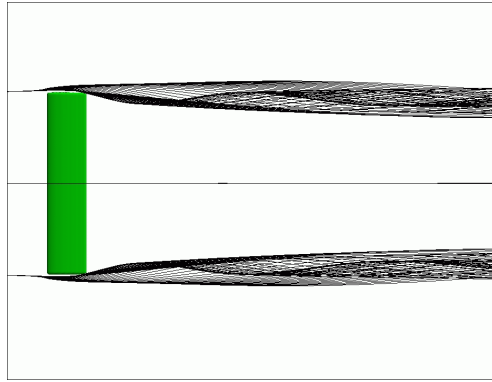


Figure 11: Top view of Fig. 10

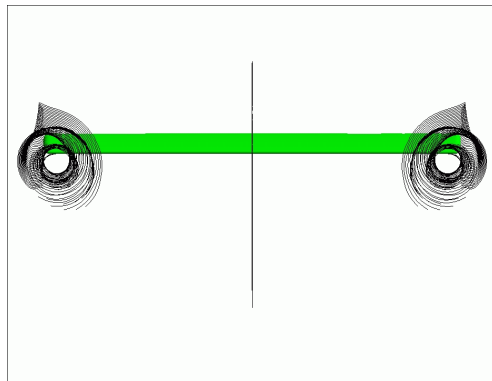
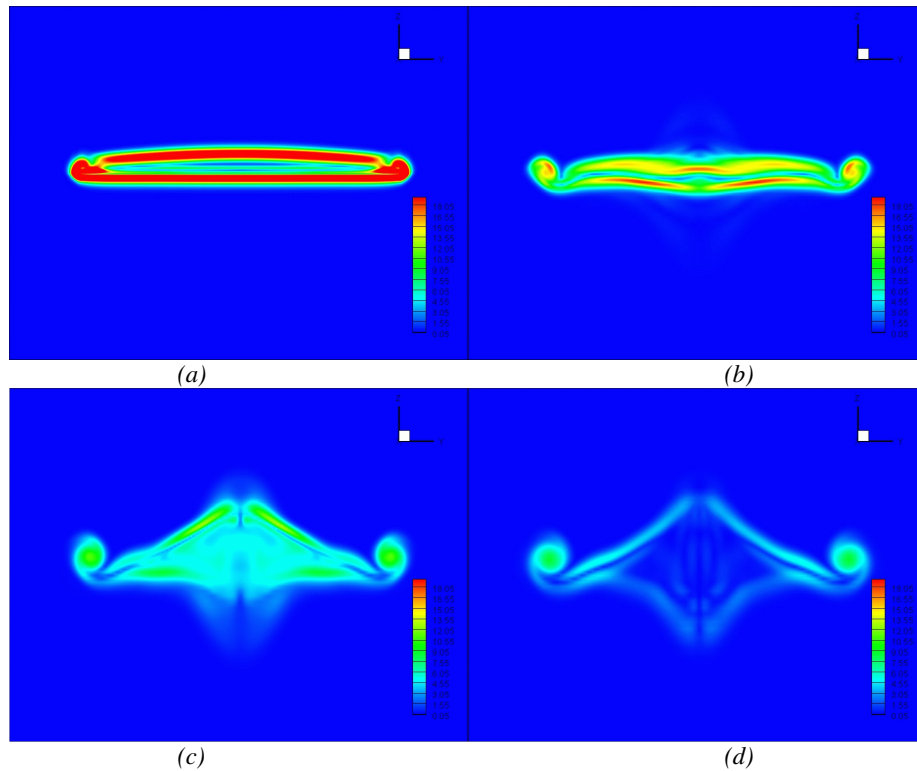


Figure 12: Back view of Fig 10.

In Figs (13a-f), the vorticity field is shown over six transversal cuts. It's possible to see the formation and evolution of the vertical structures over the NACA 0012 airfoil.



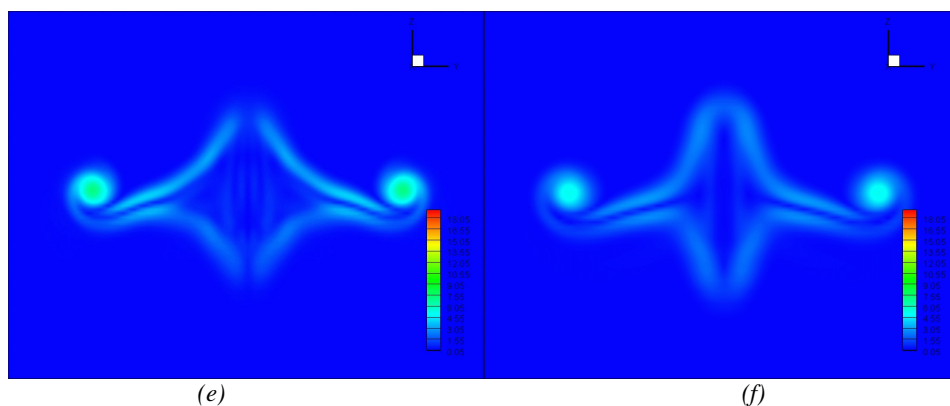


Figure 13: Evolution of the vorticity field behind the airfoil in  $t=10s$  at positions: (a)  $x=0.3m$ , (b)  $x=0.34m$  (c)  $x=0.4m$  (d)  $x=0.45m$  (e)  $x=0.5m$  (f)  $x=0.6m$

In Fig (13) it is possible to note that the main structure that remains in the flow are the vortices generated by the tip of the airfoil. From this result, it can be said that those structures are really matter of concern, when studying large aircrafts.

## 5. Conclusion

In this work, which represents an extension of Campregher's work (Campregher, 2005), the authors intended to present more applications to Virtual Physical Method. The flow around a single airfoil NACA 0012 at  $Re = 10000$  was chosen as a test case to identify and characterize the generation and the decay of wingtip vortices. It was done successfully.

The Immersed Boundary Method have shown great capability in dealing with complex geometries and/or moving bodies, once the Eulerian mesh is a Cartesian mesh, very simple to be created, and there are no further difficulties in generating the Lagrangian mesh, which is a advantage if compared with other methodologies of studying Fluid-Structure Interaction.

The results present a good agreement with literature (Chow et al, 1994 and Craft et al, 2006), and also, it is possible to understand the generation and development of the vortices. Although, further investigation about the influence of the width of the airfoil in the flow, as well as development for high Reynolds number have to be done.

## 6. References

- Campregher, Rubens, Modelagem Matemática tridimensional para problemas de interação fluido-estrutura, doctoral thesis, Uberlândia 2005
- Chow, J.S., 1994. Turbulence measurements in the near-wake of a wingtip vortex. Thermosciences Division Rep. No. MD-69, Mechanical Engineering Department, Stanford University.
- Craft, T.J.; Gerasimov, A.V.; Launder, B.E.; Robinson C.M.E.; A computational study of the near-field generation and decay of wingtip vortices. International Journal of Heat and Fluid Flow, 2006
- Ferziger J and Peric M., Computational methods for fluid dynamics 3th ed., 2002, Springer-Verlag, New-York, USA.
- Griffith, B. e Peskin, C., 2005, "On the order of accuracy of the immersed boundary method: Higher order convergence rates for sufficiently smooth problems.", Journal of Computational Physics 208, 75-105.
- Lima e Silva A.L.F., Silveira-Neto, A., and Damasceno, J.J.R; Numerical simulation of two dimensional flows over a circular cylinder using the immersed boundary method, J. Comp. Phys. 189, 351 (2003)
- Mohd-Yusof, J. Combined immersed boundaries/B-splines methods for simulations in complex geometries, CTR Annual Research Briefs, NASA Ames/Stanford University, 1997.
- Muzaferija, S. e Peric, M., 1997, "Computational of free-surface flows using the Finite-Volume method and moving grids", Numerical Heat Transfer, Part B 32, 369-384.
- Peskin C.S., Numerical analysis of the blood flow in the heart, J. Comp. Phys. 25, 220 (1977)
- Rhie C. and Chow W., Numerical study of turbulent flow past an airfoil with trailing edge separation, AIAA Journal 21, 1525 (1983)
- Schneider G.E. and Zedan M., A modified strongly implicit procedure for the numerical solution of field problems, Numer. Heat Transfer 4, 1 (1981)
- Van Doormal, J. e Raithby, G., "Enhancements of the simple method for predicting incompressible fluid flows", Numerical Heat Transfer 7, 147-163, 1984

## COMPARISON OF POISSON SOLVERS IN A TRANSIENT 2D FLUID FLOW PROBLEM WITH DIRICHLET BOUNDARY CONDITIONS

**Leandro F. Souza**

ICMC – USP, Av. Trabalhador São-carlense, 400 - Cx. Postal: 668 - CEP: 13560-970 - São Carlos - SP  
lefraso@icmc.usp.br

**Cassio M. Oishi**

ICMC – USP, Av. Trabalhador São-carlense, 400 - Cx. Postal: 668 - CEP: 13560-970 - São Carlos - SP  
oishi@icad.icmc.usp.br

**José A. Cuminato**

ICMC – USP, Av. Trabalhador São-carlense, 400 - Cx. Postal: 668 - CEP: 13560-970 - São Carlos - SP  
jacumina@icmc.usp.br

**Aristeu da Silveira Neto**

UFU, Av. Engenheiro Diniz, 1178 - Cx. Postal: 593 - CEP: 38.400-902 - Uberlândia - MG  
aristeus@mecanica.ufu.br

**Abstract.** *The numerical simulation of incompressible fluid flow often requires a Poisson solver subroutine. For the majority of these Computational Fluid Dynamics codes, this subroutine is one of the greatest CPU time consumer. In the present paper 6 different Poisson solver subroutines are presented and compared, namely: point SOR (Successive Over-Relaxation), line SOR, conjugate gradient method, Full Approximation Storage (FAS) multigrid solver with: line SOR, conjugate gradient method and a mixed (line SOR – conjugate gradient). All of the above methods use second order finite difference discretization in space. A Cartesian grid is used for each test case. The distance between consecutive points is constant in both directions. A performance test of the Poisson subroutines is carried out using a problem with analytical solution. An evaluation of the Poisson subroutines in a 2D incompressible flow simulation is done using a driven cavity flow as benchmark. In these cases, the equations are written in vorticity-stream function formulation. A fourth order Runge-Kutta scheme is used as time integrator. Different CFL numbers are used with each Poisson solver. The results show that the multigrid solvers are the most efficient for the tests performed here.*

**keywords:** *Poisson equation, Vorticity-Stream function formulation, Multigrid methods, incompressible flow.*

### 1. Introduction

Incompressible flow phenomena arise in a number of applications in science and engineering. Almost all numerical solutions of incompressible fluid flows requires a Poisson solver subroutine. Therefore, the study of efficient iterative methods and multigrid schemes for solving the Poisson equation are very important in Computational Fluid Dynamic (CFD).

The numerical solution of the Poisson equation can be obtained by *iterative methods* or *direct methods*. In the present study only *iterative methods* are evaluated. The term *iterative method* refers to a wide range of techniques that use successive approximations to obtain more accurate solutions of linear systems at each time step. In this work we will study two classes of iterative methods: stationary and non-stationary methods.

Stationary methods, known as Basic iterative methods, that are older but simpler to understand and implement, and usually not very efficient. In this paper, the Basic iterative method considered is the Successive Over-Relaxation. The Successive Over-Relaxation can be derived from the Gauss-Seidel method by introducing an relaxation parameter  $\omega$ . For the optimal choice of  $\omega$ , Successive Over-Relaxation may converge faster than Gauss-Seidel by an order of magnitude. An elementary introduction is given in Wesseling, 2001; Saad, 2003.

Non-stationary methods, known as Krylov subspace methods are relatively recent; their analysis are usually harder to understand, but they can be highly effective. The Conjugate Gradient method was adopted here as a type of Krylov subspace method. The conjugate gradient method derives its name from the fact that it generates a sequence of conjugate (or orthogonal) vectors. These vectors are the residuals of the iterations. They are also the gradients of a quadratic functional, the minimization of which is equivalent for solving the

linear system. The conjugate gradient converges to the correct solution when the matrix is symmetric positive definite, and is very efficient, since storage for only a limited number of vectors is required. Various authors have presented the theory and implementations of conjugate gradient (Golub and Van Loan, 1989; Baker, 1991; Wesseling, 2001; Saad, 2003).

Convergence acceleration techniques linked with iterative methods have recently been developed and have led to a dramatic improvement in convergence rates Hirsch, 1988. The multigrid method works as an acceleration technique and is the most effective and general iterative technique known today. This method has its origin in the properties of conventional iterative techniques, in particular, their asymptotic slow convergence because of the poor damping of high frequency errors (short wavelengths). The basic ideas of a multigrid method follows the lines:

- apply one or more sweeps of an iterative method with good smoothing properties of the higher-frequency components;
- transfer the problem to a coarse grid;
- transfer the corrections obtained to the fine grid in order to generate a new approximation of the solution.

In the current work an analysis of Poisson solvers subroutines was done. In the first test case the subroutines were evaluated in a simple test where the analytical solution is known. The performance of the Poisson solvers subroutines in a numerical simulation of a 2D incompressible flow was also analyzed. In these cases the Navier-Stokes equation were written in vorticity-stream function formulation. The benchmark test adopted was a driven cavity flow. Three different mesh sizes were tested, and for each mesh many CFL numbers were imposed. These tests enable a evaluation of the performance of the Poisson solver subroutines in a 2D flow simulation.

## 2. Formulation

For the numerical solution, the Navier-Stokes equations were written in vorticity-stream function formulation. The vorticity in the spanwise direction, denoted by  $\omega$ , is:

$$\omega = \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}. \quad (1)$$

and the stream function  $\psi$  is defined such that:

$$u = \frac{\partial \psi}{\partial y}, \quad (2)$$

and

$$v = \frac{\partial \psi}{\partial x} \quad (3)$$

The vorticity transport equation can be obtained by applying a rotational of the momentum equation is:

$$\frac{\partial \omega}{\partial t} = -u \frac{\partial \omega}{\partial x} - v \frac{\partial \omega}{\partial y} + \frac{1}{Re} \left( \frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} \right). \quad (4)$$

The continuity equation is:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \quad (5)$$

Using the definitions of stream function Eqs. (2) and (3) and applying them in the vorticity definition Eq. (1) a Poisson-type equation for the  $\psi$  can be derived:

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = \omega. \quad (6)$$

Equations (2), (3), (4) and (6) were solved numerically by the schemes described below. The solution was marched in time according to the following steps:

1. Impose initial conditions for  $u$ ,  $v$ ,  $\omega$  and  $\psi$  compatible with each other;
2. Calculate the vorticity from the vorticity transport equation (4), for time  $t + dt$ ;
3. Calculate  $\psi$  from the Poisson equation (6);

4. Calculate  $v$  velocity from Eq. (3);
5. Calculate  $u$  velocity from Eq. (2);
6. Calculate the vorticity generation at the wall for the new velocity distribution;
7. return to the second step until the desired integration time is reached.

### 3. Numerical Method

The time derivatives in the vorticity transport equations were discretized with a classical 4<sup>th</sup> order Runge-Kutta integration scheme Ferziger and Peric, 1997. The spatial derivatives were calculated using a 2<sup>nd</sup> order finite difference schemes. The vorticity updating at the wall, for all boundaries but the upper boundary, was done using the formula:

$$\omega_{wall} = \frac{-85\psi_{wall} + 108\psi_{wall\pm1} - 27\psi_{wall\pm2} + 4\psi_{wall\pm3}}{18(\Delta)^2} \quad (7)$$

The vorticity at the upper boundary was calculated by:

$$\omega_{jmax} = \frac{11u_{jmax} - 18u_{jmax-1} + 9u_{jmax-2} - 2u_{jmax-3}}{6\Delta y} \quad (8)$$

In the present work, six different Poisson solver subroutines were evaluated. The details of each one are given below.

#### 3.1. Poisson Solvers

The Poisson solvers tested in the current work are classified as iterative methods. The algebraic systems that arise from the discretization are sparse and can be very large (depending on the number of points in both directions). Therefore iterative methods are more efficient and demand far less storage than direct methods. The methods implemented can be classified as:

##### 3.1.1. Basic iterative methods

These methods were very popular in the 60's. The first iterative methods for solving large linear system were based on relaxation of the coordinates. Beginning with a given approximate solution, they modify the components of the approximation, one or a few at a time and in a certain order, until convergence is reached. One of the main difficulties with these methods is finding an optimal relaxation factor for general matrices. However when combined with more efficient methods they can be quite successful. Moreover, there are some application areas where variations of these methods are still quite popular Wesseling, 2001.

##### 3.1.2. Krylov subspace methods

The system to be solved can be written as  $Ax = b$ , where  $A$  is a large and sparse matrix. These techniques are based on projection processes, both orthogonal and oblique, onto Krylov subspace, which are subspaces spanned by vectors of the form  $p(A)v$ , where  $p$  is a polynomial. In short, these techniques approximate  $A^{-1}b$  by  $p(A)v$ , where  $p$  is a "good" polynomial Saad, 2003. These methods are widely used because their rate of convergence can be estimated. The attractiveness of those methods, for large sparse systems, is that they reference  $A$  only through the multiplication by a vector. The gradient (or steepest descent) and the conjugate gradient methods can be applied to the solution of  $Ax = b$  only when  $A$  is symmetric and positive definite.

##### 3.1.3. Multigrid methods

The convergence of Krylov's subspace methods for solving systems arising from discretized partial differential equations tends to slow down considerably as these systems become larger. In contrast the multigrid methods are, in theory, independent of the mesh size. One significant difference with the Krylov subspace approach is that multigrid methods were initially designed specifically for the solution of the discretized elliptic Partial Differential Equations. The multigrid methods were later extended to nonlinear systems, and other type of equations. The discovery of multigrid methods Brandt, 1977 is the most significant development in numerical analysis in the last 30 years, and is having a strong impact on Computational Fluid Dynamics (CFD) Wesseling, 2001. A review of the application of multigrid in CFD is showed by Wesseling, 2001. The main advantage of these methods is their very fast convergence rate, however they may require implementations that are specific to the physical problem at hand. Three multigrid methods were tested and their main differences are the solver

used in each grid. A Full Approximation Scheme (FAS) was adopted, and a  $V$  cycle with 4 meshes were used in all the methods described below.

There are many multigrid methods that solve elliptic problems (Briggs, 1987; Gupta *et al.*, 1997; Spitaleri, 2000; Zhang, 1996; Zhang, 1997). However, the multigrid idea can also be applied to nonlinear problems. The adopted algorithm in this work is the FAS applied in a  $V$  cycle with 4 meshes (Stüben and Trottenberg, 1981). The Poisson equation, Eq. (6) can be written in the form:

$$\nabla^2 f = st, \quad (9)$$

where  $\nabla^2$  is the Laplacian operator,  $f$  is the unknown solution, and  $st$  is the source term. This equation is solved in each grid, using an iterative procedure.

The multigrid scheme works in the following way, taking  $h$  as the distance between two consecutive points in the finest mesh, 2 iterations is performed in this mesh ( $h$ ):

$$\nabla^2 f_h = st_h. \quad (10)$$

The residual is calculated next from ( $res_h$ ):

$$res_h = st_h - \nabla^2 f_h. \quad (11)$$

To define the variables in the next grid (coarser), a operation called restriction is used:

$$f_h^i \Rightarrow f_{2h}^0 \quad (SI), \quad (12)$$

$$res_h^i \Rightarrow res_{2h}^0 \quad (FW), \quad (13)$$

where  $SI$  stands for Straight Injection and  $FW$  stands for Full Weight. More details of the restriction operation can be found in Press *et al.*, 1997.

Then, the source term for the second mesh is calculated:

$$st_{2h} = res_{2h}^0 + \nabla^2 f_{2h}^0. \quad (14)$$

Here again 2 iterations are done in this mesh ( $2h$ ):

$$\nabla^2 f_{2h} = st_{2h}. \quad (15)$$

After that, the residual is calculated from( $res_{2h}$ ):

$$res_{2h} = st_{2h} - \nabla^2 f_{2h}, \quad (16)$$

and this procedure is repeated until the coarsest mesh ( $8h$ ) is reached, where 40 iterations are performed:

$$\nabla^2 f_{8h} = st_{8h}. \quad (17)$$

The way back to the finest mesh starts with the calculation of the correction on the coarsest mesh:

$$corr_{8h} = f_{8h}^i - f_{8h}^0. \quad (18)$$

This correction is interpolated, using a bi-linear interpolation to the grid  $8h$  to the next grid  $4h$ :

$$corr_{8h} \Rightarrow corr_{4h}. \quad (19)$$

After that, the function is corrected, using new values:

$$f_{4h} = f_{4h}^i + corr_{4h}. \quad (20)$$

One single iteration is performed in this mesh ( $4h$ ):

$$\nabla^2 f_{4h} = st_{4h}, \quad (21)$$

Then the correction in this mesh is also calculated, and this procedure is repeated until the finest mesh is reached:

$$\nabla^2 f_h = st_h. \quad (22)$$

The number of  $V$  cycles used to obtain the solution of the Poisson equation depends on the number of points in the mesh and on the minimum residual allowed on it. Figure 1 illustrates the above multigrid procedure.

Three different iterative schemes were used with the multigrid method and are described below.

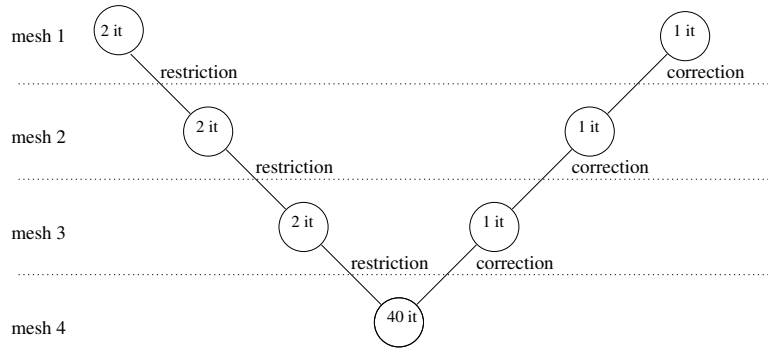


Figure 1: V Cycle.

- Multigrid with LSOR (FAS-LSOR)

This method uses a Line Successive Over-Relaxation (LSOR) as the iterative procedure in each mesh. Special attention should be given to the relaxation factor used in this procedure, a value between 1 and 2 can be used when the method goes from the fine to the coarse grid but this factor must be 1 when the method goes from coarse to the fine grid, otherwise the smoothing of the high frequencies, main idea of this scheme, does not work.

- Multigrid with Conjugate Gradient (FAS-CG)

This method uses the Conjugate Gradient (CG) iterative procedure in each mesh. The main advantage is that no relaxation factor is needed in this scheme.

- Multigrid with LSOR and CG (FAS-mix)

Several tests were performed changing the iterative procedures in the multigrid scheme. The best CPU time results were obtained when Line Successive Over-Relaxation (LSOR) was adopted in the 3 first meshes and a Conjugate Gradient (CG) was used in the coarsest mesh.

#### 4. Results

This section is divided in two, the first one shown the comparison of the Poisson solvers for a simple test case; and the second part shows the results of a 2 dimensional incompressible flow simulation where the Poisson solvers were evaluated. The results were obtained using Atlon AMD 64 3200+ computers. The processing time obtained in each simulation is shown in seconds.

##### 4.1. Poisson equation case

Initially, the efficiency of each solver is tested for the Poisson equation (6) where the minimum and maximum values of  $x$  and  $y$  were 0 and 1, respectively. The boundary conditions and the source term for this test case are given by:

Left boundary	⇒	$\psi(0, y) = \sin(y)$
Right boundary	⇒	$\psi(1, y) = \cos(1)\sin(y)$
Lower boundary	⇒	$\psi(x, 0) = 0.0$
Upper boundary	⇒	$\psi(x, 1) = \cos(x)\sin(1)$
Source term	⇒	$\omega(x, y) = -2\cos(x)\sin(y)$

Observing the results in Tab. 1, and comparing the PSOR, LSOR and CG methods, one can note that the CG method is faster than the other PSOR and LSOR methods when we have a large number of points. Comparing the results obtained by the multigrid methods, the FAS-MIX method is the faster in almost all cases. The FAS-CG method is the slowest method, when comparing with the other multigrid methods, but is faster than the PSOR, LSOR and CG methods in the most tested cases. The result obtained with the multigrid method with CG solver demands a construction of a complex system in each mesh, and this causes an increase in processing time.

Table 1: CPU time (seconds) for test case.

Mesh	PSOR	LSOR	CG	FAS-LSOR	FAS-CG	FAS-MIX
65 × 65	0.02299	0.04999	0.04799	0.02599	0.03499	0.02599
65 × 129	0.14497	0.52791	0.17597	0.05598	0.16397	0.05598
65 × 257	1.06381	4.38333	0.85687	0.11700	0.89385	0.11596
65 × 513	9.60253	38.0681	4.19140	0.25	1.23681	0.24218
129 × 65	0.14597	0.10198	0.17497	0.05698	0.15797	0.05499
129 × 129	0.48791	1.03584	0.51791	0.12197	0.20797	0.12098
129 × 257	2.68560	8.86767	2.32775	0.25891	1.01672	0.25195
129 × 513	20.2465	75.6035	9.66210	0.54199	11.7773	0.52441
257 × 65	1.06584	0.20295	0.86486	0.24996	0.86788	0.24295
257 × 129	2.61560	2.08966	2.48657	0.28198	1.25488	0.27795
257 × 257	8.88867	18.0651	6.02392	0.58789	1.24169	0.57812
257 × 513	45.0468	146.985	21.5058	1.22851	18.1875	1.18359
513 × 65	8.41259	0.40405	4.58837	1.97070	1.32690	1.91577
513 × 129	18.3930	4.27832	8.84277	1.14892	7.58691	1.14794
513 × 257	43.5468	35.9785	19.8515	1.99609	8.56445	1.22656
513 × 513	139.898	291.054	43.7070	4.12500	5.38281	2.49609

#### 4.2. Navier-Stokes equation cases

In order to verify the efficiency of the Poisson subroutine solvers in a transient simulation, many CFL numbers were imposed. The simulations were stopped when:

$$\omega_{i,j}^{n+1} - \omega_{i,j}^n \leq 1 \times 10^{-5}. \tag{23}$$

This criteria was imposed in all point  $(i, j)$  of the domain. The CPU times to obtain this criteria are shown bellow, for each Poisson subroutine solvers. The graphics are plotted in log x log scales, with CFL number in the x direction and CPU time in the y direction. The CFL numbers are plotted in a reverse axis direction.

Figure 2 shows the results obtained for a mesh of 65 x 65 points. These results using FAS-MIX and FAS-LSOR subroutines showed the lower CPU times for any CFL number. The FAS-MIX results were slightly better than the FAS-LSOR results in all tested cases. The use of FAS-CG, PSOR and LSOR subroutines in the numerical code showed similar results. The CPU times obtained with PSOR subroutine were better than the FAS-CG and LSOR. The CG subroutine results required the larger CPU times, if compared to the others. These could be explained by the fact that the solver was not preconditioned. It was verified that the CG subroutine demands more CPU time to obtain a solution, if compared to the other methods, when this solution is close to the initial condition.

The CPU time results obtained with a mesh of 129 x 129 points were very similar to the results with a mesh of 65 x 65 points as can be seen in Fig. 3. The only difference is that, with CFL number lower than  $5 \times 10^{-2}$ , the use of FAS-CG subroutine consumed more CPU time than LSOR and PSOR methods.

Table 2 shows the CPU time results for a mesh with 257 x 257 points. It can be seen that the results with FAS-MIX subroutine were the fastest than the other tested methods. The simulations with CFL = 0.01, using PSOR, LSOR and CG subroutines, were not performed because of the long time required. In these cases, the use of FAS-CG method showed best results if compared to PSOR, LSOR and CG results. The CPU time results using multigrid methods were lower than the other iterative methods. As the number of points increases this becomes more visible, as expected.

Table 2: CPU time results (seconds) versus CFL number – 257 x 257 points.

CFL Number	FAS-LSOR	FAS-MIX	FAS-CG	PSOR	LSOR	CG
0.1	4244.456	3824.622	22057.372	83874.687	90526.093	116068.031
0.05	6211.566	5964.343	34671.631	128348.285	137669.625	195534.25
0.01	16248.661	16076.160	87336.236	-	-	-



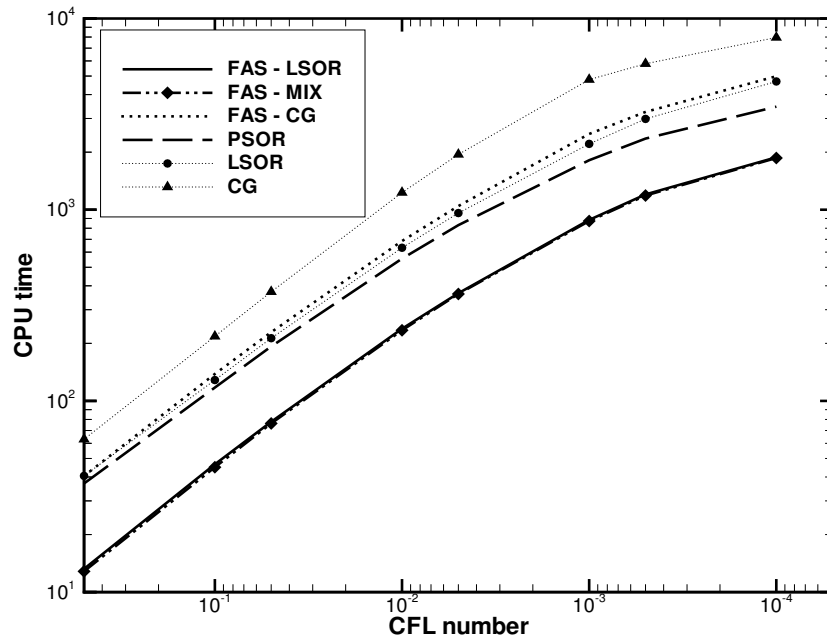


Figure 2: CPU time results (seconds) versus CFL number – 65 x 65 points.

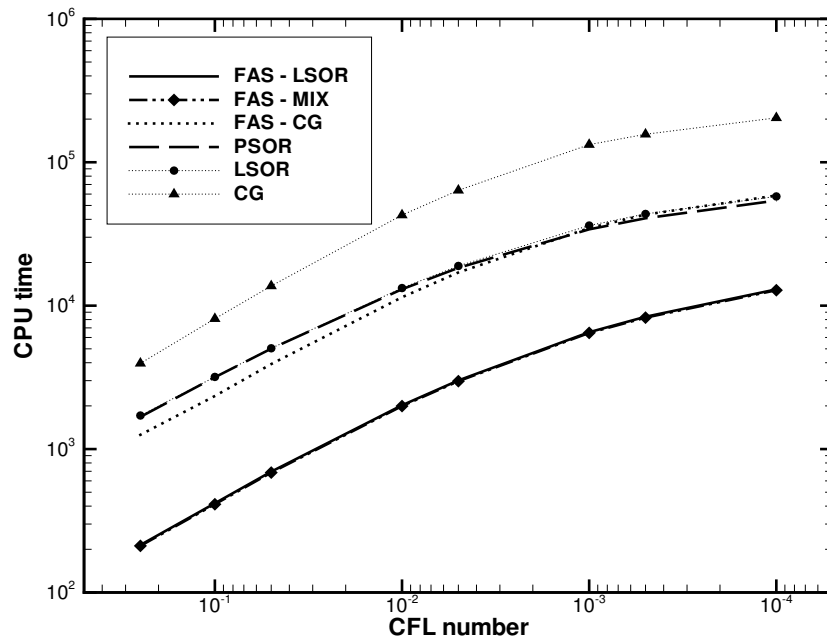


Figure 3: CPU time results (seconds) versus CFL number – 129 x 129 points.

## 5. Conclusions

In the present work, a study of the Poisson solver subroutine was carried out. Six different Poisson solver subroutines were studied and compared, namely: point Successive Over-Relaxation – PSOR, line Successive Over-Relaxation – LSOR, Conjugate Gradient method – CG, Full Approximation Storage (FAS) multigrid solver with: line SOR, conjugate gradient method and a mixed (line SOR – conjugate gradient). These solvers were tested with a Poisson equation with analytical solution and with a 2D incompressible flow simulation. A driven cavity flow was adopted as benchmark. Different CFL numbers were used with each Poisson solver. The proposed FAS-MIX method, that uses a mixture of basic methods and Krylov subspace methods showed

the best results. The main conclusion of this work is that multigrid methods, as a convergence acceleration technique, should be used if one wants to develop a high performance code for solving Poisson equation.

## **6. Acknowledgments**

The authors acknowledge the financial support given by FAPESP under grants 03/12612-9 and 04/07507-4.

## **7. References**

- Baker, L., 1991, "More C tools for scientist and engineers", McGraw-Hill.
- Brandt, A., 1977, Multilevel adaptative solutions to boundary values problems, "Mathematics of Computation", Vol. 31, pp. 333–390.
- Briggs, W., 1987, "A Multigrid Tutorial", SIAM, Lancaster Press.
- Ferziger, J. H. and Peric, M., 1997, "Computational Methods for Fluid Dynamics", Springer-Verlag Berlin Heidelberg New York.
- Golub, G. H. and Van Loan, C. F., 1989, "Matrix Computations", Johns Hopkins University Pres.
- Gupta, M. M., Kouatchou, J., and Zhang, J., 1997, Comparison of second- and Fourth-Order Discretizations for Multigrid Poisson Solvers, "J. Computational Physics", Vol. 132, pp. 226–232.
- Hirsch, C., 1988, "Numerical Computation of Internal and External Flows", Wiley & Sons.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., 1997, "Numerical Recipes in Fortran 77", Cambridge University Press.
- Saad, Y., 2003, "Iterative Methods for Sparse Linear Systems", SIAM.
- Spitaleri, R. M., 2000, Full-FAS Multigrid Grid Generation Algorithms, "Applied Numerical Math.", Vol. 32, pp. 483–494.
- Stüben, K. and Trottenberg, U., 1981, "Nonlinear multigrid methods, the full approximation scheme", chapter 5, pp. 58–71, Lecture Notes in Mathematics. Köln-Porz.
- Wesseling, P., 2001, "Principles of Computational Fluid Dynamics", Springer-Verlag.
- Zhang, J., 1996, Acceleration of Five-Point Red-Black Gauss-Seidel in Multigrid for Poisson Equation, "Applied Math. and Comp.", Vol. 80, pp. 73–93.
- Zhang, J., 1997, Residual Scaling Techniques in Multigrid, I: Equivalence Proof, "Applied Math. and Comp.", Vol. 80, pp. 283–303.

# PHYSICAL AND NUMERICAL ANALYSIS OF NAVIER-STOKES ROTOR-STATOR INTERACTION MODELS

**Alex GUEDES**

Instituto de Estudos Superiores da Amazônia. Quadra 32, No. 10, 66640-255. Mangueirão, Belém-PA, Brazil.

**Jean-Louis KUENY**

Laboratory for Hydraulic Machines, Swiss Institute of Technology. Av. de Cours 1007, Lausanne, Switzerland.

**Abstract.** The parallel rotor-stator computation capabilities of Fine Turbo and CFX-TASCflow CFD commercial codes have been investigated within the frame of the European ESPRIT project HPNURSA. Unsteady computations have been performed for simple test cases, where flow conditions are known, the numerical results indicate that the unsteady rotor-stator interaction model is capable of accurately reproduce the features of the incompressible flow encountered in hydraulic turbomachines. It is also shown that R-S computations are very demanding in mesh refinement if small scales flow structures must cross the R-S interface. Finally, the behaviour of an industrial pump-turbine, by VA Tech Hydro, is investigated using CFX-TASCflow. The unsteady computation results were validated by LDV and PIV measurements, showing that unsteady R-S computations could reproduce the experimental flow features for nominal and off-design operation conditions for this pump-turbine.

**Keywords.** Pump-turbine, Rotor-stator, Navier-Stokes, Parallel computation, Numerical method.

## 1 Introduction

Accurate numerical prediction of the flow field of a pump-turbine can allow the design of more compact and silent turbomachine, with an increased operation range, specially for pumping mode, where part load instability are common.

Many authors indicate that rotor stator interaction is paramount for pumping systems at off-design conditions, e.g. Eisele K et al. (1998), Kaupert (1996 and 1997). Recently, different authors analysed the unsteady rotor-stator interaction. For instance, an experimental and numerical analysis of the runner-diffuser interaction in a centrifugal pump has been performed using LDV and PIV by Ciocan et al. (1998 and 2000) and Dong et al. (1997). Other experimental analysis of flow between impeller and bladed diffuser using LDV technique or pressure transducer are presented in Toussaint et al (1998) and Ubaldi et al. (1996).

All these works show the unsteady nature of the rotor stator flow in pump systems, therefore, it is of prime importance to investigate the capability of the rotor-stator interaction models recently implemented in leading CFD commercial codes to represent unsteady phenomena. It is also important to study the numerical characteristics of these codes and its suitability for designing industrial turbomachines.

The following sections show an investigation of rotor-stator computation capabilities of Fine Turbo and CFX-TASCflow CFD commercial codes, developed respectively by NUMECA international and AEA Technology. This is done by two test cases, where flow conditions are known. Then, CFX-TASCflow code is applied for analysing the nominal and off-design behaviour of an industrial pump-turbine designed by VA Tech Hydro. The results for the pump-turbine flow computations are presented and validated by experimental measurements by PIV and LDV techniques.

All the numerical results shown in this work were obtained in the frame of the European ESPRIT project HPNURSA - "High Performance Numerical and Unsteady Rotor Stator Analysis".

## 2. Rotor Stator Interaction Models

Both codes use a sliding surface approach for modelling the rotor-stator interaction, where all interaction occurs over a revolution surface that separates the rotating and non-rotating domain, see Fig. (1). There are three models that differ in the way flow field information is passed across the sliding surface. The first one is the mixing plane model (referenced here as mean stage model), where flow field variables are integrated over tangential direction, so only steady interaction is possible. The second one is the quasi-steady model (referenced here as frozen rotor model), where the appropriated frame changes for vector variables is performed, but rotor-stator relative position is not updated at each time-step. Finally, the unsteady model that accounts for frame and relative position changes and is the only that can be expected to fully reproduce the features of the real rotor-stator flow.

More details of the implementation of these models for CFX-TASCflow can be found in Kuntz (2000).

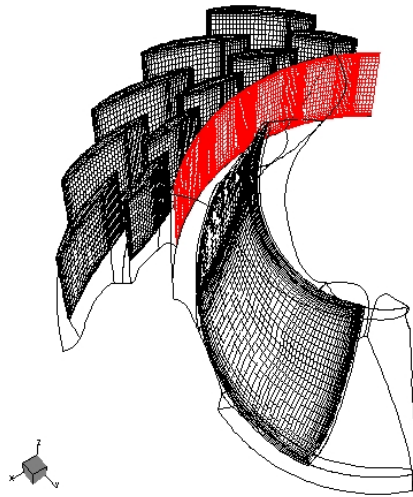


Figure 1. Sliding surface for rotor-stator computations of VA Tech pump-turbine.

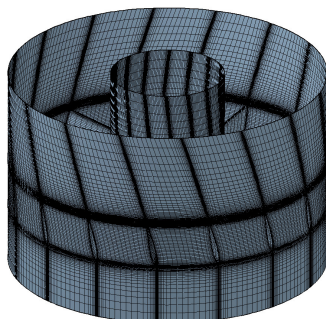
### 3. Rotor-stator models validation - Steady perturbation Test Case

The main objectives of the academic test cases is to control the quality of the Rotor-Stator interfaces implemented in the different CFD codes and to define its numerical limits to ensure a correct transfer of physical quantities through the interface for simplified geometries and flows, but still representative of typical flow in turbomachines.

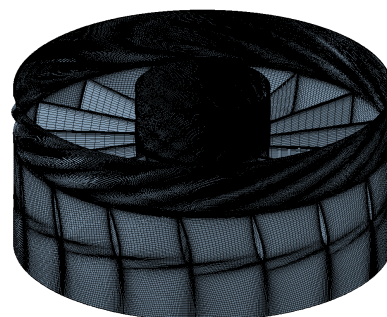
Two steady state test cases have been defined. For these geometries, the perturbation of a uniform flow field is only generated on one side of the interface and gives rise to a steady perturbation in the corresponding static or rotating frame. This steady flow perturbation must be unchanged after and before the interface. We analysed a blade wake crossing the rotor-stator interface and also the upstream pressure perturbation before a blade. A typical axial and radial *turbo machines* geometries have been defined to analyse the numerical flow behaviour in these two classical turbomachine configurations, as found in Henry P. (1992). For these two test cases the numerical unsteady rotor-stator flow will be compared to the steady flow obtained by non rotor-stator computation, with the same geometry and mesh. As the computations for axial and radial test cases present qualitatively the same behaviour and lead to the same conclusions, only results for the axial configuration will be presented.

Six different meshes have been built using ICEM mesh generator or IGG+AUTOGRID, as summarised in Table (1), for ensuring the quality of the numerical results. Some typical meshes are presented on Fig. (2).

Mesh Name	Number of Nodes
RS-1	131175
Axial-Fix (direct or flow aligned)	112791
BIG (flow aligned)	221375
BIG2 (flow aligned)	288015
BIG3 (flow aligned)	238098



A) Axial-directed.



B) Flow aligned mesh

Figure 2. Meshes for steady state test case.

The computations have been performed on a WINDOWS NT computer (Pentium III, 1GHz, 512 M Ram). For a mesh size of about 240'000 nodes, the CPU time per iteration is 0.95 ms, then 2 to 3 CPU days are necessary to obtain convergence if 5 iterations per time step are performed.

Typical results for rotor-stator computations showed the desired continuity and conservation properties of the rotor-stator interface models. But it has been shown that continuity is strongly dependant of mesh refinement in both sides of the R-S interface. For large-scale flow structures, like the one seen in Fig. (3), as for potential effects shown in Fig. (4), the continuity is assured for every rotor stator relative position. For small scale structures, the continuity is only assured for rotor-stator relative positions where mesh is refined enough for capturing the flow structure in both side of the R-S interface, as seen in Fig. (5a), where the refined zones are aligned and the wake starts to cross the R-S interface, on the other hand, for another time step, shown in Fig. (5b), the refined zones are non aligned and the trailing edge wake cannot traverse the R-S interface.

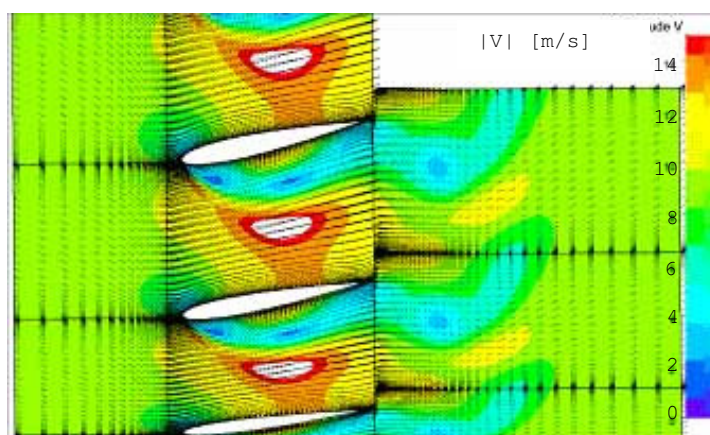


Figure 3. Large vortex crossing the R-S interfaces.(Fine Turbo computation)

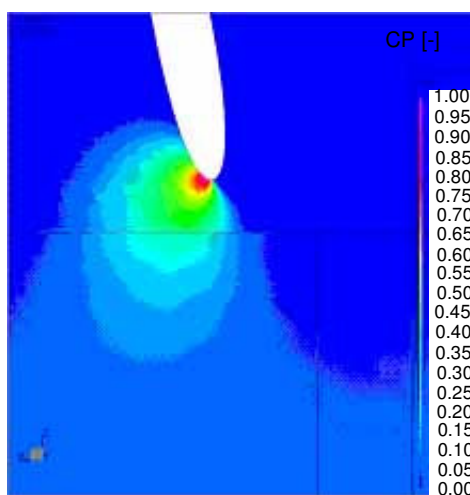


Figure 4. Upstream pressure perturbation crossing the R-S interface. (Fine Turbo computation)

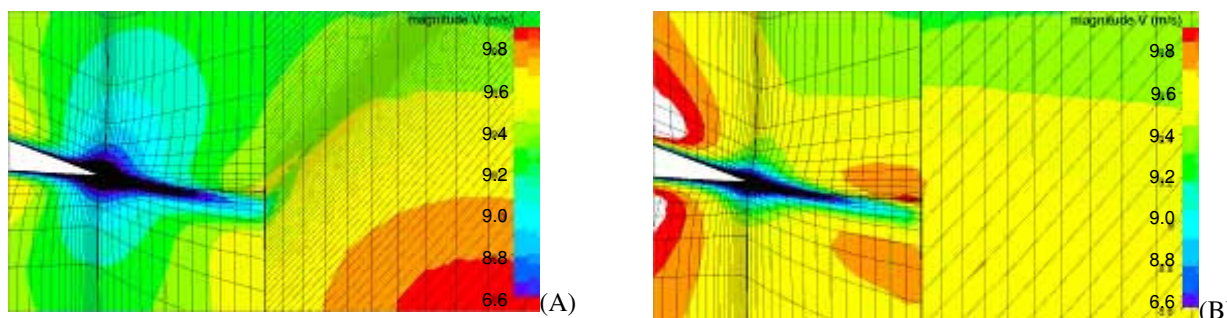


Figure 5. Velocity magnitude field at 2 time steps. (Fine Turbo computation)  
 (A) The wake crosses R-S through the refined mesh in both sides. (B) The wake is diffused by coarse mesh.

It is important to note that the limitation with continuity across the R-S interface for small scale flow structures is a pure mesh refinement effect and is not related to the R-S interaction models implemented in the commercial codes, as this same limitation can be found in non R-S computations as well, but for unsteady computations, the relative R-S position changes at each time step, then the only way to assure good spatial discretization is to have a fine mesh all over the angular extension of the rotating domain. This fact makes very resource intensive when one is interested to catch small-scale structures, because good refinement should be achieved for every R-S relative position. One possible solution is to apply mesh-adapting techniques as the relative position changes.

#### 4. Rotor-stator models validation - Unsteady perturbation test case

An unsteady test case in a rotating frame has also been defined. The unsteadiness corresponds to periodic vortex shedding generated by a square section body in an uniform axial flow field, as can be seen in Fig. (6), with flow parameters based on a classical vortex shedding setup, for witch experiments and multiple CFD solution are available. The simulation is done in 60 deg. channel periodicity; the square side is 0.1m, the Reynolds number is 22000 and the Strouhal number is 0.132. This test case allows verifying possible influence of the rotor-stator interface on unsteady behaviour of the flow, in particular the vortex shedding frequency and the vortices advection.

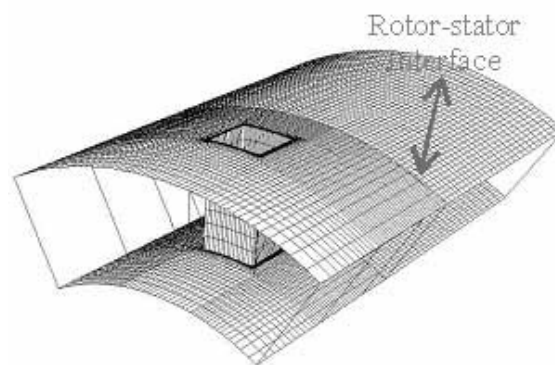


Figure 6. Mesh for the unsteady test case.

The goal of this test is not to reproduce the experimental unsteady features of the flow, as it does not depends on rotor-stator interaction model, but only investigate if the computed unsteady flow field changes as a result of the introduction of the R-S interface. Fig. (7) and Fig. (8) show respectively the instantaneous velocity field for a computation with and without rotor-stator interface, comparing both cases, one can see that the shape and continuity of the vortex structures shaded past the square body is achieved by R-S computation. The shading frequency was exactly the same in both cases as it is driven by the flow conditions around the square body; witch is located entirely in the stator domain and is not influenced by the introduction of the R-S interface.

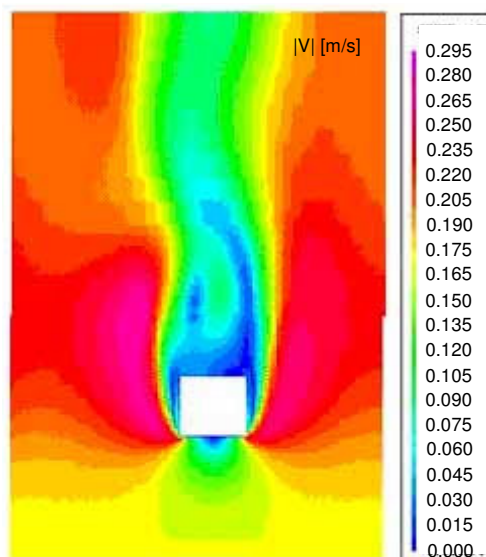


Figure 7. Flow field for unsteady non R-S computation. (CFX-TASCflow computation).

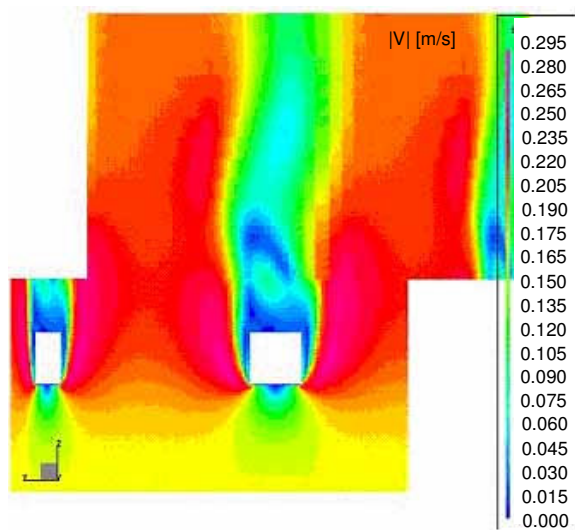


Figure 8. Flow field for unsteady R-S computation. (CFX-TASCflow computation).

The results obtained for steady and unsteady test cases by both Fine Turbo and CFX-TASCflow show that the rotor-stator interaction models implemented in these codes are equivalent to the limits of these tests. Furthermore, considering the results of both test cases together, it was shown that the rotor-stator models implemented are fully conservative and capable of reproducing the typical flow conditions encountered in the rotor-stator interaction zone of a hydraulic turbomachine.

### 5. Pump-turbine analysis

The Francis type pump-turbine studied has specific speed  $N_q=66$ , with 5 runner blades and 22 guide-vanes, as can be seen in Fig. (9). The computations for the pump-turbine were all performed for pumping mode operation, for which this turbomachine displays strong instability behaviour at part-load operation at a guide-vane opening of 18deg., as shown in Fig. (10).

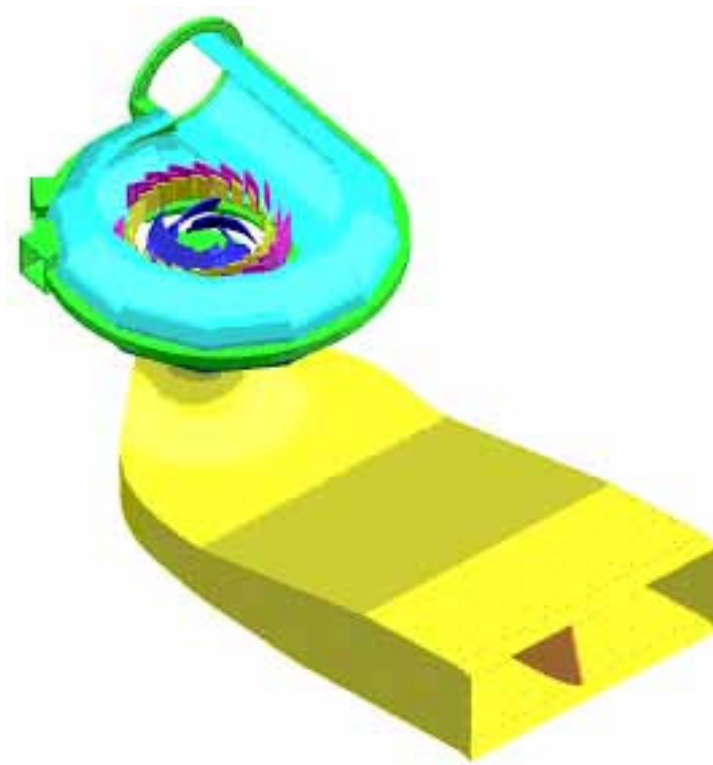


Figure 9. Pump-Turbine geometry.

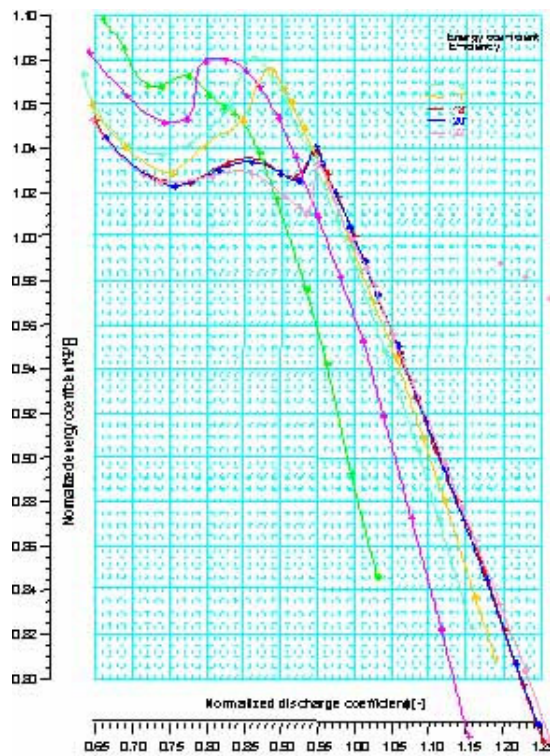


Figure 10. Pumping characteristic curves.

The aim of this study is to show that rotor-stator interaction models implemented in the commercial CFD codes can predict all the complex 3D unsteady turbulent flow features encountered in off-design operation of an industrial pump-turbine, allowing a better understanding of the behaviour of such a turbomachine. To achieve this goal, a numerical simulation campaign, as well as experimental flow field measurements with Particle Image Velocimetry and Laser Doppler Velocimetry techniques were performed over four different operating points, comprising full discharge, nominal and part-load operation.

## 6. Numerical simulations

For computational resources limitation reasons, all the computations for the pump-turbine were performed using CFX-TASCflow CFD software, which is a structured finite volume solver for the incompressible Reynolds averaged Navier-Stokes model, along with the standard  $k-\epsilon$  model for turbulence closure.

The original geometry has not exact periodicity, as we have 5 runner channels and 22 stator channels. A different stator configuration was used for unsteady computation, corresponding to a 22 stator channels, allowing the use of periodicity condition for 1 runner channel and 4 stator channels, reducing greatly the computational resources needed. The final domain configuration for unsteady computations can be seen in Fig. (11).

The boundary conditions for the computations are of steady state type, as summarised in Table (2), so the unsteadiness is generated only by rotor-stator interaction.



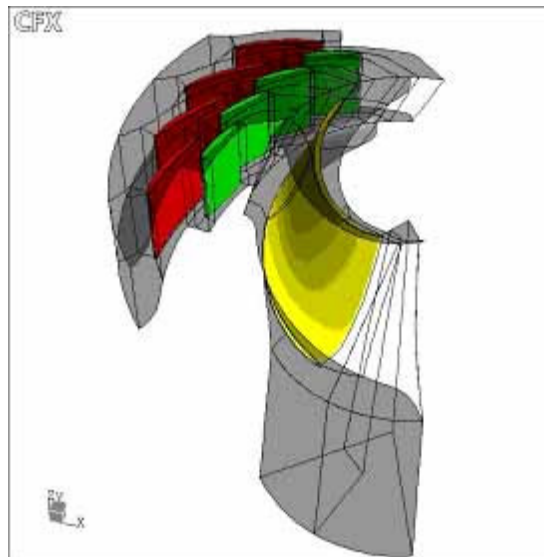


Figure 11. Pump- turbine computation domain.

Four meshes refinements were tested for ensure the non-sensitivity to meshes effects, their names and respective number of nodes is shown in Table (3).

Table 2 Boundary conditions for the Navier-Stokes computations.

Boundary condition	Information supplied
Inlet	Flow rate and direction Turbulence intensity and length scale
Outlet	Average static pressure
Solid walls	Non slip / Log law.

Table 3 Meshes configuration.

R-S Model	Grid Configuration	Number of Nodes
Mean Stage	1 runner channel + 1 stator channel	230 886
Frozen Rotor	1 runner channel + 5 stator channels	356 901
Unsteady	1 runner channel + 4 stator channels	1 019 364

For steady state applications, a suitable time step size is  $0.5/\Omega$ , where  $\Omega$  is the rotational frequency of the computation. For transient simulation, the time scale of the transient effects has to be estimated. Alternatively, it can be expressed by the number of time steps necessary to cover one runner blade passage, which should be not bigger than the number of grid cells in tangential direction over the rotor-stator sliding surface, for achieving a good relationship between spatial and time resolution. This results in the following time step expression:

$$\Delta t = \frac{2\pi}{\Omega} \frac{1}{N_{blades} N_{step}},$$

where  $N_{blades}$  is the number of blades of the machine and  $N_{step}$  the number of time steps per blade passage

## 7. Flow analysis

First it is presented some results for the different operating points in order to assert the flow field change as the flow rate – Q - reduces. The Fig. (12) shows the contours of the pressure coefficient over the blade of the runner four operating points, at the middle of the channel in spanwise (hub to shroud) direction. One can see increasing chocks effects on the leading edge as the flow rate reduces.

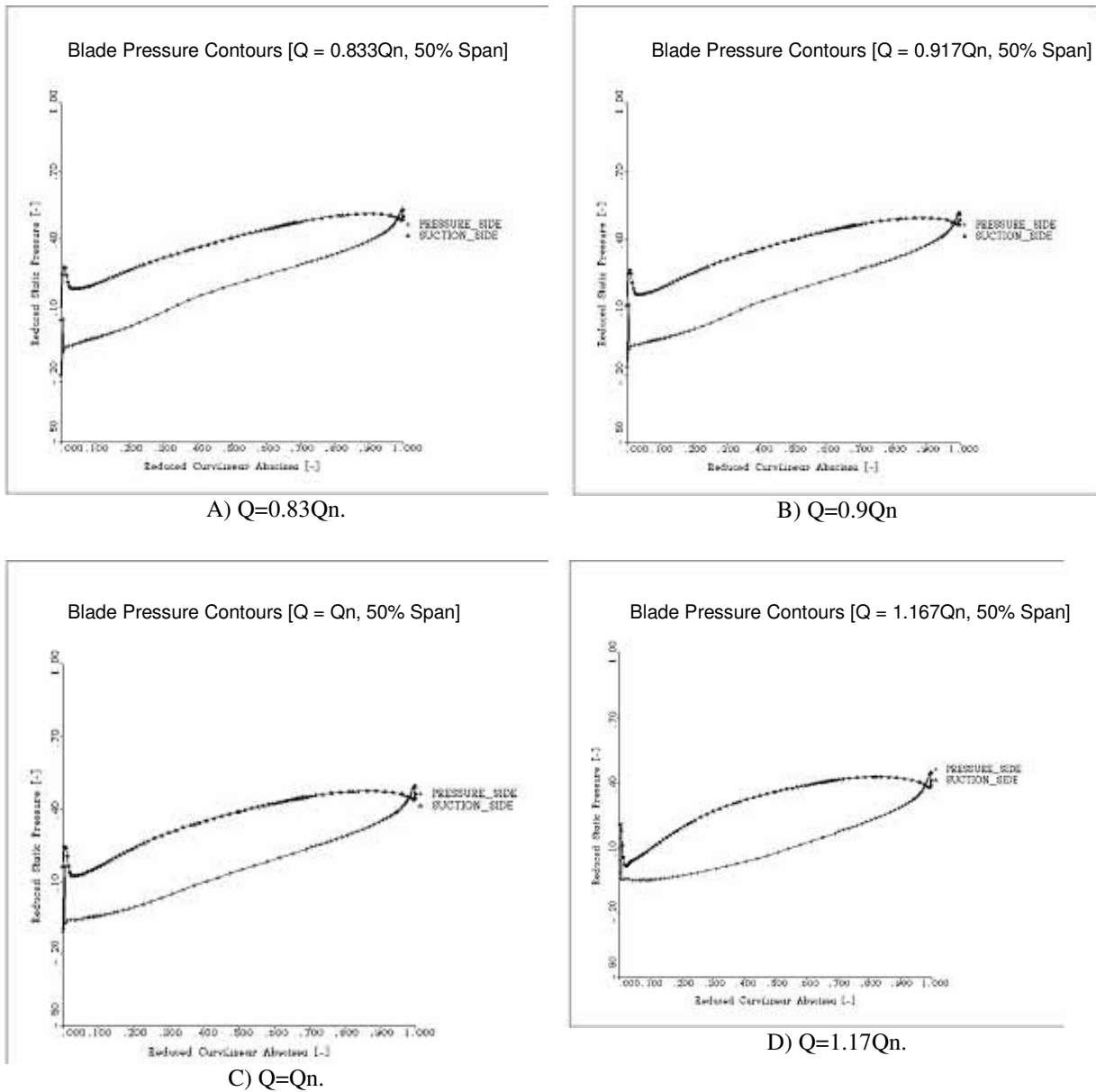


Figure 12. Pressure coefficient at 50% of span.

On Fig. (13) it can be seen that the hydraulic losses over blade to blade surfaces in the runner is concentrated in the shroud side for the operating point  $Q=0.83Q_n$ . This leads to non-adapted flow angles entering the stator domain near the shroud surface.

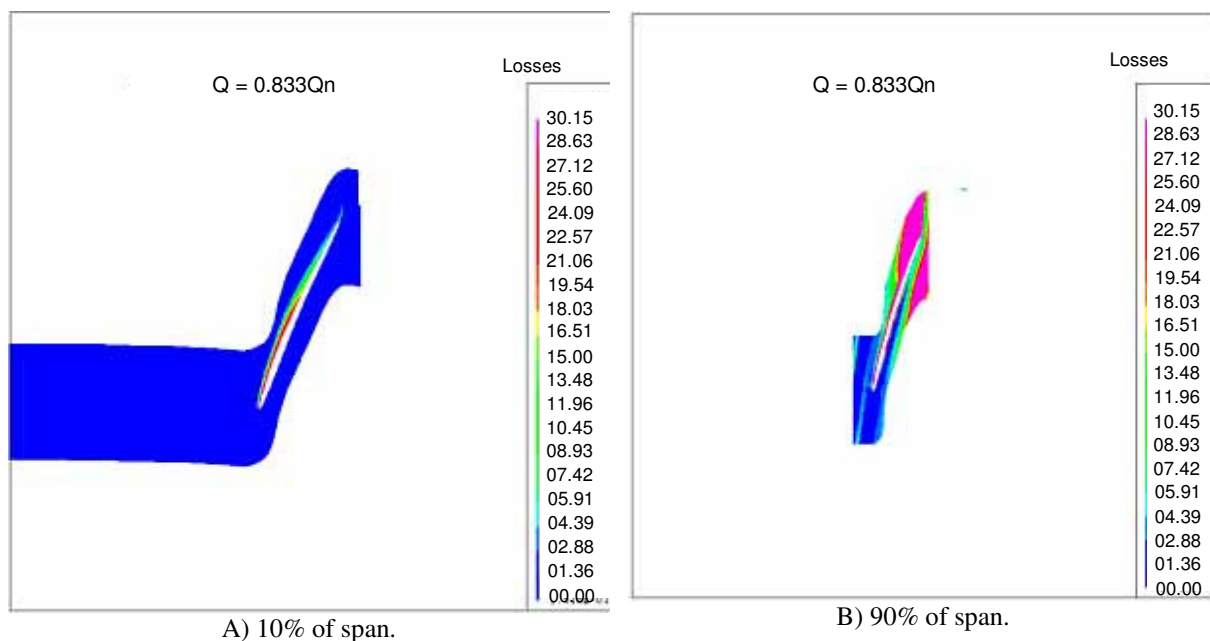


Figure 13. Hydraulic losses for  $Q=0.83Q_n$ .

Another interesting result is seen in Fig. (14). The prediction of the viscous force over the guide vane is strongly influenced by the rotor stator interaction model, as for the mean stage model (Fig. 14A), the direction of the flow entering the stator domain is given by its  $\square$ -integrated value over the rotor-stator sliding surface. This mean flow orientation can greatly differ from the instantaneous flow for a given rotor-stator relative position (Fig. 14B) even for the nominal operating point.

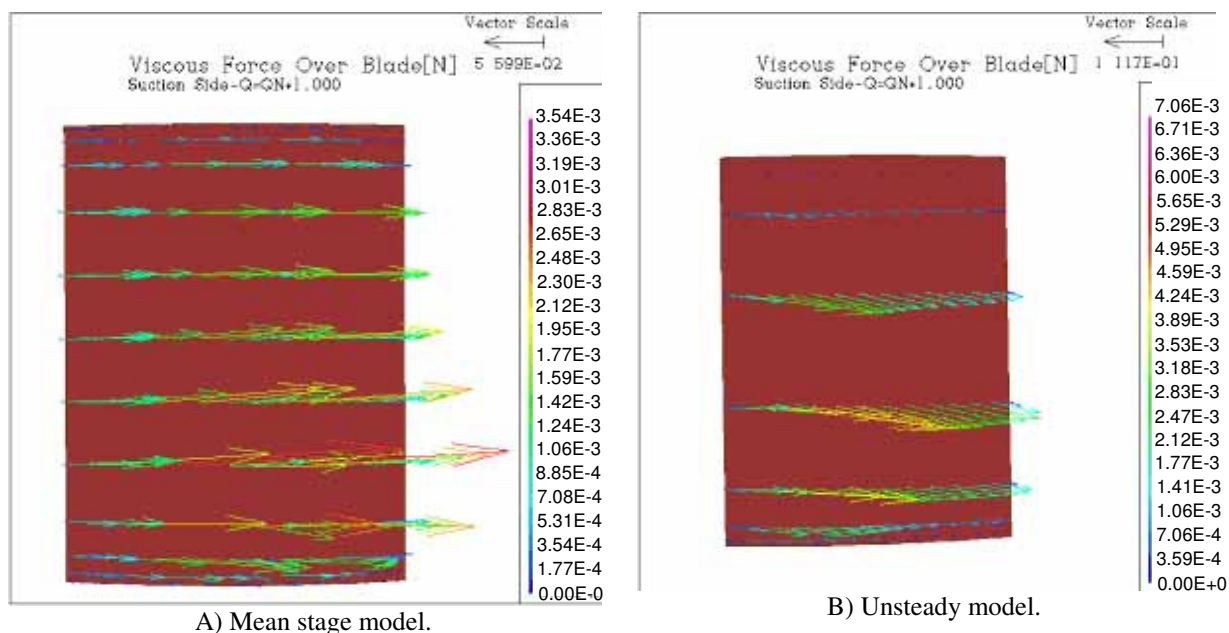


Figure 14. Viscous force over the suction side of the guide blade, coloured by  $|V|$  [m/s].

The instantaneous absolute speed colour map displayed on Fig. (15) shows that the flow field in the stator zone is highly non-uniform, with noticeable changes from one channel to another. For both operating points  $Q=1.17Q_n$  (Fig. 15A) and  $Q=Q_n$  (Fig. 15B) it can be seen that, for a given R-S relative position, the flow around the leading edge of the guide vanes has good or bad incidence according to the channel considered; it can be seen as well that there is a low speed zone close to the trailing edge of the stay-vanes, show a tendency of flow detachment. These flow non-uniformities are more important for off-design operation conditions (Fig. 15A) when compared to the nominal operation point (Fig. 15B).

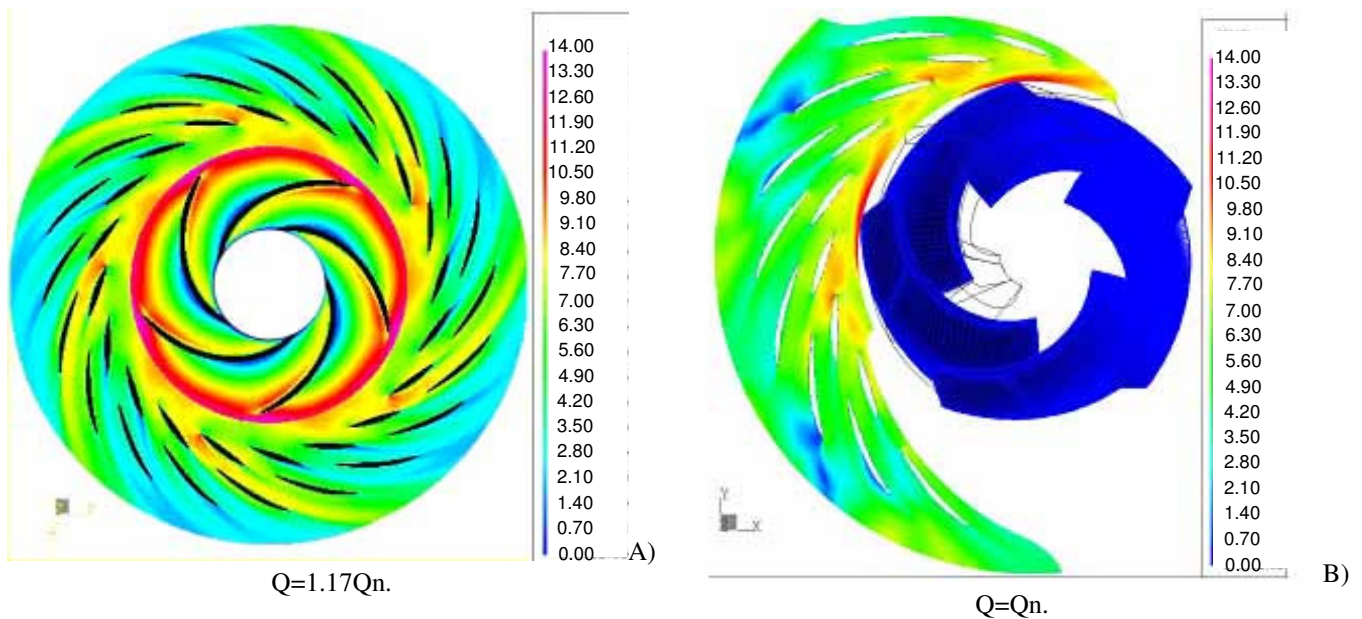


Figure 15. Absolute speed on the stator at mid-span [m/s].

An experimental measurements campaign has been carried-out, using PIV and LDV techniques in order to obtain the instantaneous flow field in the rotor stator interaction zone, allowing a better understanding of the flow features in this zone, as well as validating the numerical computations. All the details of the experimental work are described in Ciocan (1998).

The raw results of the experimental campaign have been treated in order to represent them into the same post-processing software used to analyse the numerical results, this approach allowed a direct comparison and validation of the unsteady numerical computations.

A direct comparison of the instantaneous PIV flow field near the leading edge of the guide vanes, for two adjacent stator channels, shown in Fig. (16) A and B, to the results of the Fig. (15B) indicates that the unsteady rotor stator computation was able to accurately predict the flow behaviour at the rotor stator interaction zone, as the magnitude of the measured flow velocity and its distribution over the stator channels matches very closely the numerical predictions. A further indication of the good accuracy of the computations can be seen at Fig. (17), where the LDV measured (Fig. 17 A) and computed (Fig. 17 B) velocity vector fields near the leading edge of the guide vanes agree very closely both, in magnitude and direction.

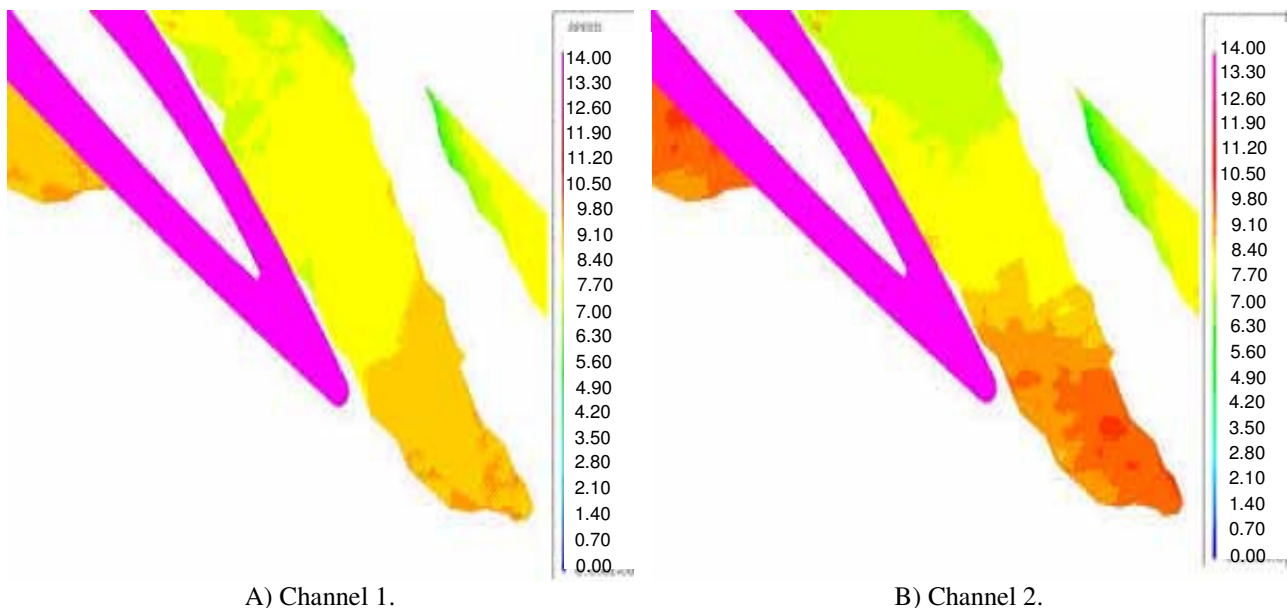


Figure 16. Instantaneous PIV velocity magnitude near guide vane leading edge - 50% span;  $Q=Q_n$ .

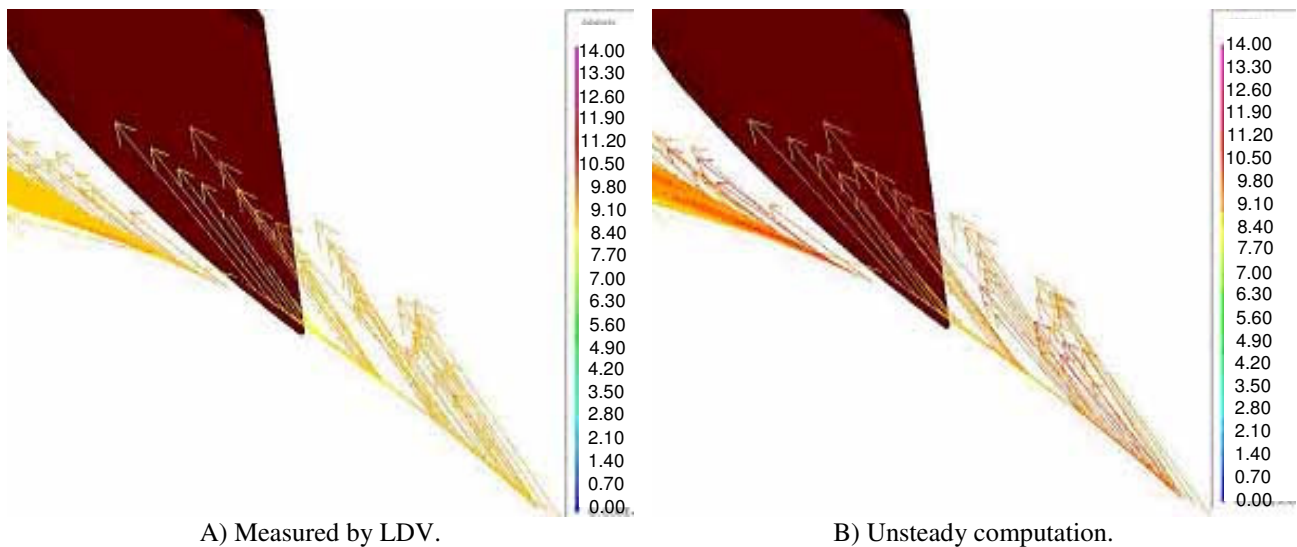


Figure 17. Instantaneous velocity vectors - 50% span ;  $Q=Q_n$ .

The prediction of the flow features in unstable operation zone is of great importance for pump-turbine design. The experimental measurements of the flow field in the rotor-stator interaction zone shown no signs of rotating stall nor runner inlet pre-rotation for the part-load operating points considered in this study. The unstable operating zone is characterised by a strongly non uniform velocity profile, for both spanwise and tangential direction, at the outlet of the runner, as a result, for the VA Tech Pump-turbine, the hub side of the stator channel is dominated by a large flow detachment at the suction side of the guide-vanes, that occupies the entire tangential extension of the channel, originating important energy losses that could explain the drop in pump efficiency. This behaviour could be accurately predicted by the unsteady computation, as it can be seen in Fig. (18), comparing the LDV instantaneous flow field (Fig. 18A) with the corresponding numerical simulation. It should be noted that the recirculating zone predicted by numerical simulation does not occupies the entire tangential extension of the stator channel, this could be due to the larger cross sectional area of the stator channel, as it counts only 20 guide-vanes instead 22 for the original geometry. It suggests that the exact dynamic behaviour of the flow in the unstable operating zone could only be reproduced if the entire 360 deg. domain is computed.

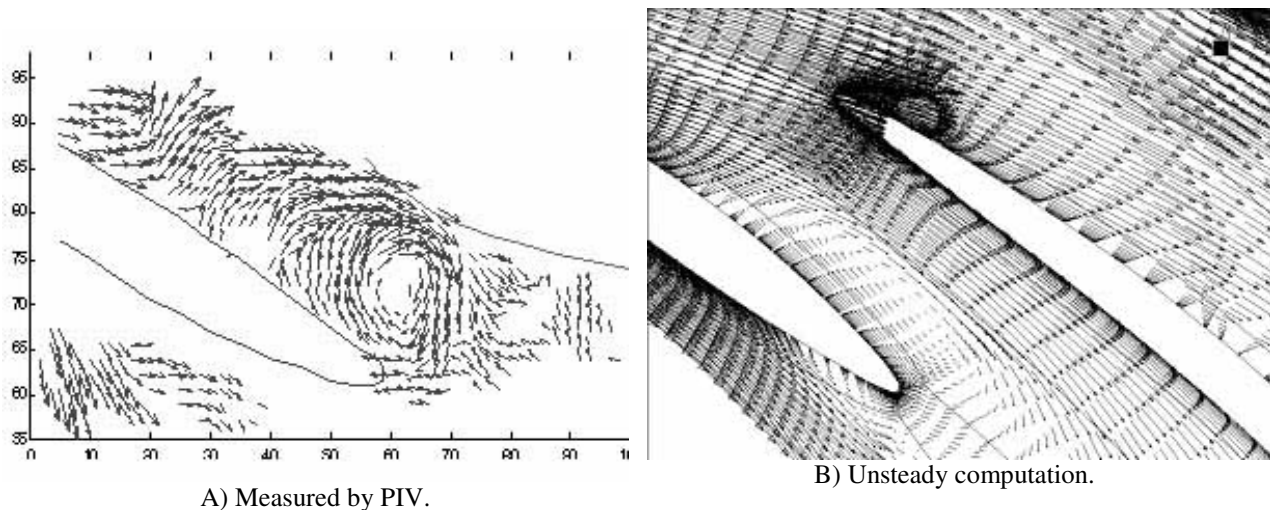


Figure 18. Instantaneous velocity field near trailing edge of guide vanes at 10% span –  $Q=0.83Q_n$ .

## 8. Conclusions

A detailed study of the rotor stator capabilities implemented into Fine Turbo and CFX-TASCflow CFD codes has been performed. According to the steady and unsteady test cases, it was shown that the unsteady rotor stator interface models implemented in both codes are capable to accurately represent the different flow features that are founded in the frame of incompressible turbomachines. All rotor-stator interaction effects: potential, viscous or pure advective ones can be expected to be reproduced by the unsteady R-S model.

It was demonstrated that the sliding surface approach adopted by both commercial codes is very demanding in terms of mesh and time step sizes, leading to computation set-ups with several millions of nodes, even for simple geometries, in order to be able to capture small scale effects, like viscous wakes. In other hand, if the main flow phenomena are large scale ones, like the velocity field non uniformities in the runner outlet of the pump-turbine, this approach can be used for accurately simulate the flow field with reasonable meshes sizes.

The analysis of the pump-turbine behaviour showed that the flow field leaving the runner is highly non-uniform, both in hub to shroud direction and over the tangential direction at the outlet of the runner, even for operating conditions close to the nominal point. As a result, for off-design operating points, the flow entering one channel of the stator is strongly influenced by the rotor stator relative position, and even for an given channel, one can find well adapted flow over the leading edge of the guide vane near the shroud surface and bad adapted flow near the hub surface. This behaviour of the flow field in the rotor-stator zone leads to a complex flow pattern into the stator channels, which cannot be reproduced accurately by steady state rotor-stator interaction models. The unsteady computations showed good accuracy, when compared to experimental results, displaying qualitatively the same flow features found by the measurements and indicating the way the instability of machine is induced. In spite of that fact, the exact behaviour of the machine could not be predicted in the instability zone, this limitation can be linked to geometry modification applied to the stator domain (20 channels instead of 22) and the use of the periodicity boundary conditions. If one were interested on predicting the exact unstable behaviour of the machine, a full 360 deg. Computations would be necessary.

## 9. Acknowledgement

The authors wish to acknowledge the support by the European Community and the Swiss Federal Office for Education and Science in the frame of the ESPRIT project HPNURSA. The CFD Quality and Trust Network, QNET, also supports this initiative.

## 10. References

- Ciocan, Gabriel Dan "Contribution à l'analyse des écoulements 3D complexes en turbomachines" Phd Thesis, INPG, Grenoble, 1998
- Ciocan, Gabriel Dan, Avellan, François, Kueny, Jean-Louis, Optical Measurement Techniques for Experimental Analysis of Hydraulic Turbines Rotor-Stator Interaction, Proceedings (CDRom) of ASME Fluids Engineering Division Summer Meeting, June 11-15 2000, Boston, Massachusetts, USA
- Dong R., Chu S., Katz J. "Effect of modification to tongue and impeller geometry on unsteady flow, pressure fluctuations, and noise in a centrifugal pump" *Journal of Turbomachinery*, vol. 119, p.506-515, 1997
- Eisele K., Muggli F., Zhang Z., Casey M.V., "Ecoulement instationnaire dans un diffuseur aubé d'une pompe centrifuge", *La Houille Blanche* no. 3/4, p. 23-30, mai 1998
- Henry Pierre *Turbomachines Hydrauliques*, Lausanne: Presses Polytechniques et Universitaires Romandes, 1992
- Kaupert K., Holbein P., Staubli T., "A first analysis of the flow field hysteresis in a pump impeller", *Journal of Fluid Engineering*, vol. 118, p. 685-691, december 1996
- Kaupert K., Staubli T., "Transient hysteresis in a centrifugal pump characteristic and associated unsteady flow field", ASME Fluid Engineering Division Summer Meeting, USA, 22-26 June 1997
- Kuntz, M., Garcin, H., Guedes, A., Menter, F., Parkinson, E., Kueny, J.L., 2000, "Developments of CFX-TASCflow for unsteady rotor-stator simulations in the frame of the European project HPNURSA", CFX Users Conference 2000, Munich, Germany
- Toussaint M., Hureau F., Lapray J-F, "Influence des diffuseurs aubés sur le fonctionnement des pompes centrifuges ", *La Houille Blanche* no. 3/4, p.38-44, mai 1998
- Ubaldi M., Zunino P., Barigozzi G., Cattanei A., "An experimental investigation of stator induced unsteadiness on centrifugal impeller outflow", *Journal of Turbomachinery*, vol. 118, p. 41-54, January 1996

## DESENVOLVIMENTO DE UMA INTERFACE GRÁFICA PARA DINÂMICA DOS FLUIDOS COMPUTACIONAL, UTILIZANDO A METODOLOGIA DE FRONTEIRA IMERSA

**Gustavo Prado de Oliveira**

UFU / FEMEC / LTCM

[gpoliveira@mecanica.ufu.br](mailto:gpoliveira@mecanica.ufu.br)

**João Marcelo Vedovoto**

[vedovotojm@yahoo.com](mailto:vedovotojm@yahoo.com)

UFU / FEMEC / LTCM

**Ana Paula Fernandes**

UFU / FEMEC / LTCM

[apfernandes@mecanica.ufu.br](mailto:apfernandes@mecanica.ufu.br)

**Aristeu da Silveira Neto**

UFU / FEMEC / LTCM

[aristeus@mecanica.ufu.br](mailto:aristeus@mecanica.ufu.br)

Universidade Federal de Uberlândia – Faculdade de Engenharia Mecânica – Av. João Naves de Ávila, 2160 - Campus Santa Mônica - Bloco 1M - Uberlândia/MG - CEP: 38400-902

Resumo: Atualmente, o Laboratório de Transferência de Calor e Massa e Dinâmica dos Fluidos - LTMC/FEMEC/UFU possui diversos trabalhos para simulações e análise computacional da dinâmica dos fluidos, porém com etapas providas por softwares proprietários, tornando onerosa a utilização de tais soluções. Além disso, a ferramenta atualmente funcional existente, que efetua tais cálculos, foi desenvolvida na linguagem Fortran, desprovida de uma interface gráfica, além de não permitir uma visualização prévia das geometrias a serem trabalhadas, o que dificulta o manuseio da mesma. Dessa forma, no presente trabalho pretende-se desenvolver uma ferramenta para a construção de geometrias e a geração de suas malhas de elementos finitos triangulares superficiais, utilizando-se de softwares livres, para posterior importação em códigos computacionais dedicados a análise computacional da dinâmica dos fluidos, através do método de fronteira imersa. Dentre as ferramentas analisadas para a modelagem da geometria e geração da malha, optou-se pela utilização do Gmsh, por ser esse um software livre, portátil para as plataformas Linux e Windows, intuitivo e de fácil aprendizado. Já dentre as ferramentas analisadas para o desenvolvimento da Interface gráfica, optou-se pela QT, por ser uma aplicação compreensiva para desenvolvimento de framework em C++. Isto inclui uma biblioteca de classes e ferramentas para o desenvolvimento e migração de plataforma, além de ser também portátil para Windows e Linux. O trabalho apresentará algumas soluções de geometrias / malhas geradas no Gmsh, e a preparação dessas soluções para a utilização do método de fronteira imersa na análise computacional da dinâmica dos fluidos em geometrias como esferas, cilindros, aerofolios, buscando-se utilizar os softwares para trabalhar com formas mais complexas como aeronaves e automoveis.

*Palavras chave: fronteira imersa, geração de malhas, software livre*

### 1. INTRODUÇÃO

Atualmente, o Laboratório de Transferência de Calor e Massa e Dinâmica dos Fluidos - LTMC/FEMEC/UFU possui diversas ferramentas para simulações e análise computacional da dinâmica dos fluidos. No entanto, para efetuar tais simulações são necessários softwares proprietários (requerem registro) o que dificulta o desenvolvimento de aplicações acadêmicas devido ao custo dos mesmos.

Além disto, a ferramenta de mecânica dos fluidos encontrada no LTCM, não oferece uma interface amigável de forma a facilitar o manuseio da aplicação. Para tal, será utilizadas técnicas de programação visando criar uma **interface gráfica (GUI, Graphical User Interface)**, que é uma ferramenta de interação entre usuário e sistema computacional baseada em símbolos visuais, como ícones, menus e janelas. Seguindo esta linha de desenvolvimento, pode-se oferecer uma ferramenta simples para resolver problemas acadêmicos com entradas paramétricas e capacidade avançada de visualização. Com isso pode-se delimitar visualmente as etapas de pré e pós-processamento, gerando arquivos de saída que possam ser usados com os códigos existente no LTCM, possibilitando maior facilidade de uso.

Juntamente com o problema de interação computacional, outro grande desafio em engenharia é a solução numérica da ação de forças sobre corpos imersos em escoamentos, com suas possíveis deformações e deslocamentos, para tal problema uma solução viável pode ser obtida através da modelagem matemática.

Evidentemente, é notório o crescimento dos recursos computacionais, viabilizando a realização de simulações numéricas cada vez mais aperfeiçoadas de escoamentos em domínios de topologia diversas. Em geometrias complexas, a escolha apropriada da malha computacional depende da metodologia de discretização do domínio contínuo. Desta forma, um algoritmo desenvolvido, por exemplo, para lidar com malhas curvilíneas ortogonais, não é capaz de lidar com malhas não-ortogonais. Assim, procura-se lidar com o problema, empregando malhas que se ajustam à sua topologia, quer seja acompanhando a superfície do corpo a ser estudado, quer seja se adaptando melhor ao escoamento, melhorando a qualidade da solução.

Escoamentos ao redor de geometrias complexas ainda apresentam sérias dificuldades em serem resolvidos, pois as geometrias apresentam problemas na geração da malha computacional, que nem sempre são triviais, e são propensa a inserir severos erros nos balanços entre seus nós elementares. Isto acontece quando se procura ajustar a malha computacional ao objeto de estudo e ao domínio, nas discretizações.

Nos casos de discretizações em blocos, o acoplamento entre os diversos domínios, se mal construído, pode implicar em sérias inconsistências físicas. A discretização de domínios tridimensionais exige grande capacidade de armazenamento e processamento de dados que, se somados aos freqüentes refinamentos locais das malhas, torna os cálculos muito caros.

Uma metodologia alternativa para se resolver escoamentos sobre geometrias complexas e móveis vem sendo desenvolvida no LTCM. Trata-se da metodologia de Fronteira Imersa (*Immersed Boundary - IB*). Neste método, a presença de uma interface fluido-sólido, ou mesmo uma interface fluido-fluido, pode ser simulada pela adição de um termo de força (que alguns autores denominam como termo forçante) às equações do movimento do fluido. Dessa forma, o escoamento reconhece a presença de um objeto imerso, por exemplo, apenas pela ação de um "campo de força". A forma como este campo de força é calculado diferencia a metodologia empregada atualmente. Cabe salientar que tais forças são formadas a partir de efeitos diversos, agindo ao mesmo tempo no fluido, em decorrência de cisalhamentos e deformações, variações na quantidade de movimento e gradientes de pressão. Silva(2002), Campregher(2005)

O uso da metodologia *IB* requer uma malha cartesiana para a solução do escoamento, independentemente da complexidade e da mobilidade da fronteira imersa. No entanto, requer-se a construção de uma malha superficial para a interface fluido-sólido. Empregando-se uma malha triangular para a discretização desta interface, torna-se possível aproveitar as facilidades associadas a este tipo de malha, para a geração de geometrias complexas.

Para tanto, no presente trabalho optou-se por utilizar o software livre Gmsh como ferramenta geradora desta malha, devido a facilidade de criar geometrias complexas em formatos padrões como o STL. Além disso mostrar-se-á que geometrias geradas com essa ferramenta apresentam resultados compatíveis com outras geometrias geradas por softwares proprietários de renome, apresentando algumas soluções de geometrias/malhas geradas no Gmsh.

Dentre as ferramentas analisadas para o desenvolvimento da Interface gráfica, optou-se pelo QT, por ser uma aplicação compatível para desenvolvimento de framework em C++. Isto inclui uma biblioteca de classes e ferramentas para o desenvolvimento e migração de plataforma, além de ser também portátil para Windows e Linux.

## 2. GMSH

Gmsh é um gerador automático de malhas (primariamente de Delaunay) tridimensionais em elementos finitos com facilidades de pré e pós-processamento. Ele foi desenvolvido para oferecer uma ferramenta simples para resolver problemas acadêmicos com entradas paramétricas e capacidade avançada de visualização. Uma de suas principais características é a facilidade em respeitar a regularidade na geração de malhas adaptativas em linhas, superfícies e volumes, agrupando estas estruturas uma malha única.

Gmsh está construído com base em quatro módulos, dentre os quais, os de interesse para este projeto são: geometria e malha (módulos utilizados na construção de malhas de geometrias). Todas essas instruções são prescritas utilizando-se de interface gráfica de usuário (GUI) ou em arquivos de dados ASCII usando a própria linguagem script do Gmsh. Ações interativas geram bits de linguagens em arquivos de saída e vice-versa, o que torna possível automatizar todo tratamento, utilizando *loops*, quanto as chamadas de sistemas externos e condicionais.

Geometrias são criadas pela definição sucessiva de pontos, linhas orientadas, superfícies orientadas e volumes, compondo grupos de entidades geométricas que podem ser definidas, baseando-se nestas entidades geométricas. A linguagem *script* do Gmsh permite que toda entidade geométrica seja totalmente parametrizada.

Malha de elementos finitos viabiliza a construção de um dado subconjunto de espaços tridimensionais por elementos de várias formas geométricas elementares (linhas, triângulos, quadrados, tetraedros, prismas, hexaedros, e pirâmides), arranjos de tal forma que se duas destas formas se interceptam, eles formam uma face, uma borda ou um nó. A geração de malha segue a mesma performance da criação de geometria: define-se as coordenadas de cada ponto



no eixo cartesiano; os pontos são usados para definir linhas existente entre eles; limitada pelas linhas, define-se uma superfícies, então, através das superfícies pode-se definir uma geometria complexa para se gerar a malha e também os volumes.

Solver externos podem ser inter-relacionados com Gmsh, através do Unix ou sockets TCP/IP, o que permite encaminhar facilmente cálculos externos para coletar e aproveitar resultados de simulação dentro do módulo de pós-processamento do Gmsh.

### 3. Fatores de qualificação do Gmsh como gerador de malha:

- Descrição simples e rápida de geometrias, graças a funções de usuário definidas: *loops*, funções condicionais e bibliotecas;
- Parametrização de geometrias. A linguagem *script* do Gmsh faz com que todos os comandos e seus argumentos dependam de cálculos prévios;
- Gera com simplicidade 1D, 2D e 3D;
- Gmsh oferece vários mecanismos para controlar o tamanho dos elementos na malha final: através de interpolação de comprimento característico de pontos geométricos;
- Cria exportações simples de geometrias e malhas;
- Interação com *solver* externo. Gmsh oferece interfaces C, C++, Perl, Python, e outras linguagens que podem ser facilmente adicionadas;
- Visualização de resultados computacionais de grande variedade;
- Exporta plotagem em diferentes formatos: Látex, PNG, JPEG, STL;
- Gera animações complexas;
- Executável em várias plataformas (Windows, Mac e Unix);
- É gratuito.

### 4. Linguagens de programação

Como a principal preocupação, em achar um gerador de malha que acatasse as características necessárias para resolver os problemas físicos, foi resolvida, a primitiva agora é valer-se de uma ferramenta de programação capaz de dar suporte a ferramenta geradora de malha escolhida, para que se possa desenvolver uma ferramenta capaz de efetuar os cálculos necessários para problemas de interação fluido-estrutura.

Para tanto, é necessário ressaltar que, para desenvolver uma interface de uso amigável e de boa aparência, os melhores tipos de linguagem de programação, são aquelas que trabalham com desenvolvimento de aplicações visuais de interface com o usuário. Para isso, tal ferramenta deve se valer de uma Biblioteca de componentes visuais (VCL – *Visual Component Library*) consistindo de objetos reutilizáveis, incluindo objetos padrões de interface com o usuário, gerenciamento de dados, gráficos e multimídia, gerenciamento de arquivos e quadros de dialogo padrão.

Como a interface gráfica terá que ser baseada na plataforma Linux(plataforma base do cluster utilizado no LTCM), pode-se descartar facilmente a linguagem Delphi, que apesar de se enquadrar perfeitamente nas características acima citadas, esta ferramenta está disponível somente para Windows, além de não ser uma ferramenta gratuita. Com isso, uma opção que seria plausível é o Kylix, que é um Delphi criado para plataforma Linux, porém, seus direitos autorais são de poderio da Borland, o mesmo proprietário do Delphi, sendo assim, Kylix é ferramenta que onera custos.

Na busca por ferramentas visuais que tenham como plataforma o sistema operacional Linux, encontrou-se uma ferramentas que apresenta uma relevância muito importante, tanto quanto o próprio Kylix. O Qt foi escolhido para ser utilizado, por ser uma aplicação compreensiva para desenvolvimento de framework em C++. Isto inclui uma biblioteca de classes e ferramentas para o desenvolvimento e migração de plataformas, além de possuir uma ferramenta gráfica chamada QtDesigner que é uma poderosa construtora de *layouts* e *forms* com características GUI, a qual está hábil ao desenvolvimento rápido de interfaces de usuários com alta performance com visual nativo e suportado em várias plataformas(Windows, Linux/Unix, Mac OS X).



Fig. 4.1- Escalabilidade de plataformas para QT

## 5. Resultados

### 5.1 Geometrias geradas: esfera, cilindro, trem de pouso, aeronave

Com o Gmsh pôde-se obter geometrias simples como mostrado na Fig. 5.1, geradas com bastante agilidade se comparado ao uso de pacotes comerciais. Levando-se em conta que para atividades acadêmicas esses tipos de geometrias podem ser utilizadas facilmente para realizar simulações.

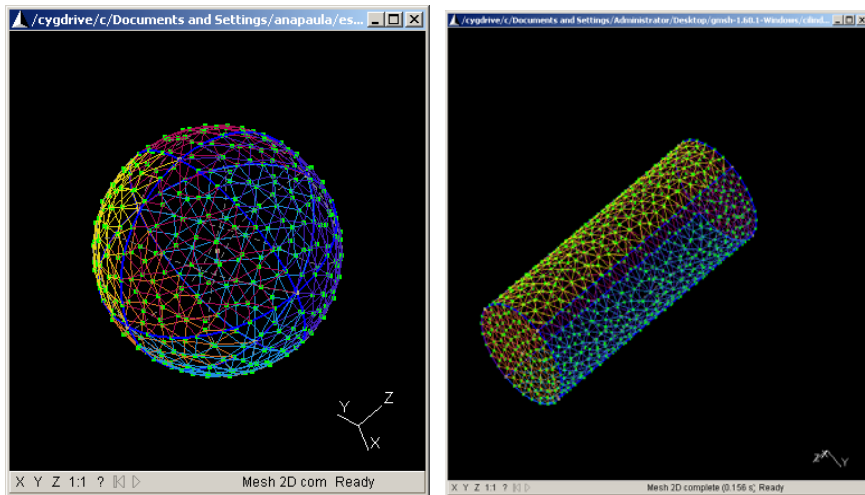


Fig 5.1 – Geometrias simples geradas no Gmsh

Em se tratando de geometrias mais complexas como mostrado na Fig. 5.2 e Fig. 5.3, a tarefa de gerá-las no Gmsh torna-se um pouco mais complicada, por exigir maior destreza e habilidade em criação de figuras 3D. Apesar de ser uma tarefa mais difícil, não é impossível de ser efetuada, porém necessita-se de uma quantidade maior de tempo para obter sucesso.

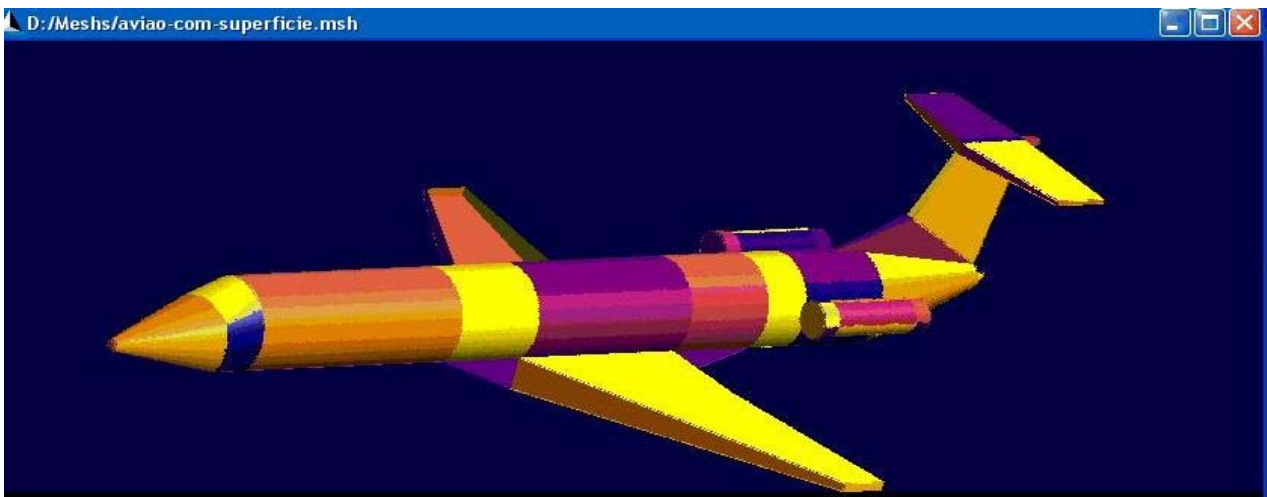


Fig. 5.2 Malha de um avião gerada com Gmsh

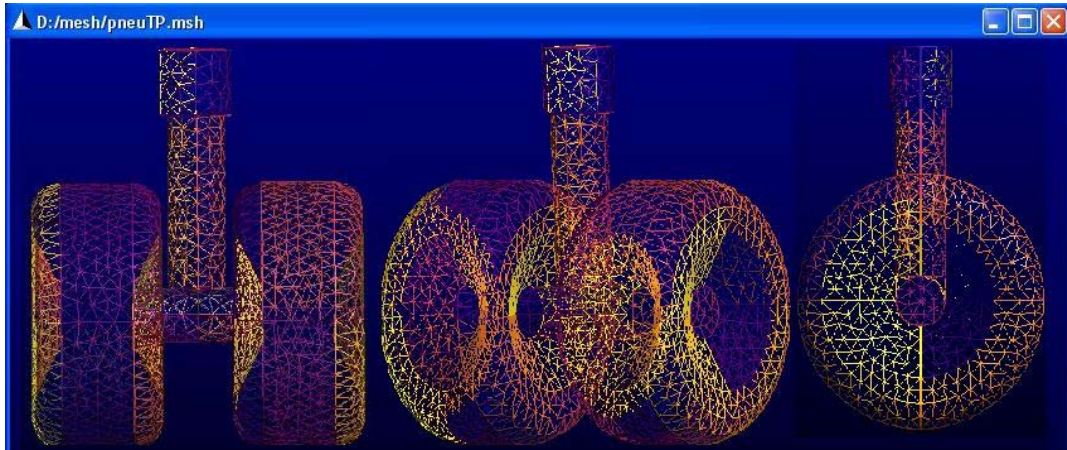


Fig. 5.3 – Malha de um trem de pouso gerada com Gmsh

## 6. Validando as malhas geradas pelo Gmsh

Para analisar a topologia do escoamento iremos utilizar de alguns parâmetros, efetuando comparações dos resultados ao utilizar o Gmsh para gerar uma esfera e resultados utilizando pacotes comerciais para a mesma geometria. Um importante parâmetro que pode ser empregado é o critério  $Q$ , Eq. 6.1 (Jeong e Hussain, 1995), definido da forma:

$$Q = \frac{1}{2} [|\Omega|^2 - |S|^2] > 0 \quad (6.1)$$

, onde  $S$  e  $\Omega$ , são as componentes simétrica e assimétrica do gradiente de  $u$ , portanto  $Q$  representa o balanço local entre a taxa de deformação e a vorticidade, ou seja, a norma Euclidiana para a qual o tensor vorticidade supera o tensor deformação (Haller, 2005), definidos respectivamente como:

$$\Omega = \frac{1}{2} [\nabla v - (\nabla v)^T] \quad (6.2)$$

$$S = \frac{1}{2} [\nabla v + (\nabla v)^T] \quad (6.3)$$

Portanto o critério  $Q$  é uma das várias ferramentas usadas para analisar as propriedades das estruturas turbilhonares. Sua principal vantagem é mostrar as estruturas que adquirem mais peso devido ao efeito rotacional, permitindo visualizar as regiões do escoamento com fortes deformações do fluido.

O parâmetro adimensional número de Reynolds ( $Re$ ), Eq. 6.4, foi empregado para caracterizar o regime de escoamento do fluido sobre a esfera. O  $Re$  foi definido baseado no diâmetro  $D$  da esfera na forma:

$$Re_D = \frac{\rho U_\infty D}{\mu} \quad (6.4)$$

onde  $U_\infty$  é a velocidade da corrente livre,  $\rho$  e  $\mu$  são a massa específica e a viscosidade dinâmica do fluido, respectivamente.

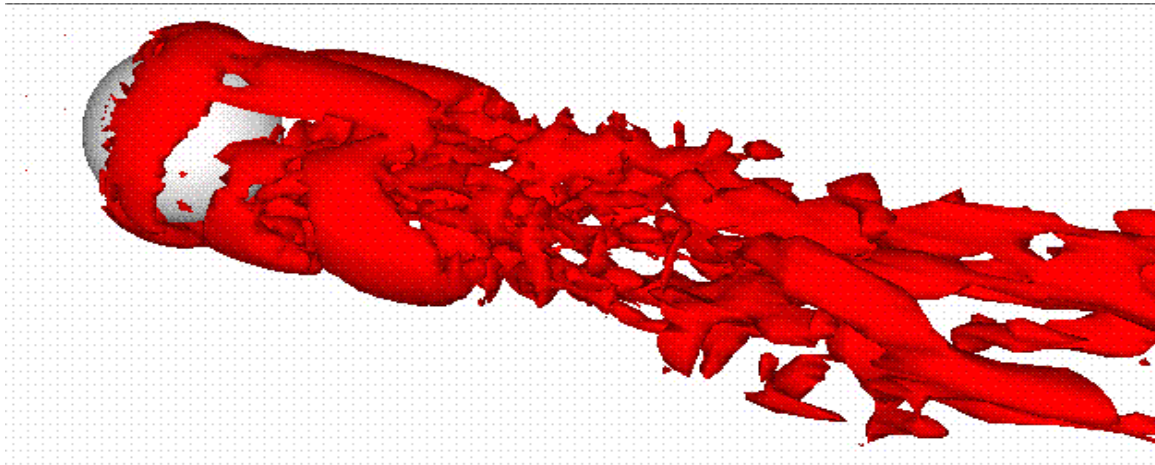


Fig. 6.1 - Detalhe dos vórtices sobre a esfera a  $Re = 1.000$  evidenciado pelo critério Q.

A visualização da Fig. 6.1 a cima foi realizada com o isovalor do critério  $Q = 10$  para uma esfera gerada pelo Gmsh, sendo esta com elementos da malha quase todos de mesmo tamanho, já a Fig. 6.2 abaixo, foi retirada de Campregher(2005) e gerada com uma malha de elementos triangulares isósceles.

O coeficiente de arrasto é definido abaixo, Eq. 6.5:

$$C_D = \frac{F_x}{\left(\frac{1}{2}\right)\rho U_\infty^2 \left(\frac{\pi D^2}{4}\right)} \quad (6.5)$$

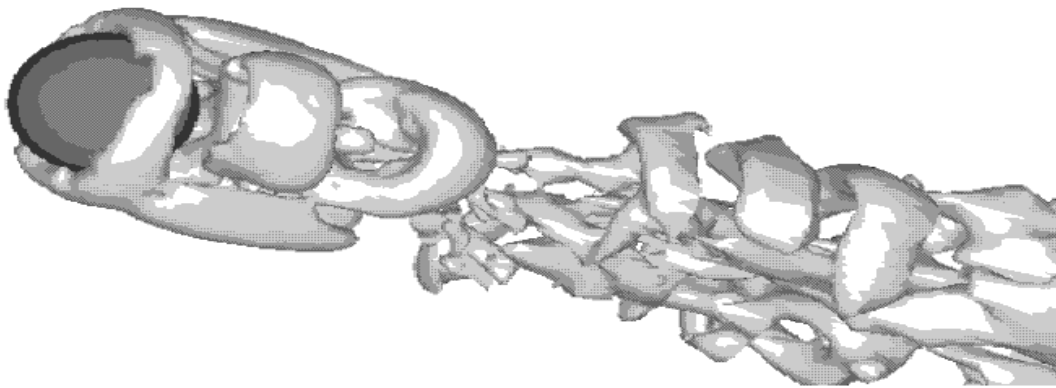


Fig. 6.2 - Detalhe dos vórtices sobre a esfera a  $Re = 1.000$  evidenciado pelo critério Q.

sendo  $F_x$  a força de arrasto, agindo sobre o corpo imerso (no caso, a esfera), obtida no presente trabalho a partir do Modelo Físico Virtual. É interessante salientar que seu valor é uma resposta direta da metodologia de fronteira imersa, não sendo necessário nenhum procedimento adicional para a avaliação da força de arrasto. Esta é uma forma totalmente diferente de se avaliar a força de arrasto (ou de sustentação) em um corpo, pois, nas metodologias tradicionais, é necessário obtê-la de forma indireta como, por exemplo, a partir da distribuição de pressão e de tensões cisalhantes na superfície do corpo.

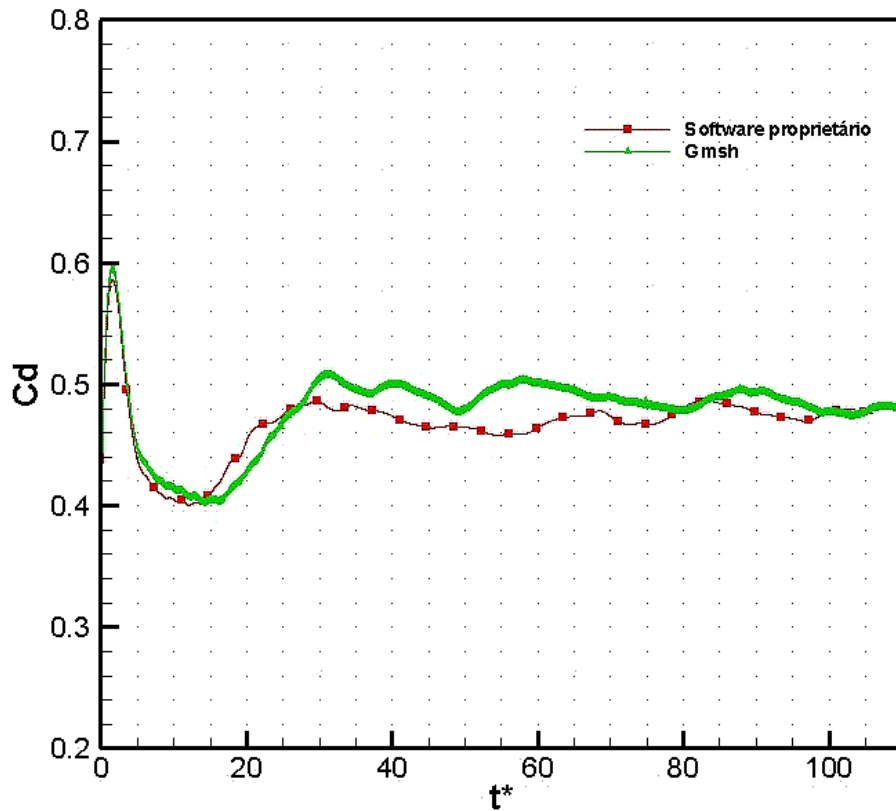


Fig. 6.3- Resultados para o coeficiente de arrasto a Re = 1000

A partir da Fig.6.3 acima, pôde-se perceber que os resultados para o coeficiente de arrasto são bastante satisfatórios para o número de Reynolds simulado, tanto para o pacote comercial quanto para o Gmsh.

Outros parâmetros igualmente importantes, obtidos de maneira semelhante ao coeficiente de arrasto, são o coeficiente de sustentação (lift coefficient – CL) e o coeficiente lateral (side coefficient – CS) avaliados, respectivamente, como Eq. 6.6 e Eq. 6.7:

$$C_L = \frac{F_z}{\left(\frac{1}{2}\right)\rho U_\infty^2 \left(\frac{\pi D^2}{4}\right)} \quad (6.6)$$

$$C_S = \frac{F_y}{\left(\frac{1}{2}\right)\rho U_\infty^2 \left(\frac{\pi D^2}{4}\right)} \quad (6.7)$$

onde  $F_y$  e  $F_z$  são a somatório das forças que atuam sobre a esfera nas direções y e z, respectivamente.

Nas Figuras 6.4 e 6.5, respectivamente, pode-se visualizar os resultados de CL e CS para Re = 1.000, indicando o comportamento aleatório da emissão de estruturas turbilhonares à jusante da esfera.

Ele permite visualizar as regiões do escoamento onde se encontram fortes tadas de deformação do fluido.

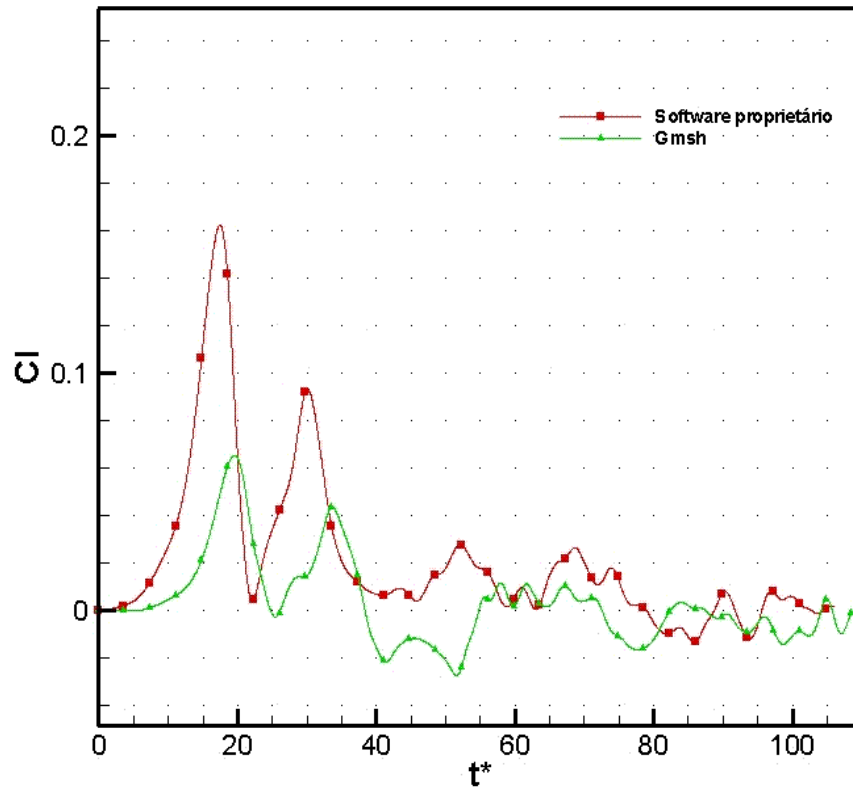


Fig. 6.4 - Coeficiente de sustentação (CL) para Re = 1000

Avaliando estatisticamente, os valores para o coeficiente de sustentação gerada pelos dois softwares geradores de malha são em média compatíveis, porém para o coeficiente lateral podemos notar maiores variações devido ao mesmo ser mais sensível ao tipo de malha gerada, uma vez que o Gmsh não consegue gerar uma malha feita com elementos completamente iguais (triângulos isósceles), havendo nesta malha algumas pequenas variações de tamanho dos triângulos. Contudo podemos verificar que nos cálculos realizados, os valores médios tendem a convergir para um valor semelhante, validando o resultado para esfera gerada pelo Gmsh.

Um outro parâmetro importante a ser calculado sobre a geometria é o L2, que é definida aqui como a raiz quadrada da norma da diferença entre a velocidade do fluido sobre a interface fluido/sólido e da velocidade desta interface (que no caso da esfera estacionária tem valor nulo). Este parâmetro tem por finalidade garantir a condição de não deslizamento sobre a interface, através do campo de força calculado. Como esperado, L2 deve tender a zero à medida que o tempo evoluir, visto que a velocidade do fluido sobre a interface tende ao valor da velocidade da interface. Na Fig. 6.6 mostra-se a evolução temporal do parâmetro L2, para o escoamento sobre a esfera onde o número de Reynolds é igual a 1000.

A pequena diferença existente entre o L2 produzido pela esfera feita com o pacote comercial, e o L2 produzido pela esfera feita no Gmsh, é devido ao fato de que esta última esfera possui aproximadamente duzentos pontos a mais que a do pacote comercial, representando assim uma área superficial um pouco maior que a outra esfera, portanto a velocidade do fluido sobre a interface fluido/sólido é maior na esfera do Gmsh do que na do software proprietário, o que justifica tal diferença.

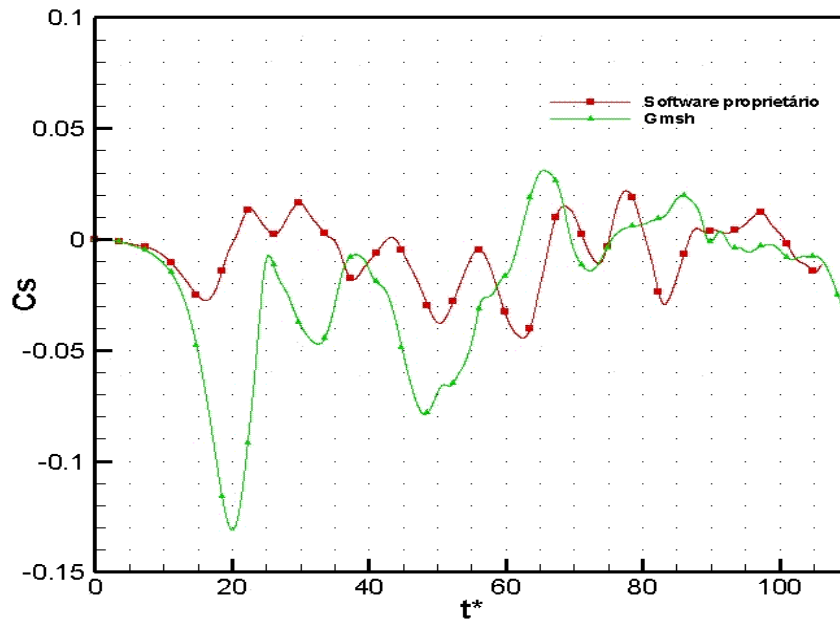


Fig. 6.5 - Coeficiente lateral ( $C_s$ ) para  $Re = 1000$

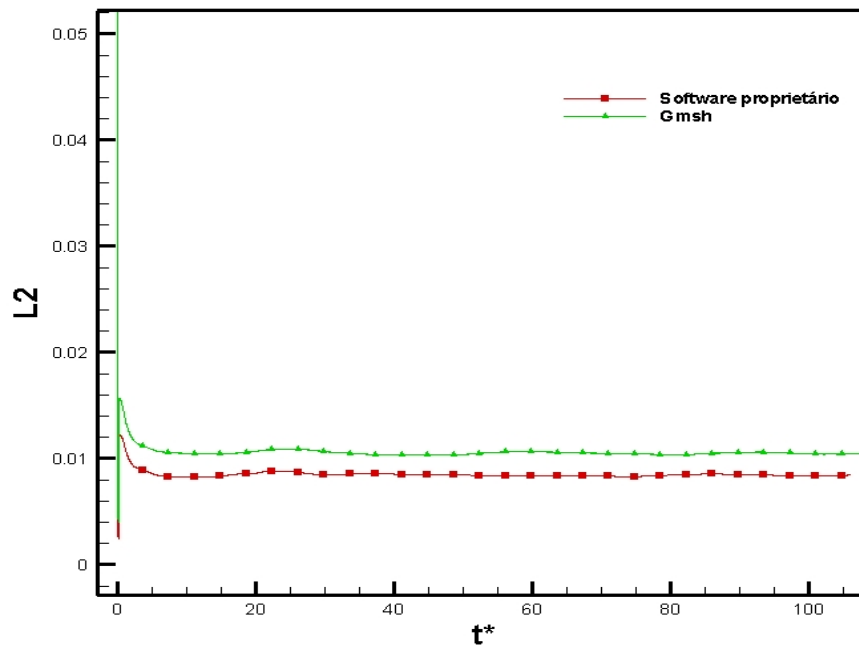


Fig. 6.6 - Evolução temporal do parâmetro  $L2$  para  $Re = 1000$

## 7. Pontos para futuras otimizações de uma interface para CFD

A atual ferramenta disponível no LTCM, além de ser desenvolvida em fortran, é uma ferramenta executável sobre plataforma Linux. Tal característica, faz com que o usuário final da ferramenta tenha que ser hábil com o sistema operacional em questão, para que possa simular qualquer caso com o código disponível.

O objetivo de se criar uma interface gráfica de usuário (GUI), tem como finalidade facilitar o uso da ferramenta para efetuar simulações, de forma que o usuário final possa interagir com o software intuitivamente.

Atualmente com o código do LTCM, levando-se em consideração a fase de pré-processamento, o usuário deve efetuar uma conexão com o cluster (local onde o código está armazenado e será executado em paralelo), feito isso, é

necessário conhecer a estrutura de diretórios do código para localizar os arquivos os quais devem ser editados. Para editar os arquivos deve-se fazer uso de uma ferramenta de edição de arquivo, através do comando “vi” do Linux, por exemplo.

Todas as condições de contorno são configuradas manualmente em um arquivo texto, sendo que algumas das condições não precisam ser modificadas, por tanto não existe a necessidade de que o usuário tenha permissão para isso, muito menos para visualizar o arquivo inteiro.

Além de visualizar um arquivo ao qual não se tem a necessidade de alterar todos os seus parâmetros, como feito com o arquivo de condições de contorno, no atual código, o usuário faz alterações no arquivo do código principal do programa. Tal tipo de operação, é de grande periculosidade, pois o usuário, caso não conheça bem o código do programa, pode fazer modificações de forma a inutilizar o código.

Algumas modificações no arquivo do código principal, têm como função, mostrar o local (pasta de arquivos) ao qual o programa irá buscar informações para poder executar seus cálculos. Uma solução viável para isso, pode ser dada ao inserir uma variável global no programa principal a ser configurada pelo usuário, de modo que o mesmo só indique os caminhos onde irá buscar e salvar informações apenas uma vez, e que essa informação seja enviada ao código principal sem que o usuário tenha contato com o código.

Outro problema a ser modificado na atual estrutura é o arquivo de configuração que faz referência aos dados iniciais a serem lidos e ao local de gravação dos resultados, além de ser um arquivo extenso, o usuário ainda tem que localizar por exemplo, o local ao qual deve acrescentar o nome da malha e o local onde irá gravar as sondas. Semelhante ao problema com o arquivo do código principal, a solução a ser tomada seria a mesma, ou seja, uma janela para indicar os arquivos em questão, porém, ao modificar tanto o arquivo de dados a serem lidos quanto o programa principal o código deve ser recompilado para que se faça valer as modificações.

Para finalizar as modificações a serem feitas a fim de se criar uma interface gráfica, existe a necessidade de criar um campo na interface para determinar quais máquinas irão executar o cálculo da simulação, e é claro um botão de comando para inicializar a execução do código, com as informações do número de máquinas e o arquivo com o nome das máquinas.

Como diferencial para o código existente, há uma necessidade de se criar um pacote na interface responsável por pós processamento, a fim de mostrar a quantificação dos resultados, de forma que a ferramenta possa se tornar o mínimo usual para fins acadêmicos.

De acordo com comparações feitas entre interfaces gráficas de softwares proprietários, optou-se por utilizar um padrão semelhante a um pacote comercial conhecido, já que o mesmo possui um formato organizacional de fácil manuseio, de forma a oferecer facilidade para que o usuário trabalhe com o mesmo, sendo assim teremos uma interface a qual se possa visualizar a malha da geometria a ser simulada, um campo para configurar as condições de contorno mostrado na Fig. 7.1 abaixo e outro para visualização de resultados como mostrado na Fig. 7.2.

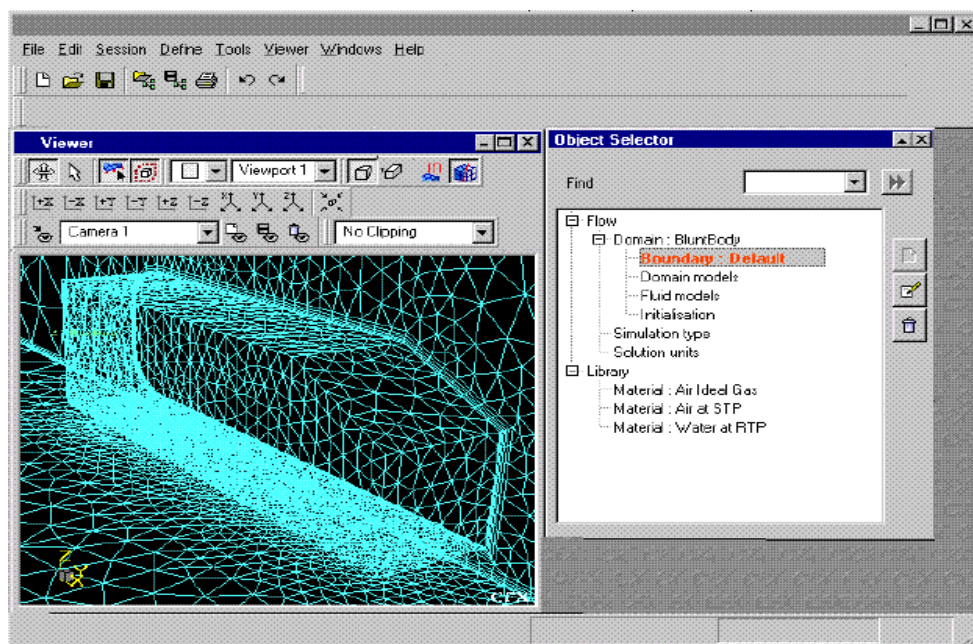


Fig. 7.1 – Interface com campo de visualização da geometria e condições de contorno



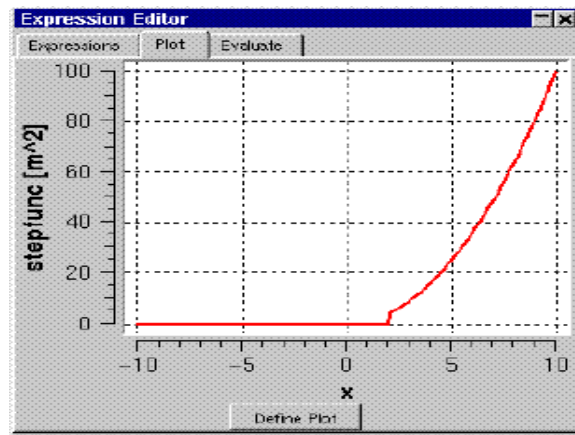


Fig. 7.2 – Exemplo de visualização de resultados

## 6.1 Referências

- Geuzaine, C., et al. "Gmsh Reference Manual: The documentation for Gmsh 1.60", 2005. Disponível em: <<http://www.geuz.org/gmsh/>> Acessado dia: 20/04/2005;
- Campregher, R. J. "Modelagem Matemática Tridimensional para Problemas de Interação Fluido-Estrutura". Tese de doutorado do Programa de Pós Graduação em Engenharia Mecânica da Universidade Federal de Uberlândia 2005.
- Jeong, J. e Hussain, F., 1995, "On the identification of a vortex", Journal of Fluid Mechanics 285, 69-94.
- Silva, A. L. F. L. "Metodologia Para Modelagem De Escoamentos Sobre Geometrias Complexas: Método da Fronteira Imersa com Modelo Físico Virtual". Tese de doutorado do Programa de Pós Graduação em Engenharia Mecânica da Universidade Federal de Uberlândia 2002.
- White, F., 1991, "Viscous Fluid Flow", McGraw-Hill, New York, USA.

## **DEVELOPMENT OF A GRAPHIC INTERFACE FOR COMPUTACIONAL FLUID DYNAMIC'S, USING THE METHODOLOGY OF BORDER IMMERSED.**

**Gustavo Prado de Oliveira**

UFU / FEMEC / LTCM  
[gpoliveira@mecanica.ufu.br](mailto:gpoliveira@mecanica.ufu.br)

**João Marcelo Vedovoto**

[vedovotojm@yahoo.com](mailto:vedovotojm@yahoo.com)  
UFU / FEMEC / LTCM

**Ana Paula Fernandes**

UFU / FEMEC / LTCM  
[apfernandes@mecanica.ufu.br](mailto:apfernandes@mecanica.ufu.br)

**Aristeu da Silveira Neto**

UFU / FEMEC / LTCM  
[aristeus@mecanica.ufu.br](mailto:aristeus@mecanica.ufu.br)

Universidade Federal de Uberlândia – Faculdade de Engenharia Mecânica – Av. João Naves de Ávila, 2160 - Campus Santa Mônica - Bloco 1M - Uberlândia/MG - CEP: 38400-902

Abstract: Now, the Laboratory of Transfer of Heat and Mass and Dynamics of the Fluids - LTMC/FEMEC/UFU possesses several works for simulations and analysis computational of the dynamics of the fluids, however with stages provided by software proprietors, turning onerous the use of such solutions. Besides, the tool now functional existent, that it makes such calculations, it was developed in the language Fortran, without a graphic interface, besides not allowing a visualization foresaw of the geometries to be worked, what hinders the handling of the same. In that way, in the present work it intends to develop a tool for the construction of geometries and the generation of their meshes of triangular finite elements superficial, being used of free softwares, for subsequent import in codes dedicated the analysis computacional of the dynamics of the fluids, through the border method immersed. Among the tools analyzed for the modelling of the geometry and generation of the mesh, she opted for the use of Gmsh, for being that a free software, functional for the platforms linux and windows, intuitive and of easy learning. Already among the tools analyzed for the development of the graphic Interface, she opted for QT, for being an understanding application for framework development in C++. This includes a library of classes and tools for the development and platform migration, besides being also functional for Windows and Linux. The work in geometries as spheres, cylinders, airfoils, being looked for to use the softwares to work with more complex forms as aircrafts and automobiles.

Keywords: border immersed, generation of meshes, free software

## AN OVERVIEW OF HIGH-RESOLUTION SCHEMES FOR AEROACOUSTICS

### Odenir de Almeida

EMBRAER – Empresa Brasileira de Aeronáutica S.A  
Av. Brigadeiro Faria Lima, 2170 – São José dos Campos - SP  
odenir.almeida@embraer.com.br

### Francisco José de Souza

EMBRAER – Empresa Brasileira de Aeronáutica S.A  
Av. Brigadeiro Faria Lima, 2170 – São José dos Campos - SP  
francisco.souza@embraer.com.br

**Abstract.** *The present work can be conceived as a first step in the direction of constructing an aeroacoustic solver. Such development in Computational Aeroacoustics (CAA) is not achieved before a complete understanding of the physics involved in acoustic problems. Thus a brief technical background in CAA is presented in order to introduce the numerical activities performed herein. The one-dimensional wave equation has been discretized with optimized high order finite difference schemes taking into consideration usual conditions of consistency, stability and convergence. These schemes are known as DRP (Dispersion-Relation-Preserving). To advance the solution in time, high order Runge-Kutta schemes are applied. In order to validate the numerical scheme, initial disturbance functions are used, which are propagated in a 1D mesh. The advantages and limitations for the numerical schemes are deeply discussed shedding light to the relevant parameters which are necessary to achieve a good aeroacoustic simulation. The numerical results generated in this work are compared with the exact analytical solutions showing a good agreement.*

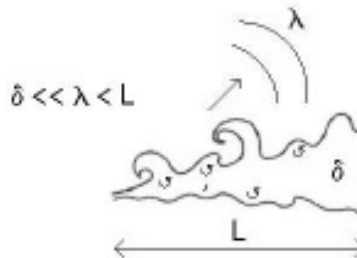
**Keywords.** *Aeroacoustics, wave-equation, finite difference, numerical schemes, aerodynamic noise*

### 1. Introduction

Computational aeroacoustics (CAA) has made impressive progress along the last decades. Most of this advance has been driven by car aerodynamics and essentially aircraft designs for low noise. From the dawn of Sir James Lighthill work in 1952 up to recent days, aeroacoustic has evolved in terms of acquired knowledge about physical phenomena and development of tools for academic and industrial applications. Indeed, it cannot be neglected that many classes of acoustic problems were completely understood and solved by analytical and numerical approaches so far. Also, it is important to emphasize that these acoustic problems are essentially canonical cases involving wave propagation in very simple domains. With the need for industrial application of CAA in the last decade, numerical efforts have been specially concentrated in the creation of tools aimed to predict the generation and propagation of aerodynamic sound.

The following discussion has been addressed by some authors in the literature and consists of an important key about separating classical computational fluid dynamics (CFD) of the recent CAA. In essence, it is important to keep in mind that classical CFD simulations cannot handle acoustic phenomena.

Differently from a classic computational fluid dynamics (CFD) analysis, there are other important physical and numerical issues related to computational aeroacoustics. From the physical standpoint, it is largely known that sound is most generated by transitional and turbulent flow phenomena like separation and shearing, vortex shedding, impingement and shocks. In all these cases there are a wide range of length and time scales associated. In the flow field these length and time scales are quite different from those ones existing in the acoustic field. Figure 1 illustrates the length and time scales difference between fluid mechanic and acoustic. Kolmogorov's scales ( $\delta$ ) of turbulence are much smaller than the wavelength ( $\lambda$ ) of the generated sound waves which by its turn is smaller than the characteristic fluid dynamic length or the largest eddy in the flow field ( $L$ ).



**Figure 1.** Disparity between the flow field and acoustic scales.

It's important to remember that for aerodynamics or fluid mechanic problems, flow disturbances, generally, tend to decay very fast away from a body or their source of generation. That is, any flow disturbance suffers the effects of dispersion and dissipation along the flow field. On the other hand, acoustic waves are non-dispersive and non-dissipative and consequently decay very slowly in the acoustic field, traveling for long distances until reaching any observer. From the numerical standpoint, such physical implications listed above drive numerical requirements like dealing with different length scales in different parts of the computational domain. Yet, numerical schemes used to evaluate time and space-derivatives should be of low dispersion and dissipation error in order to guarantee long wave propagations. Another limiter is the computation domain that is inevitably finite in size. Thus, acoustic waves being propagated decay very slowly and actually reach the boundaries of the computation domain. To avoid the reflection of outgoing sound waves back into the computation domain and contamination of the solution, radiation and outflow boundary conditions must be imposed at the artificial exterior boundaries to assist the waves to exit smoothly. And finally, still considering a finite domain, the wave propagation needs to be performed outside of the computational domain to reach an observed that is often far away from the sound source generation. The discretization of such big domain is completely out of question and so, numerical propagation methodologies should deal with this problem.

According to Tam (2005), the need for highly accurate numerical methods was recognized from the earliest stages in the development of Computational Aeroacoustics (CAA) since the classical computational fluid dynamics could not deal with long-time integration with minimal dissipation and dispersion in order to guarantee the propagation of sound waves in the far-field.

As a summary presented by Tam (2004) to simulate an aeroacoustic phenomenon or problem computationally, the numerical algorithm must consist of several basic elements which are listed below:

- (1) A time marching computation scheme;
- (2) A suitably designed computation grid;
- (3) An artificial selective damping algorithm or filtering procedure if the marching scheme has no built-in artificial selective damping;
- (4) A set of radiation/outflow numerical treatments for use at the boundaries of the computation domain.

Along the years, different methodologies have been created and applied to build aeroacoustic numerical codes. It started with the use of higher order schemes by believing that they were more suitable than lower-order schemes in terms of stability. For acoustic calculations, however, the stability consideration alone is not sufficient, since Runge-Kutta schemes retail both dissipation and dispersion errors. The numerical solutions need to be also time accurate to resolve the wave propagations. Time integration has then been optimized with the same aim in view, especially using Runge-Kutta (RK) algorithms – Berland (2005).

By reviewing the literature and seeking for low-dissipative, low-dispersive and large spectral bandwidth numerical algorithms it can be found relevant works in aeroacoustics. For instance, Tam & Webb (1993b) exploited the DRP (Dispersion-Relation-Preserving) schemes, Lele (1992) developed compact finite differences, and recently Bogey & Bailly (2004) built up explicit finite differences and selective filters accurate for waves down to four points per wavelength.

Other numerical works as of Roeck *et al* (2004), Tam (2004), have presented high-order finite difference schemes for computational aeroacoustics. However, when dealing with space and time derivatives discretization, special attention should be given to DRP schemes which in essence possess the same dispersion relations as the original partial differential equations. As presented by Tam & Webb (1993a) the central idea of this approach is that a time marching high order finite difference scheme can faithfully reproduce the three types of waves, namely the acoustic, vorticity, and entropy waves, of the linearized Euler equations. Nevertheless, this is only true if the dispersion relations of the finite difference scheme are the same as those of the partial differential equations. In terms of numerical results, the DRP schemes tend to be more suitable for wave solutions. The complete description of these schemes can be found in the work of Tam & Webb (1993b). However, it is important to emphasize that the main focus of the work presented herein is given to the exploration of DRP (Dispersion-Relation-Preserving) spatial schemes.

In general, the present work can be divided in two objectives. The first one is to provide a general overview about computational aeroacoustic requirements through the discussion of some relevant associated issues and by presenting recommended references to those ones that are initiating researches in this area. The second objective is to present the validation of time and spatial discretization of the wave equation by using different schemes and focusing on the use of DRP schemes. To accomplish this task, numerical simulations for the propagation of a pulse or disturbance are conducted. The solution is compared with analytical results showing a good agreement. Finally, the advantages and limitations for the numerical schemes are deeply discussed shedding light to the relevant parameters which are necessary to achieve a good aeroacoustic simulation.

## 2. Numerical Schemes

This section is aimed to provide a general guidance about numerical schemes that can be used for aeroacoustic simulations. The focus will be given to DRP schemes for spatial discretization and Runge-Kutta time integration methods which will be summarized and presented hereafter.

Standard central differences schemes are presented in Roeck *et al* (2004), in which a spatial derivative of a function in a grid point is approximated by a weighted summation of the function values in the neighboring grid points. For a 7-point (ST7) and 9-point (ST9) stencil, a 6<sup>th</sup> and an 8<sup>th</sup> order accurate scheme, respectively, is obtained.

With the same approach of improving the dissipation and dispersion characteristics, other works were proposed (e.g. Bogey *et al.* (2002), Hu *et al.* (1996) and Lele (1992)). The importance of dispersion relations of the finite difference schemes have been emphasized in these works with good results in terms of wave propagation and robustness of the computation.

In this context, the low-dispersion and low-dissipation numerical schemes also known as DRP (Dispersion-relation-preserving) presented by Tam & Webb (1993b) seems to be one of the most suitable for aeroacoustic problems. Tam (1993b, 2004) has discussed the numerical issues related to these schemes.

Regarding temporal discretization the most common methods are the Runge-Kutta schemes. These methods have been evolved following the same idea of DRP schemes, where dispersion and dissipation of the scheme are minimized in the Fourier space over a large range of frequencies for linear operators, as presented by Hu *et al.* (1996) and Bogey and Bailly (2003).

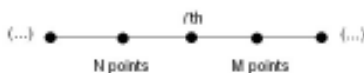
In the following section, a brief description of these schemes is presented and discussed. The numerical implementation was concentrated in the DRP 7-point, DRP 15-point schemes of Tam & Webb (1993b) and Tam (2004), respectively, and the 9-point optimized central difference scheme of Bogey *et al.* (2002). For time integration two Runge-Kutta schemes were selected based on independent criteria of low-storage (Williamson (1980)) and low-dispersion/dissipation (Hu *et al* (1996)).

In summary, the understanding and numerical implementation of these schemes could be considered an initial step in order to build an aeroacoustic solver which is the main goal for work future.

### 2.1. Spatial Discretization

The description of optimized spatial discretization for the Dispersion-Relation-Preserving (DRP) schemes is given in the work of Tam & Webb (1993b) as follows:

Consider the approximation of the first derivative  $\partial f / \partial x$  - Equation (1) at the  $l$ th node of a uniform grid. Suppose  $M$  values of  $f$  to the right and  $N$  values of  $f$  to the left of this point are used to form the finite difference approximation – Figure 2.



**Figure 2.** Uniform grid representation.

$$\left( \frac{\partial f}{\partial x} \right)_i \approx \frac{1}{\Delta x} \sum_{j=-N}^M a_j f_{l+j} \quad (1)$$

The usual procedure is to expand the right-hand side of (1) in Taylor series of  $\Delta x$  and then determine the coefficients  $a_j$  by equating coefficients of the same powers of  $\Delta x$ . Finite difference schemes constructed in this way will be referred to as the standard schemes. In the DRP schemes the coefficients  $a_j$  are found in a different way. It is proposed that they be determined by requiring the Fourier transform of the finite difference scheme on the right of (1) to be a close approximation of that of the partial derivative on the left.

The finite difference equation (1) is a special case of the following equation in which  $x$  is a continuous variable:

$$\frac{\partial f}{\partial x}(x) \approx \frac{1}{\Delta x} \sum_{j=-N}^M a_j f(x + j\Delta x) \quad (2)$$

(1) can be recovered from (2) by setting  $x = l\Delta x$ . The Fourier transform and its inverse of a function are related by

$$\tilde{f}(\alpha) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(x) e^{-i\alpha x} dx \quad (3)$$

$$f(x) = \int_{-\infty}^{\infty} \tilde{f}(\alpha) e^{i\alpha x} d\alpha \quad (4)$$

The Fourier transform of the left and right sides of (2) are:

$$i\alpha \tilde{f} \approx \left( \frac{1}{\Delta x} \sum_{j=-N}^M a_j e^{i\alpha_j \Delta x} \right) \tilde{f} \quad (5)$$

by comparing the two sides of (5) it is clear that the quantity

$$\bar{\alpha} = \frac{-i}{\Delta x} \sum_{j=-N}^M a_j e^{i\alpha_j \Delta x} \quad (6)$$

is effectively the wave number of the Fourier transform of the finite difference scheme (2) or (1);  $\bar{\alpha}\Delta x$  is a periodic function of  $\alpha\Delta x$  with period  $2\pi$ . To assure that the Fourier transform of the finite difference scheme is a good approximation of that of the partial derivative over the range of wave numbers of interest (waves with wave length longer than four  $\Delta x$  or  $|\alpha\Delta x| < \pi/2$  it is required that  $a_j$  be chosen to minimize the integrated error  $E$  defined:

$$E = \int_{-\pi/2}^{\pi/2} |\alpha\Delta x - \bar{\alpha}\Delta x|^2 d(\alpha\Delta x) \quad (7)$$

$$= \int_{-\pi/2}^{\pi/2} \left| ik - \sum_{j=-N}^M a_j e^{ijk} \right|^2 dk \quad (8)$$

The conditions that  $E$  is a minimum are

$$\frac{\partial E}{\partial a_j} = 0, j=-N \text{ to } M; \quad (9)$$

(9) provides a system of linear algebraic equations by which the coefficients  $a_j$  can be easily determined.

More details about DRP schemes can be found in Tam & Webb (1993b).

The coefficients of the 7-point and 15-point DRP schemes used in this work are:

**7-point scheme:**

$$\begin{aligned} a_0 &= 0 \\ a_1 = -a_1 &= 0.77088238051822552 \\ a_2 = -a_2 &= -0.166705904414580469 \\ a_3 = -a_3 &= 0.02084314277031176 \end{aligned}$$

**15-point scheme:**

$$\begin{aligned} a_0 &= 0 \\ a_1 = -a_1 &= 0.91942501110343045059277722885 \\ a_2 = -a_2 &= -0.35582959926835268755667642401 \\ a_3 = -a_3 &= 0.15251501608406492469104928679 \\ a_4 = -a_4 &= -0.059463040829715772666828596899 \\ a_5 = -a_5 &= 0.019010752709508298659849167988 \\ a_6 = -a_6 &= -0.0043808649297336481851137000907 \\ a_7 = -a_7 &= 0.00053896121868623384659692955878 \end{aligned}$$

### Artificial Selective Damping:

To obtain a high-quality numerical solution, it is necessary to eliminate the short wavelength spurious numerical waves. The origin of these spurious waves is related to grid-to-grid oscillations and a way to eliminate or reduce such instabilities is introducing artificial selective damping terms in the finite difference equations. Tam & Webb (1993b) used tailored damping terms specifically for eliminating only the short waves without affecting the physical long waves.

Additional information about this approach can also be found in Roeck *et al* (2004) and Bogey and Bailly (2004). The present work considers both the effect of the artificial selective damping as described in Tam & Webb (1993b) and no use of selective damping. The artificial selective damping has been applied only for the DRP scheme. The coefficients for the damping function are given as follows:

$$c_0 = 0.3276986608$$

$$c_1 = -c_1 = -0.235718815$$

$$c_2 = -c_2 = 0.0861506696$$

$$c_3 = -c_3 = -0.0142811847$$

### 2.2. Time Integration

Runge-Kutta algorithms are largely used in fluid dynamics applications because stability over a wide range of frequencies. Many of these algorithms were optimized for low-storage requirements in order to make easier the numerical computation on systems that are very large. A good reference in this area is the work of Williamson (1980) where low-storage schemes up to fourth order were evaluated without losing stability and accuracy.

According to Hu et al. (1996) for computing acoustic waves, only the stability consideration is not sufficient, since the Runge-Kutta schemes also entail both dissipation and dispersion errors. Many works go to the direction of optimizing Runge-Kutta schemes for low dispersion and dissipation properties. This is the case of works from Hu et al. (1996) and Bogey and Bailly (2004).

In order to present a summary of the approach done herein, let the time evolution equation be written as follows:

$$\frac{\partial U}{\partial t} = F(U) \quad (10)$$

in which  $U$  represents the vector containing the solution values at the spatial mesh points and the operator  $F$  contains the discretization of the spatial derivatives. For simplicity, it's assumed that  $F$  does not depend on  $t$  explicitly. An explicit low storage  $p$ -stage Runge-Kutta scheme advances the solution from time level  $t_n$  to  $t_n + \Delta t$  as:

$$\begin{aligned} u^0 &= u^n \\ u^l &= u^n + \alpha_l \Delta t F(u^{l-1}) \quad \text{for } l = 1, \dots, p \\ u^{n+1} &= u^p \end{aligned} \quad (11)$$

The standard  $p$ -stage Runge-Kutta schemes of  $p$ th-order (for a linear operator  $F$ ) can be obtained with Taylor Series expansion of  $u(t_n + \Delta t)$ . The coefficients  $\alpha_j$  should be chosen to obey several conditions as maximum order of accuracy. Like the spatial integration, presented in section 2.1, the coefficients  $\alpha_j$  can also be chosen to minimize the dispersion and dissipation errors.

More details about the determination of these coefficients can be found in Hu et al. (1996), Roeck (2004) and Berland *et al.* (2005). In this paper, the low-storage and low-dispersion/dissipation properties are evaluated through the implementation of the following Runge-Kutta schemes:

- 3<sup>rd</sup> order low-storage Runge-Kutta scheme – Williamson (1980) – RK3W
- 5 stages low-dissipation and low-dispersion 2<sup>nd</sup> order Runge-Kutta scheme – Hu et al (1996) – LLDRK5
- 5 stages 2<sup>nd</sup> order optimized Runge-Kutta scheme – Bogey and Bailly (2002) – RKO5

The coefficients  $\alpha_j$  for the schemes of Hu et al (1996) and Bogey and Bailly (2002) are written in Table 1.

Table 1 -  $\alpha_j$  for RK5 and RK6

	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\alpha_4$	$\alpha_5$	$\alpha_6$
LLDRK5	0.197707993	0.237179241	0.333116	0.5	1	-
RK05	0.181575486	0.238260222	0.330500707	0.5	1	-

### 3. Validation problem

To validate the scheme properties, two initial value problem of a one-dimensional convective wave equation are considered.

The mathematical problem consists in finding the solution for the wave equation:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 \tag{12}$$

with an initial disturbance  $u = f(x)$  at time  $t = 0$ . The solution consists of the initial disturbance propagating to the right in an one-dimensional mesh. The propagation of this disturbance posses a dimensionless speed equal to 1 ( $u = f(x-t)$ ).

The initial disturbance functions  $f(x)$  used in this work are described below:

**(1) Box-car function:**

$$t = 0, \quad f(x) = 0.5[H(x+50) - H(x-50)] \tag{13}$$

The factor 0.5 defines the amplitude of the step function and 50 is the haft width of the box-car. The fourier transformation of this function contains high frequency components and these must be damped in order to avoid spurious high-frequency waves. For this reason, this function is a good way to evaluate the effect of artificial damping.

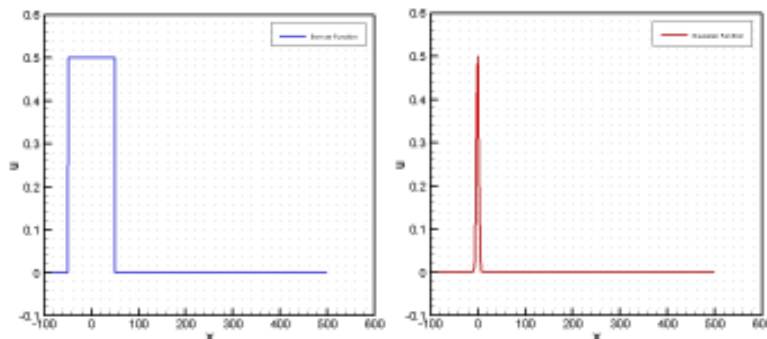
**(2) Gaussian function:**

$$t = 0, \quad f(x) = 0.5 \exp\left[-\ln 2 \left(\frac{x}{3}\right)^2\right] \tag{14}$$

According to Roeck (2004) the Gaussian function is used to evaluate the general dispersion and dissipation properties and the stability ranges of the scheme. The Fourier transformation of a Gaussian pulse is also a Gaussian pulse with a maximum at  $k\Delta x = 0$ . The initial disturbance will therefore be mainly situated in the long wave range and artificial damping is not necessary and will not be added to the numerical scheme.

The variables of equation (12) are made dimensionless with  $\Delta x$  as length scale and  $\Delta x/c_0$  as the time scale. The computational domains have a dimensionless length of  $700 \Delta x$  for the Gaussian and Box-car functions.

Figure 3 presents the analytical solution for the functions (1) and (2) at the initial time ( $t = 0$ ).



**Figure 3.** Analytical solution for the functions (1) and (2) at the initial time ( $t = 0$ ).



All the calculations were done with a grid spacing  $\Delta x$  equal to unity and a time step  $\Delta t$  equal to 0.1. The stability requirements were satisfied with a CFL ( $\Delta t / \Delta x$ ) of 0.1.

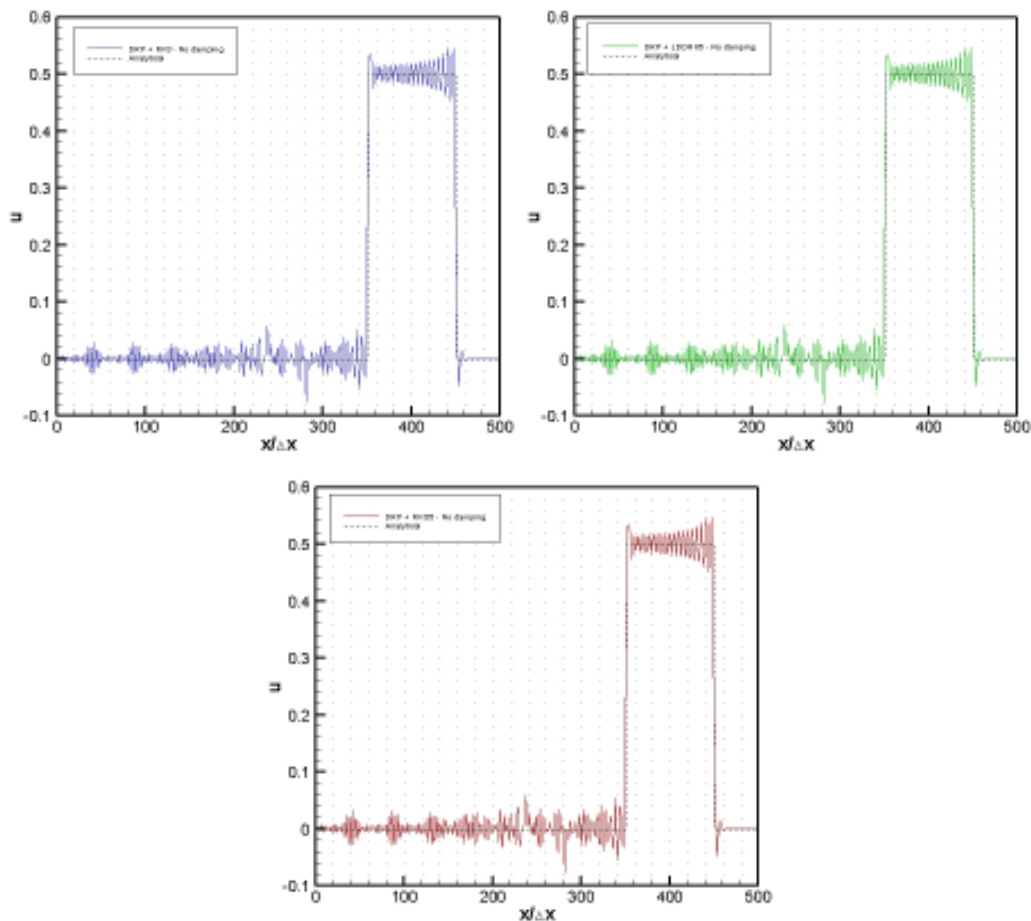
In this work it's not applied any kind of boundary conditions. The simulations were evaluated for a dimensionless time of 400. At this time, the initial disturbance reaches the dimensionless position of 400. Taking this into account, no specific boundary conditions have been added since spurious reflections of the initial disturbance at the boundaries are negligible if the disturbance has not reached the boundary yet.

## 4. Results

### 4.1 Initial Disturbance Function - Box-car Function

Figure 4 shows the computed results of the seven-point DRP scheme with different Runge-Kutta time integration methods. For these results damping was not used. For comparison purposes, the analytical solution is plotted against the numerical results.

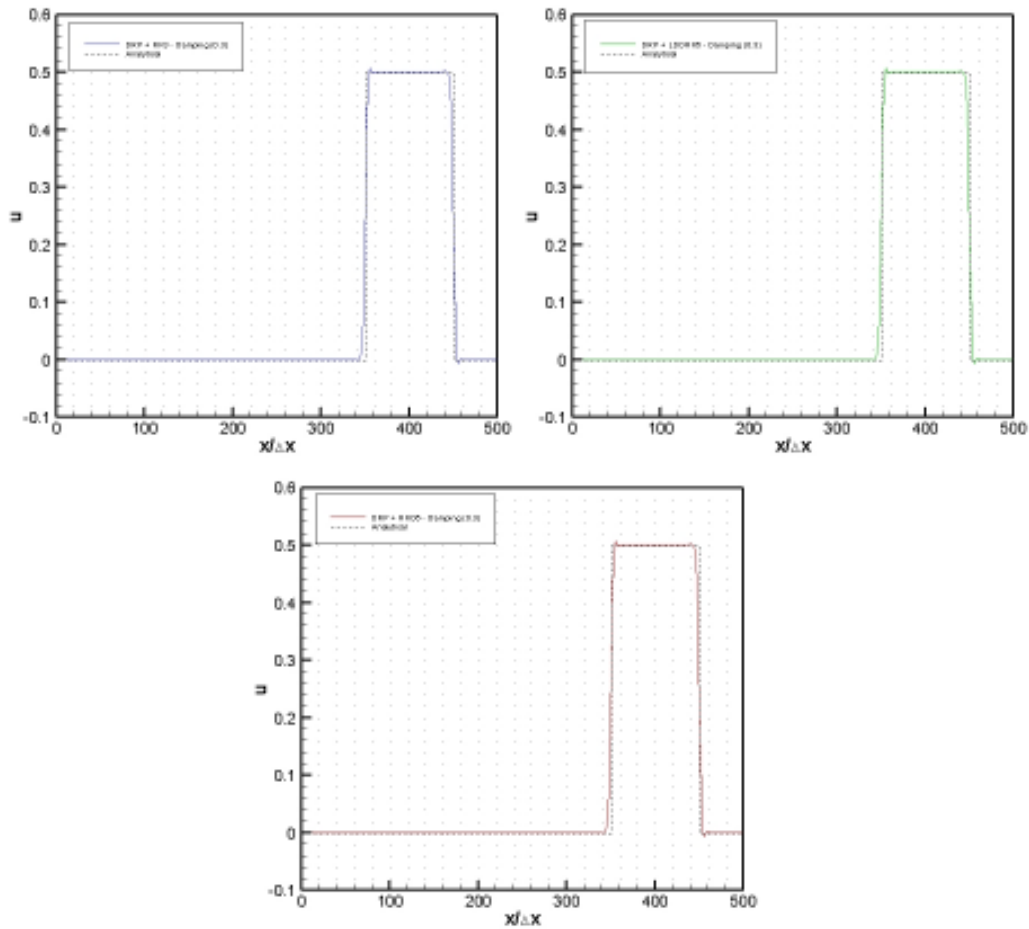
The presence of spurious waves is evident in the computed solution. These spurious waves are generated by the discontinuities of the initial condition and due to grid-to-grid oscillations. The Runge-Kutta schemes were quite stable for all simulations and it's possible to say that all the numerical error was introduced by the spatial discretization.



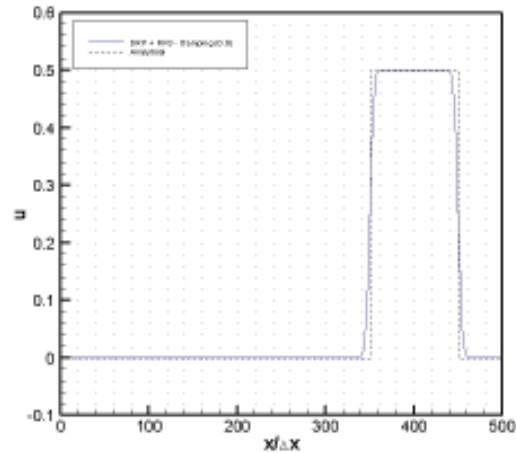
**Figure 4.** Comparison between the computed and the analytical solutions of the one-dimensional wave equation for the Box-car initial disturbance

The effect of the artificial selective damping terms in the wave equation can be clearly seen in Figure 5. Almost all the spurious oscillations were damped out. To obtain these results a damping constant of 0.3 was used. This value was selected based on the work of Roeck (2004). A larger value of this constant adds more damping into the scheme making the spurious waves smaller.

Even with the artificial selective damping it can be seen some overshoot in the left corner of the discontinuity. This overshoot is completely removed by increasing the value of the damping constant to 0.9 – Figure 6. However, the corners of the step have been smeared.



**Figure 5.** Effect of the damping function in the propagation of the Box-car function (damping constant of 0.3)



**Figure 6.** Effect of the damping function in the propagation of the Box-car function (damping constant of 0.9)

Contrary to the work of Roeck (2004), the increase of the damping constant over 0.7 did not bring any instability to the numerical scheme and the solution shown in Figure 6 could be obtained.

Figure 7 and 8 shows the numerical results for the propagation of the box-car function with the 15-point DRP scheme and the 9-point optimized central difference scheme of Bogey et al. (2002) associated with different Runge-Kutta schemes. For both schemes no artificial selective damping was applied. Even without damping, it's possible to see that the spatial discretization with more points perform better than the DRP 7-point scheme for the waves behind the pulse. However, at the discontinuity there is more overshoot.

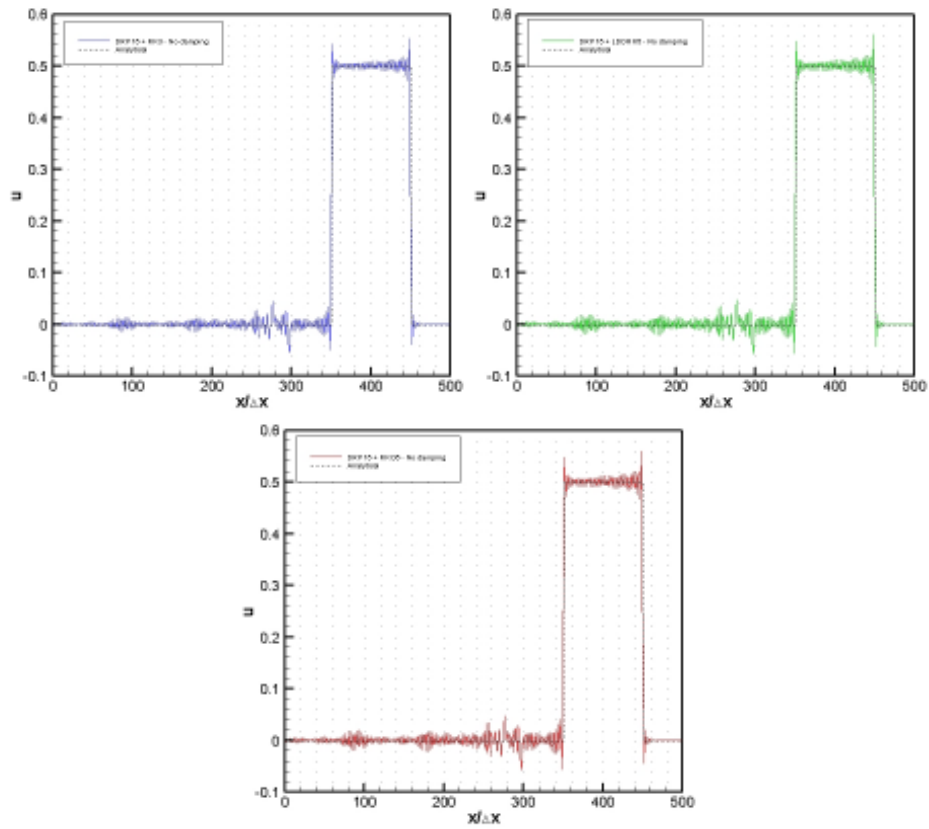


Figure 7. Propagation of Box-car function with 15-point DRP scheme – no artificial selective damping.

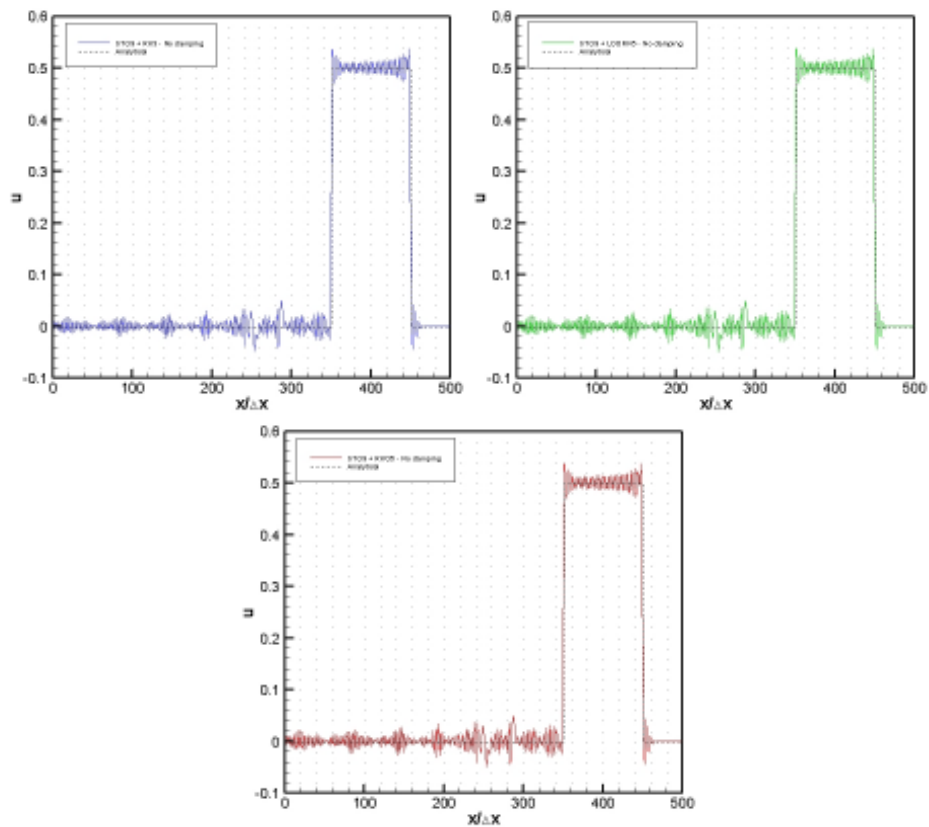
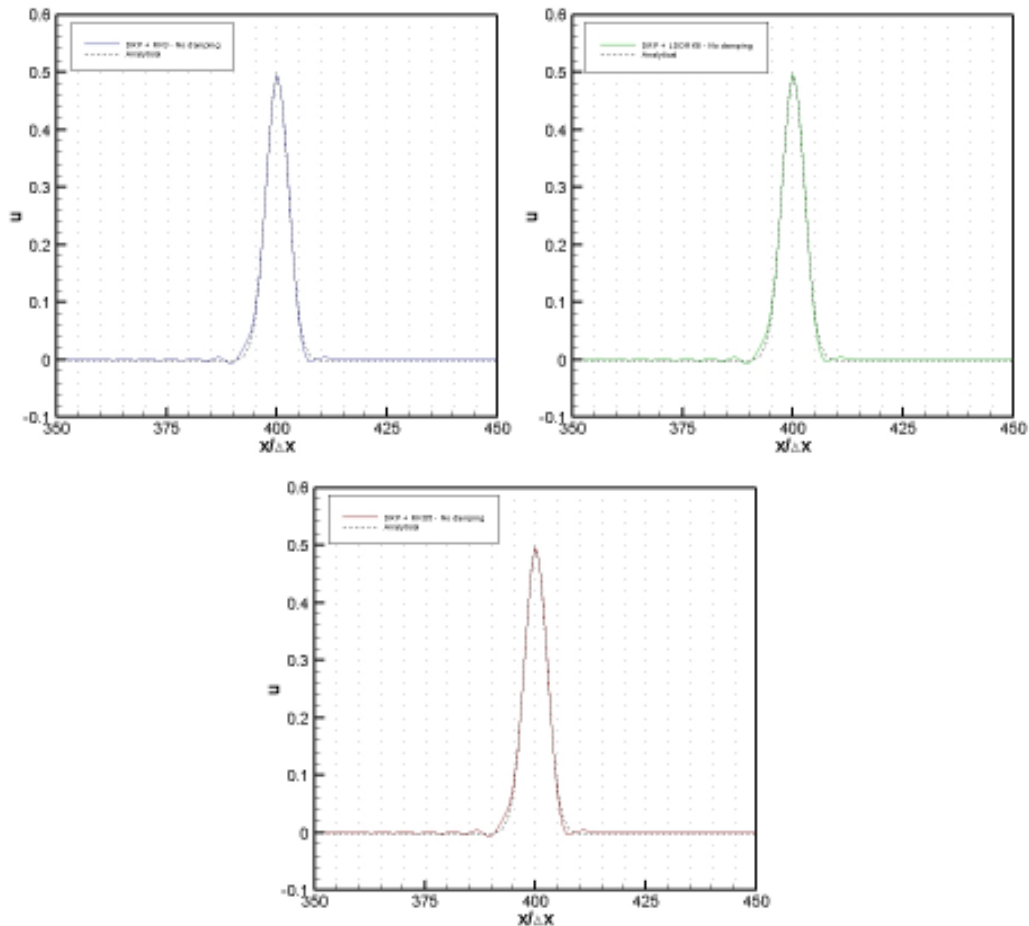


Figure 8. Propagation of Box-car function with 9-point optimized central-difference scheme – no artificial selective damping.

#### 4.2 Initial Disturbance Function - Gaussian

Figure 9 shows the computed results of the seven-point DRP scheme with different Runge-Kutta time integration methods for the propagation of the Gaussian disturbance. In spite of the good agreement with the analytical solution some dispersed waves can be seen behind the pulse. These waves are originated by high-frequency components that are propagating at a lower wavespeed. It was expected that the low dispersion of the DRP scheme could diminish the amplitude of these trailing waves. However they are still present for the 7-point DRP scheme. It is important to emphasize again that the time integration schemes are almost not contributing to the numerical error since the stability range was satisfied with the Courant number. The main error is then introduced by the spatial difference scheme.



**Figure 9.** Propagation of Gaussian function with 7-point DRP scheme – no artificial selective damping.

The same calculations were done by increasing the stencil of the spatial discretization for 15 points using DRP schemes and for 9 points using an optimized central difference scheme - Figure 10 and 11 respectively. The increase in the number of points for the spatial discretization and the better dispersion properties of these schemes led to very good agreement for the pulse propagation when compared to the analytical solution. Practically, all the trailing waves were removed and the pulse was propagated correctly with the constant wave speed.

Both 15-point (DRP) and 9-point (STO9) schemes had a better performance when compared to the 7-point (DRP) scheme. However, these schemes are computationally more expensive due to a stencil with more points.

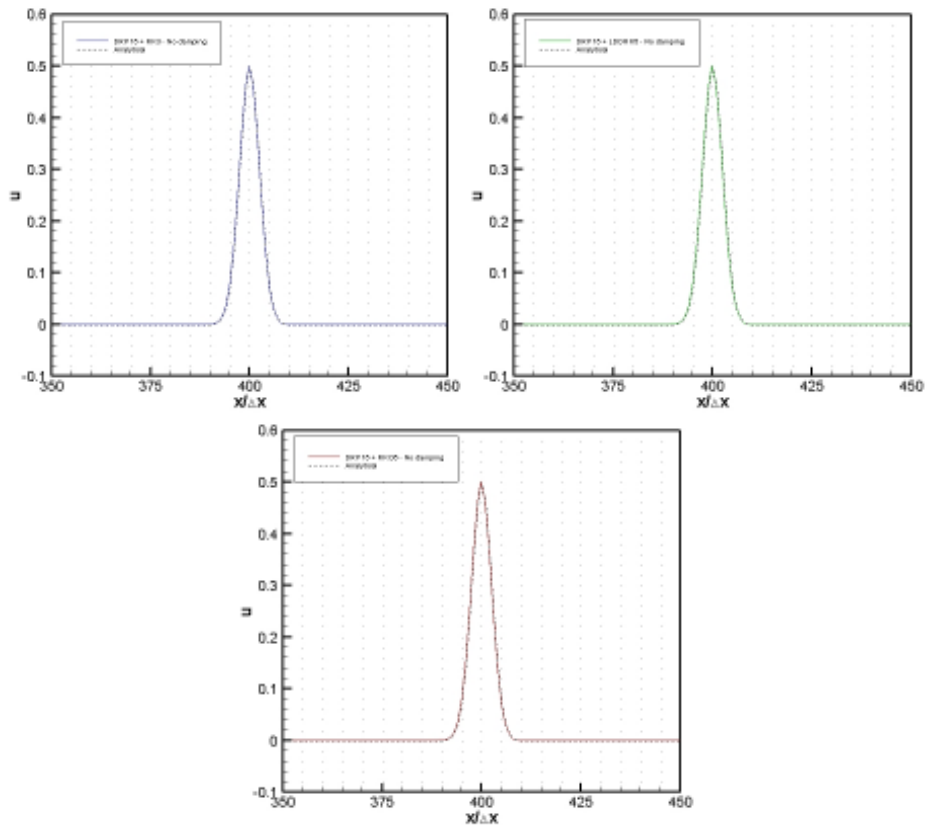


Figure 10. Propagation of Gaussian function with 15-point DRP scheme – no artificial selective damping.

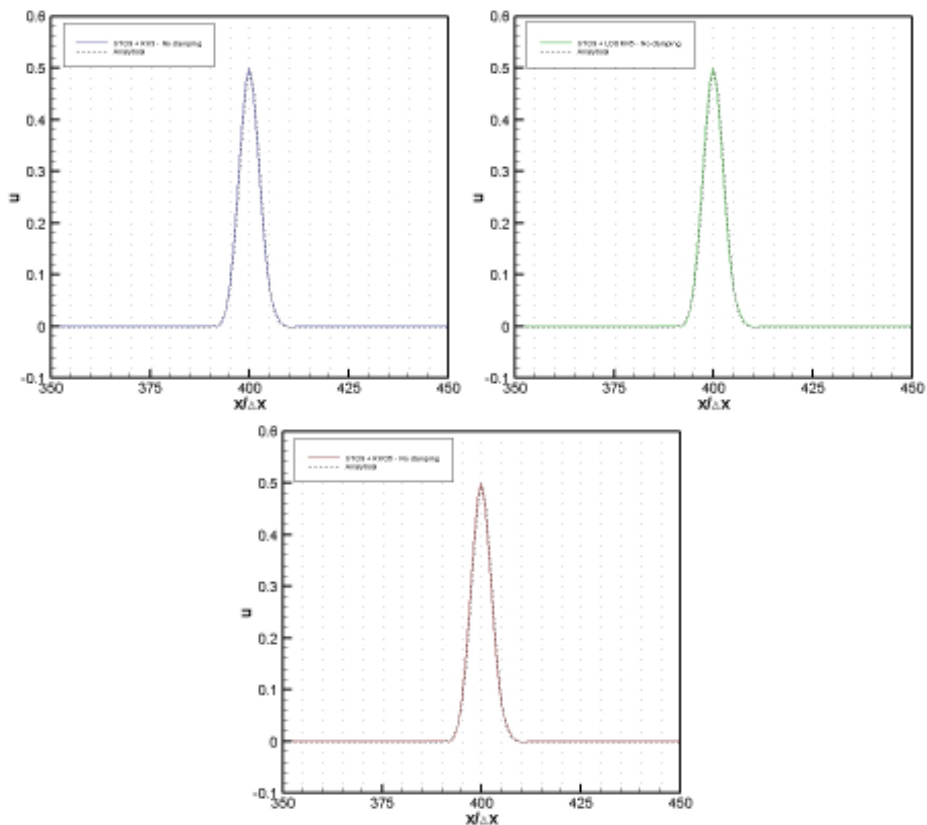


Figure 11. Propagation of Gaussian function with 9-point optimized central-difference scheme – no artificial selective damping.

## **5. Conclusions**

The present work provided an overview of different spatial and temporal schemes that can be used for aeroacoustics calculation. Since, spatial and temporal discretization schemes with low-dispersion and low-dissipation properties have shown good performance in wave propagation problems it was investigated the DRP schemes and Runge-Kutta time integration methods which such properties. The DRP schemes with 7-point, 15-point and an optimized 9-point central difference scheme were evaluated for spatial discretization in the solution of a one-dimensional wave equation considering different initial disturbance propagation. To advance the solution in time, three different Runge-Kutta schemes were evaluated.

In general, the DRP schemes showed a good performance for wave propagation mainly due to the low dissipation and low dispersion properties. The 15-point DRP scheme was much more efficient than the 7-point, however the computational time increased. Also, the optimized 9-point central difference scheme presented good performance and it was better than the 7-point DRP scheme. The use of artificial selective damping was important to propagate the Box-car function correctly and to eliminate the spurious waves present in the solution. By increasing the damping constant, it was possible to reach a reasonable solution, but still with some smearing of the discontinuity.

Regarding the Runge-Kutta schemes, it's possible to say that over the range of stability, all the three schemes performed very well no bringing dispersion and dissipation to the solution. However, it's important to keep in mind that the schemes LDDRK5 and RK05 are computationally more time consuming since at each stage evaluations of the spatial derivative need to be performed.

As a final recommendation of this work, the combination of the 9-point central difference (STO9) and the 15-point (DRP) scheme with the Runge-Kutta RK05 is a good approach to initiate aeroacoustic computations.

## **6. Acknowledgement**

The authors would like to thank EMBRAER (Empresa Brasileira de Aeronáutica – S.A) for the possibility of participation in this event.

## **7. References**

- Berland, J., Bogey, C., Bailly, C., “Low-dissipation and low-dispersion fourth-order Runge-Kutta algorithm”, *Computers & Fluids*, 2005.
- Bogey C., Bailly, C., “A family of low dispersive and low dissipative explicit schemes for computing the aerodynamic noise”, *AIAA-paper 2509*, 2002.
- Bogey C., Bailly, C., “A family of low-dispersive and low-dissipative explicit schemes for flow noise and noise computations”, *Journal of Computational Physics*, 2004, 194, pp. 194-214.
- Hu, F. Q., Hussaini, M. Y., Manthey, J. L., “Low-dissipation and low-dispersion Runge-Kutta schemes for computational aeroacoustics”, *Journal of Computational Physics*, 124, 1996, pp. 177-191.
- Lele, S. K., “Compact finite difference schemes with spectral-like resolution”, *Journal of Computational Physics*, 1992, pp-16-42.
- Lighthill M. J., “On sound generated aerodynamically”, I, *General Theory*, *Proceedings of Royal Society, London Ser.*, A221:564-87.
- Roeck, W. De., Desmet, W., Baelmans, M., Sas, P., 2004, “An overview of high-order finite difference schemes for computational aeroacoustics”, *Proceedings of ISMA*, pp. 353-368.
- Tam, C. K., “Computational aeroacoustics: An overview of computational challenges and applications”, *International Journal of Computational Fluid Dynamics*, Vol 18 (6), Aug 2004, pp. 547-567.
- Tam, C. K., “Computational aeroacoustics: Issues and Methods”, *AIAA Journal*, Vol 33 (10), October 2005, pp. 1788-1796.
- Tam, C. K., Webb, J. C., “Dispersion-relation-preserving finite difference schemes for computational acoustics”, *Journal of Computational Physics*, 107, 1993b, pp. 262-281.
- Tam, C. K., Webb, W., Dong, J. C., “A study of short wave components in computational acoustics”, *Journal of Computational Acoustics*, Vol 1, March 1993a, pp. 1-30.
- Williamson, J. H., “Low-storage Runge-Kutta Schemes”, *Journal of Computational Physics*, 35, 1980, pp. 48-56.

## **8. Copyright Notice**

The author is the only responsible for the printed material included in his paper.

## VERIFICATION OF A DNS CODE OF HIGH ORDER OF ACCURACY USING THE METHOD OF MANUFACTURED SOLUTIONS

**Homero Ghioti da Silva**

Departamento de Engenharia de Materiais, Aeronáutica e Automobilística  
Escola de Engenharia de São Carlos - Universidade de São Paulo, Brasil  
ghioti@sc.usp.br

**Leandro Franco de Souza**

Departamento de Matemática Aplicada e Estatística  
Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo, Brasil  
lefraso@icmc.usp.br

**Marcello Augusto Faraco de Medeiros**

Departamento de Engenharia de Materiais, Aeronáutica e Automobilística  
Escola de Engenharia de São Carlos - Universidade de São Paulo, Brasil  
marcello@sc.usp.br

**Abstract.** *This paper presents a discussion of the results of a verification test using an unsteady manufactured solution in a numerical code of high order of accuracy. The verification procedure was performed by introducing a forcing term in the Navier-Stokes equation creating a nonrealistic problem that has an analytical solution. An important advantage of the Method of Manufactured Solutions compared with standard verification procedure is that it allows exercising the full governing equations sustaining the original boundary conditions implemented in the code. Moreover, if the code computes the fictitious problem without programming errors then the conversion of the code to compute the real problem is done only deleting the source term in the Navier-Stokes equations. The numerical solution was compared with the manufactured solution and a mesh convergence test to quantify the order of accuracy of the numeric code was made. As explained in this paper, the mesh convergence test must be carried out for small simulation time. Then simulation of the fictitious problem for a long time simulation also was performed. Therefore, it was possible to verify if programming error grew after a long time simulation.*

**keywords:** *Temporal Numerical Simulation, Verification Test, Method of Manufactured Solutions, High Order Compact Finite Differences, Spectral Method.*

### 1. Introduction

Due the strong technological advance of computers, Computational Fluids Dynamics (CFD) has been a field of great importance in Fluids Mechanics, and became a strong ally of experimental and theoretical studies for the interpretation of physical phenomena.

Numerical codes are developed and utilized by researchers and engineers and they should give confidence that the results of the numerical simulation are a good representation of the problem under study. Therefore the concepts at verification and validation should be applied. Verification and validation are different concepts, where one can not be substituted by other. According to Roache, 1997, verification is a study of numerical code that consists in assuring that the equations chosen to a given model are resolved correctly and validation assures that the chosen equations are adequate to study the physical problem. In other words, a verification test should give confidence that programming error from implementation of equations, of boundary conditions and of the discretization scheme does not affect the accuracy of the numerical solution. On the other hand, comparisons between the numerical solution of the physical problem with theoretical and experimental results that has been documented in the literature can be classified as validation test.

The proposal of the present paper is to discuss some results of a purely mathematical verification test using the Method of Manufactured Solutions (MMS). This test was applied in a numerical code of high order of accuracy that intended for simulating an incompressible two-dimensional flow. The basic idea of the MMS is to introduce a source term in the equations creating a nonrealistic problem that has an analytical solution. After, this manufactured solution can be used for comparisons with numerical results (Shih, 1985; Steinberg e Roache, 1985). An advantage of using the MMS compared with standard verification procedure, such as code-to-code comparison and the simulation of benchmark solutions, is that the MMS is capable of exercising fully the code

and makes it possible to evaluate all of the leading error terms in the discretization of the equations (Roache, 1998). Also using other methods it is not easy to find a solution that satisfies the boundary conditions of the problem under study. On the other hand, using the MMS it is quite simple to create an analytical solution that satisfies these boundary conditions. In general the MMS does not represent a flow that is realistic or physically realizable, but, it can give good confidence of the numerical code performance.

Some of the first researchers that proposed the use of the MMS for the purpose of code verification are Shih, 1985, and Steinberg e Roache, 1985. Already in 1986 the MMS was recommended as a good verification test by editors of the ASME Journal of Fluids Engineering (JFE). They published a brief policy statement requiring that, in a paper that presents numerical simulation, at least minimal attention should be paid to the quantification of numerical accuracy (Roache e White, 1986). Later, the policy statement was expanded and also adopted by other research journals (Freitas, 1993; Roache, 1994; ASME, 1994; AIAA, 1994; Gresho e Taylor, 1994).

The MMS can be applied together with a systematic mesh convergent test. Therefore it is possible to verify the order of accuracy of the solution scheme and it help to eliminate programming mistakes in the code (Roache, 1998). Nevertheless it is hard to assure that the numerical code is totally free of programming errors and that the order of accuracy obtained is the one expected. This is particularly important in typical simulations in engineering as, for example, flows over solids walls (boundary layer, Poiseuille flow, Couette flow, etc) where it is often necessary to use different orders of the approximations to the discretization in the wall normal direction. These cases results in a more complex multi order solution scheme, where it is hard to estimate a priori the order of accuracy to compare with the order of accuracy obtained via mesh convergent test.

A verification of the order of accuracy of a version of the present numerical code, which uses compact finite difference scheme in the streamwise direction, can be found in Silva *et al.*, 2005. The verification of accuracy of the code was performed using a steady manufactured solution. One important conclusion was that the order of accuracy can depend on the chosen manufactured solution. In particular, on the values of the high order derivatives of the manufactured solution. For the present work an unsteady manufactured solution was used and two tests were performed. The first test was done assuming a small simulation time. This fact was necessary because a mesh refinement test in the wall normal direction was performed. Then the order of accuracy of the numerical code was obtained. The second test was done assuming a long simulation time. The objective of the second test was to verify if programming errors, when it exists in the present code, increases in time.

The governing equations were written in a vorticity-velocity formulation only for the disturbances. The base flow was given and kept constant throughout the simulation. Since the flow is periodic in the streamwise direction then spectral method in this direction was used. Therefore the flow was divided into Fourier modes. Compact finite difference methods of 5<sup>th</sup> and 6<sup>th</sup> order of accuracy in the wall normal direction was utilized for every Fourier modes except for the primary Fourier mode, where compact finite difference of 4th and 5<sup>th</sup> order in this same direction was utilized. For the temporal integration a 4th order Runge Kutta scheme was implemented. The present numerical code was developed to simulate the linear and nonlinear evolution of unsteady infinitesimal disturbance propagating in a plane Poiseuille flow. Moreover, modulated waves as the wavepackets type will be simulated. Some numerical results obtained via the present code were compared with the Linear Stability Theory (LST) and presented in Crepaldi *et al.*, 2006.

The organization of the paper is as follow. Section 2 presents the formulation and the boundary conditions adopted in the current work. Section 3 gives the verification procedure using the MMS together with a mesh convergent test and the results obtained. Section 4 gives some final remarks.

## 2. Formulation

### 2.1. Governing Equations

In the current work the governing equations for an incompressible two-dimensional flow were written in a vorticity-velocity formulation. The velocity components  $\tilde{u}$  and  $\tilde{v}$  and the spanwise vorticity  $\tilde{\omega}_z$  were written in the following form:

$$\tilde{u} = u_B + u; \quad \tilde{v} = v_B + v; \quad \tilde{\omega}_z = \omega_{zB} + \omega_z, \quad (1)$$

where the terms with subscript  $B$  and the terms without subscripts represent the base flow (Poiseuille) and the disturbance components respectively.

Substituting the expression (1) in the spanwise vorticity transport equation and canceling the terms that satisfy the Poiseuille solution, one obtains the Navier-Stokes equation in a disturbance formulation:

$$\frac{\partial \omega_z}{\partial t} + \frac{\partial a}{\partial x} + \frac{\partial b}{\partial y} = \frac{1}{Re} \left( \frac{\partial^2 \omega_z}{\partial x^2} + \frac{\partial^2 \omega_z}{\partial y^2} \right), \quad (2)$$

where  $a = u_B \omega_z + u \omega_{zB} + u \omega_z$  and  $b = v \omega_{zB} + v \omega_z$  are the nonlinear terms.



The continuity equation is

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \quad (3)$$

Taking the  $y$ -derivative of the continuity equation and of the spanwise vorticity

$$\omega_z = \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}, \quad (4)$$

one finds a Poisson equation for  $v$ ,

$$\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} = -\frac{\partial \omega_z}{\partial x}. \quad (5)$$

An analogous scheme can be used to obtain

$$\frac{\partial^2 u}{\partial y^2} - \frac{\partial^2 v}{\partial x \partial y} = \frac{\partial \omega_z}{\partial y} \quad (6)$$

which was used to calculate the  $u$ . More details of this formulation in a three-dimensional version can be found in Marxen, 1998.

The above equations were made dimensionless using the following reference parameters:  $U_{max}$ , which is the maximum value of velocity in the channel, and  $H$ , which is half the channel height. These variables produce the following dimensionless parameters:

$$x = \frac{x^*}{H}, \quad y = \frac{y^*}{H}, \quad u = \frac{u^*}{U_{max}}, \quad v = \frac{v^*}{U_{max}}, \quad \omega_z = \omega_z^* \frac{H}{U_{max}}, \quad t = t^* \frac{U_{max}}{H},$$

where the terms with an asterisk are dimensional. The Reynolds number (Re) is  $\frac{U_{max}H}{\nu}$ , where  $\nu$  denotes the kinematic viscosity.

## 2.2. Boundary Conditions

A periodic boundary condition was adopted in the streamwise direction ( $x$ -direction) and for the wall normal direction ( $y$ -direction) a non-slip and impermeability ( $u = v = 0$  in the walls) conditions were imposed. The flow geometry is presented in fig. 1.

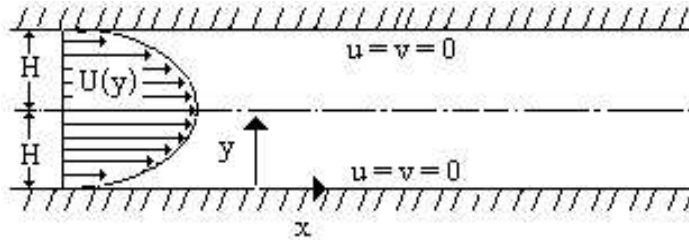


Figure 1: Schematic diagram of physical system

A crucial problem of the vorticity-velocity formulation is that there are no explicit boundary conditions for the vorticity at the wall. It must be computed from the velocity field assuring that the boundary conditions for the velocity are satisfied maintaining consistency and conservation of mass. The calculation of the vorticity at the wall was performed via the equations (5) and (6) and using the fact  $\frac{\partial v}{\partial x} = 0$  at the wall.

### 2.3. Numerical Method

In the  $x$ -direction, a spectral method was used. In this method, a generic quantity  $s$  can be decomposed in Fourier modes ( $\alpha_k$ ) of the following form:

$$s(x, y, t) = \sum_{k=0}^K S_k(y, t) e^{-i\alpha_k x}, \quad (7)$$

where  $s$  represents the variables  $u$ ,  $v$ ,  $w_z$  and  $a$  and  $b$  of equation (2).  $S_k$  represents the discrete Fourier components of the function  $s$ .

Substituting equation (7) in equations (2), (5) and (6) one obtains

$$\frac{\partial \Omega_{z_k}}{\partial t} + i\alpha_k A_k + \frac{\partial B_k}{\partial y} = \frac{1}{Re} (-\alpha_k^2 \Omega_{z_k} + \frac{\partial^2 \Omega_z}{\partial y^2}), \quad (8)$$

$$\frac{\partial^2 V_k}{\partial y^2} - \alpha_k^2 V_k = -i\alpha_k \Omega_{z_k}, \quad (9)$$

$$\frac{\partial^2 U_k}{\partial y^2} = \frac{\partial \Omega_{z_k}}{\partial y} + i\alpha_k \frac{\partial V_k}{\partial y}. \quad (10)$$

All the Fourier modes  $\alpha_k$  (except the primary Fourier mode  $\alpha_0 = 0$ ) were calculated independently using the equations (8) to (10). The primary Fourier mode calculation  $\alpha_0 = 0$  was presented in 2.3.1.

In the  $y$ -direction, a compact finite difference scheme was used. At the points in the interior of the domain, the discretization scheme used centered compact finite difference of 6<sup>th</sup> order of accuracy. For the points in and near wall boundaries, asymmetric schemes of 5<sup>th</sup> and 6<sup>th</sup> order of accuracy were adopted, except to the wall vorticity calculation, where compact finite difference of 6<sup>th</sup> order of accuracy was used. Full Details of the chosen finite differences can be found in Souza *et al.*, 2005. The chosen finite differences for the calculations in the  $y$ -direction when  $\alpha_0 = 0$  was presented in 2.3.1.

#### 2.3.1. The Primary Fourier Mode $\alpha_0$

Making  $\alpha_0 = 0$  in the equations (8) to (10), one obtains the following equations:

$$\frac{\partial \Omega_{z_0}}{\partial t} + \frac{\partial B_0}{\partial y} = \frac{1}{Re} \left( \frac{\partial^2 \Omega_{z_0}}{\partial y^2} \right), \quad (11)$$

$$\frac{\partial^2 V_0}{\partial y^2} = 0, \quad (12)$$

$$\frac{\partial^2 U_0}{\partial y^2} = \frac{\partial \Omega_{z_0}}{\partial y}. \quad (13)$$

The equation (11) was used to calculation  $\Omega_{z_0}$  only at the interior of the domain. The equation (12) together with the impermeability condition resulted in  $V_0 = 0$  for every domain. The equation (13) was used to calculation  $U_0$  for every domain. Since there is no boundary condition for  $\Omega_{z_0}$  at the wall, it was calculated using the vorticity definition equation

$$\Omega_{z_0} = \frac{\partial U_0}{\partial y}. \quad (14)$$

Note that the equation (14) cannot precede the equation (13) and vice-versa. Rather, the equations (13) and (14) should be calculated simultaneously. Two different forms to calculation the equations (13) and (14) were used. The first was an iterative procedure between the equations (13) and (14). But it resulted in a numerical instability. Another form to calculate the equations (13) and (14) was done calculating the equation

(13) independently of the  $\Omega_{z_0}$  calculation at the wall. For this case, the discretization of the term  $\frac{\partial \Omega_{z_0}}{\partial y}$  in the equation (13) was done using the following finite difference:

For the boundary point ( $y = 0$ ) the following noncentered approximation was adopted:

$$-12 \frac{\partial \Omega_{z_0}}{\partial y} \Big|_1 + 137 \frac{\partial \Omega_{z_0}}{\partial y} \Big|_2 = \frac{-25010\Omega_{z_{02}} + 40080\Omega_{z_{03}} - 21240\Omega_{z_{04}} + 7280\Omega_{z_{05}} - 1110\Omega_{z_{06}}}{120\Delta y} + O(\Delta y^4). \quad (15)$$

The near boundary point approximation was:

$$\frac{\partial \Omega_{z_0}}{\partial y} \Big|_2 + 5 \frac{\partial \Omega_{z_0}}{\partial y} \Big|_3 = \frac{-2364\Omega_{z_{02}} - 300\Omega_{z_{03}} + 3600\Omega_{z_{04}} - 1200\Omega_{z_{05}} + 300\Omega_{z_{06}} - 36\Omega_{z_{07}}}{720\Delta y} + O(\Delta y^5), \quad (16)$$

and for central points a centered approximation was adopted:

$$\frac{\partial \Omega_{z_0}}{\partial y} \Big|_{j-1} + 4 \frac{\partial \Omega_{z_0}}{\partial y} \Big|_j + \frac{\partial \Omega_{z_0}}{\partial y} \Big|_{j+1} = \frac{3\Omega_{z_{0j-1}} + 3\Omega_{z_{0j+1}}}{\Delta y} + O(\Delta y^4). \quad (17)$$

For the points at and next to the boundary of the opposite wall ( $y = 2$ ) the approximations (15) and (16) were used but with a sign inversion. After the  $U_0$  was calculated, then  $\Omega_{z_0}$  was calculated using the equation (14).

### 2.3.2. Temporal Integration

The temporal integration was done with an explicit 4th order Runge-Kutta method of four stages (Ferziger e Peric, 1997). The numerical procedure consisted of the following form: First the manufactured solution (in a time instant  $t = 0$ ), presented in the section 3, was introduced in the code as initial condition. Then for each stage of Runge-Kutta method, the following calculations were necessary:

1. Calculate the nonlinear terms  $A_k$  and  $B_k$  in the physical domain and transforms back to the Fourier space;
2. Calculate the right-hand side of the vorticity transport equation (8);
3. Integrate the vorticity transport equation in a stage;
4. Calculate the velocity  $V_k$  using the equation (9);
5. Calculate the wall vorticity  $\Omega_{z_k}$  ( $k \neq 0$ ) using the same equation (9).
6. calculate the velocity  $U_0$  and the vorticity at wall  $\Omega_{z_0}$  as described in the section 2.3.1.
7. Calculate the velocity  $U_k$  ( $k \neq 0$ ) using the equation (10).
8. Turn off to item 1 and repeat the above calculation until the time simulation expected.

As the Fourier modes were calculated independently, is not necessary the item 6 to be performed before item 7, it was done in this sequence only for a better organization of the numerical code routines.

### 3. The Verification Procedure: The Method of Manufactured Solutions

The current numerical code was verified using the Method of Manufactured Solutions (MMS). The chosen manufactured solution has the following form:

$$\begin{aligned} u(x, y) &= Ae^y y(y + 1 - \sqrt{5})(y + 1 + \sqrt{5})(y - 2) \cos(\alpha x - \omega t), \\ v(x, y) &= A \sin(\alpha x - \omega t) \alpha e^y y^2 (y^2 - 4y + 4), \\ \omega_z(x, y) &= -Ae^y \cos(\alpha x - \omega t) (8y^2 - 8 + 4\alpha^2 y^2 + \alpha^2 y^4 - 4\alpha^2 y^3 - 4y^3 + 8y - y^4). \end{aligned}$$

where  $A$  is an amplitude. The manufactured solution was produced using the following procedure: first, the  $u$ -component was chosen to satisfy the boundary conditions. The expression  $e^y$  that appear in  $u$  allows that the  $y$ -derivatives of any order were non-zero. The  $v$  was obtained using the continuity equation. For the  $v$  to satisfy the boundary conditions, the expressions  $(y + 1 - \sqrt{5})$  and  $(y + 1 + \sqrt{5})$  in  $u$  were necessary.  $\omega_z$  was then calculated from vorticity definition via Eq. (4). The requirement of nonzero  $y$ -derivatives allowed a systematic mesh refinement test for obtaining the order of accuracy of the code in the  $y$ -direction. Moreover the chosen manufactured solution oscillates in time with frequency  $\omega$  with a constant amplitude. It allowed to

imitate a neutral Tollmien-Schlichting wave that propagated in the flow direction in time with frequencies  $\omega$  and wavenumber  $\alpha$  along of a Poiseuille flow. Furthermore the chosen manufactured solution was not exponential in time to avoid confusion with numerical instability problem (Ethier e Steinman, 1994). The parameters  $\alpha$  and  $\omega$  were also made dimensionless by  $U_{max}$  and  $H$ . Therefore  $\alpha = \alpha^*H$  and  $\omega = \frac{\omega^*H}{U_{max}}$ .

Figure 2 shows the surfaces plot of  $u$ ,  $v$  and  $\omega_z$  of the manufactured solution and the energy spectrum of the Fourier modes in streamwise direction of the  $\Omega_z$ . The chosen parameters values were  $A = 5 \times 10^{-4}$ ,  $Re = 10^4$  and  $\alpha = \omega = 1$ . These parameters values were used when a long time simulation was performed. Contrary of the mesh refinement test, where the chosen Reynolds number and amplitude were  $A = Re = 1$ .

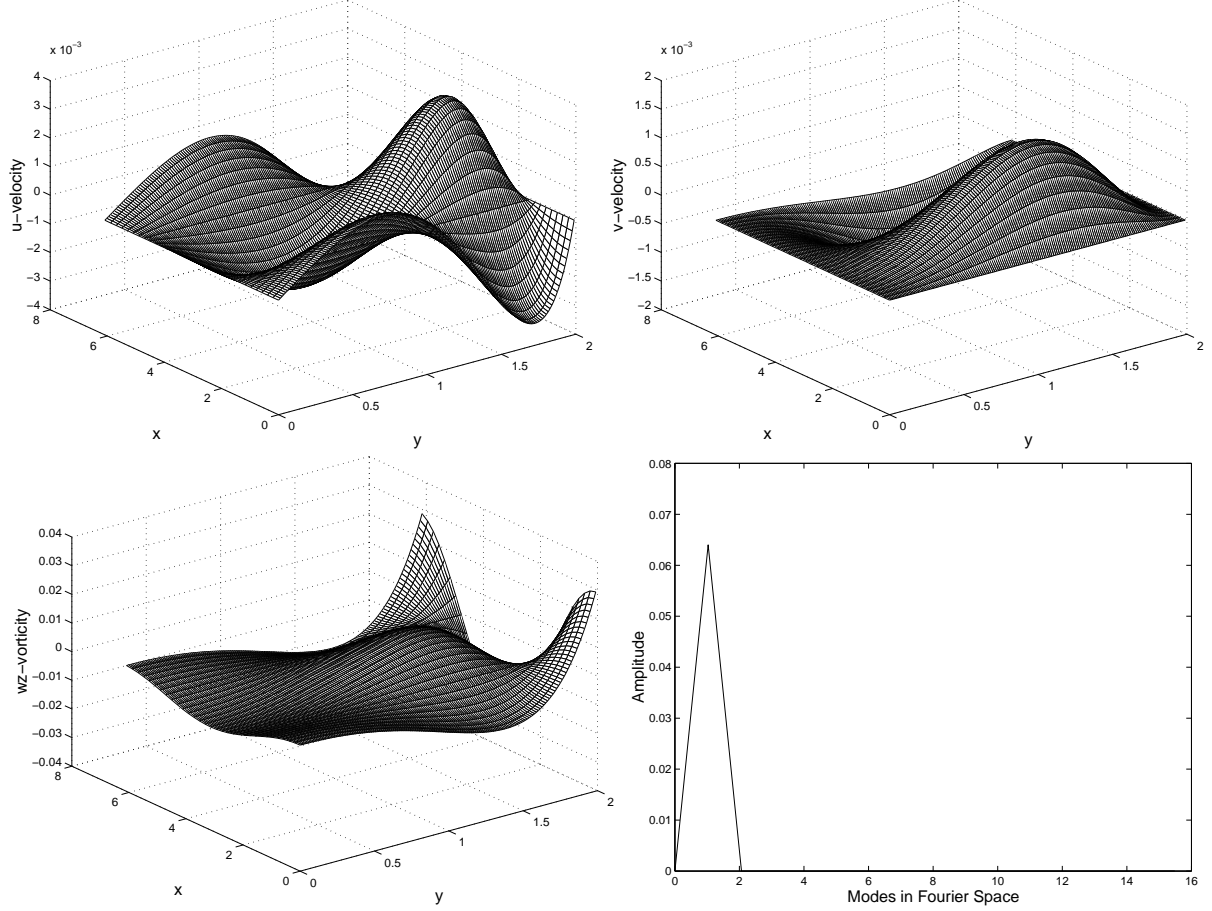


Figure 2: Surfaces plot of the chosen manufactured solution in the instant  $t = 0$ . This manufactured solution was used as initial condition for all the numerical simulations presented in the paper, but assuming different parameters values for each simulation.

The manufactured solution described above is analytical solution of the following fictitious problem:

$$\frac{\partial \omega}{\partial t} + \frac{\partial a}{\partial x} + \frac{\partial b}{\partial y} = \frac{1}{Re} \left( \frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} \right) + Forc(x, y, t) \quad (18)$$

where the forcing term in physical space is

$$\begin{aligned} Forc(x, y, t) = & -(16\cos(\alpha x - \omega t)\alpha^2 y^2 - 8\sin(\alpha x - \omega t)\omega Re + 4\cos(\alpha x - \omega t)\alpha^4 y^2 + \\ & \cos(\alpha x - \omega t)\alpha^4 y^4 - 4\cos(\alpha x - \omega t)\alpha^4 y^3 - 8\cos(\alpha x - \omega t)\alpha^2 y^3 - 2\cos(\alpha x - \omega t)\alpha^2 y^4 + \\ & 18y^4 \sin(\alpha x - \omega t)\alpha Re + 8\sin(\alpha x - \omega t)\omega y Re + 12y^4 \sin(\alpha x - \omega t)\alpha^3 Re - \\ & 16y^2 \sin(\alpha x - \omega t)\alpha Re - 4\sin(\alpha x - \omega t)\omega Re y^3 - \\ & 8Ae^y y^4 \cos(\alpha x - \omega t)\sin(\alpha x - \omega t)\alpha Re + 64Ae^y y \cos(\alpha x - \omega t)\sin(\alpha x - \\ & \omega t)\alpha Re - 16y^3 \sin(\alpha x - \omega t)\alpha Re + 16y \sin(\alpha x - \omega t)\alpha Re - 8y^3 \sin(\alpha x - \omega t)\alpha^3 Re - \\ & 4\sin(\alpha x - \omega t)\omega Re \alpha^2 y^3 + 4\sin(\alpha x - \omega t)\omega Re \alpha^2 y^2 + \sin(\alpha x - \omega t)\omega Re \alpha^2 y^4 - \\ & 6y^5 \sin(\alpha x - \omega t)\alpha^3 Re + y^6 \sin(\alpha x - \omega t)\alpha^3 Re - 2y^5 \sin(\alpha x - \omega t)\alpha Re - \\ & y^6 \sin(\alpha x - \omega t)\alpha Re + \cos(\alpha x - \omega t)y^4 + 16\cos(\alpha x - \omega t)\alpha^2 y + \end{aligned}$$

$$\begin{aligned}
& 96Ae^y y^3 \cos(\alpha x - \omega t) \sin(\alpha x - \omega t) \alpha Re + \\
& 8Ae^y y^6 \cos(\alpha x - \omega t) \sin(\alpha x - \omega t) \alpha Re - \\
& 24Ae^y y^5 \cos(\alpha x - \omega t) \sin(\alpha x - \omega t) \alpha Re - \sin(\alpha x - \omega t) \omega Re y^4 - \\
& 128Ae^y y^2 \cos(\alpha x - \omega t) \sin(\alpha x - \omega t) \alpha Re + 28 \cos(\alpha x - \omega t) y^2 + \\
& 12 \cos(\alpha x - \omega t) y^3 - 16 \cos(\alpha x - \omega t) y - 16 \cos(\alpha x - \omega t) \alpha^2 - 24 \cos(\alpha x - \omega t) + \\
& 8 \sin(\alpha x - \omega t) \omega Re y^2) Ae^y / Re
\end{aligned}$$

The verification test was carried out of two different forms: first, as suggested by Roache, 1998, the order of accuracy of the code via a mesh refinement test in the  $y$ -direction using a small time simulation was performed. This simulation time was sufficient to the manufactured solution to migrate to numerical solution. After, numerical simulation of the manufactured solution using an infinitesimal amplitude  $A$  and it imitating a neutral Tolmien-Schilitching wave in the Poiseuille flow was performed, where a long time simulation was adopted. Contrary to the mesh refinement test that require, as suggested by Ethier e Steinman, 1994, simulation time very small to avoid truncation errors in time.

The mesh refinement test was realized using double precision in the calculations to avoid round-off error, the resolution in Fourier space was chosen to be  $\frac{2}{3}$  compared to the resolution in physical space to avoid aliasing error and a hundred time steps with time step ( $dt = 10^{-5}$ ) was the time simulation to avoid time truncation error. Therefore, the round-off error, aliasing error and time truncation error were negligible relative for the truncation error in  $y$ -direction. According to the above strategy, only the truncation errors relative to the  $y$ -direction calculations and possible programming errors predominated in the total numerical error. It was denoted by  $E_m$ . The mesh refinement test was made using the parameters  $A$  and  $Re$  set to 1 assuring that the terms of the studied equations have the same magnitude order (Roache, 1998). The total numerical error  $E_m$  is expressed by

$$E_m = |f_m - f_{exact}| \simeq g_p dy_m^p, \quad (19)$$

where  $f_m$  represents some numerical solution on mesh  $m$ ,  $f_{exact}$  represents the analytical solution of the chosen fictitious problem,  $g_p$  is the coefficient (dependent on high-order  $y$ -derivatives) of the leading error term and  $p$  is the order of accuracy. Applying the logarithmic function in equation (19) for the a finer mesh  $m$  and a coarser mesh  $m + 1$  it is possible to obtain the order of accuracy by using the following equation

$$\ln\left(\frac{E_{m+1}}{E_m}\right) = p \ln(r_m), \quad (20)$$

with  $r_m = \frac{dy_{m+1}}{dy_m}$ .

The order of accuracy for the present code was determined using six different meshes ( $m = 1, \dots, 6$ ) and the number of intervals doubled from one mesh to the following mesh. The first mesh used was  $dy = 0.25$  and the spatial domain in the  $y$ -direction was 2. This gives eight intervals in the  $y$ -direction.

Figures 3 to 5 show the behavior of the average and maximum absolute errors for  $u$ ,  $v$  and  $w_z$  respectively. The results were plotted in logarithmic scale. Fig. 3 and 4 show results where the straight line has a slope of 6 for both the mean and the maximum errors. On the other hand, Fig. 5 shows  $6^{th}$  order for the mean error and  $5^{th}$  order for the maximum error. The resultant  $6^{th}$  order for the maximum error occurred because the numerical error of the wall vorticity calculation is predominant over the numerical error in the interior of the domain. This predominance occurred principally because of the boundary condition calculation to  $\Omega_z$ , where the discretization scheme utilized the great values of the high order  $y$ -derivatives of  $\Omega_z$  in the wall points. More details on this issue can be found in Silva *et al.*, 2005. But, the resultant order was coherent with the theoretical order.

Despite the obtained order of accuracy was satisfactory, the manufactured solution as an infinitesimal disturbance in a long time simulation was also performed. The objective of this simulation was to verify if programming error increased in time. To the numerical solution of the equation (18) to imitate a neutral Tolmien-Schilitching wave in Poiseuille flow, the values of the parameters  $A$  and  $Re$  were  $5 \times 10^{-4}$  and  $10^4$  respectively. This manufactured solution only oscillates in time with constant amplitude and none solution deformation should appear after a long simulation time. The chosen time step for the present simulation was  $dt = 0.1454$  with 14400 time steps. This results in a total simulation time in turn of 90 cycles in time. Figure 6 shows the energy spectrum in Fourier domain of this manufactured solution after the time simulation above described. Comparisons between the numerical simulations with and without the primary Fourier mode calculation were realized. Note that there was a nonphysical energy quantitative in the primary Fourier mode when the primary Fourier mode calculation was performed. This occurred because there is a numerical instability problem in the primary Fourier mode calculation. The results of this simulation showed that the chosen solution scheme presented in the section 2.3.1 was not sufficient to eliminate the numerical instability in the primary Fourier mode calculation. In attempt to obtain greater stability for this calculation, a semi implicit scheme to the temporal integration of the wall normal diffusion terms will be implemented in the present code.

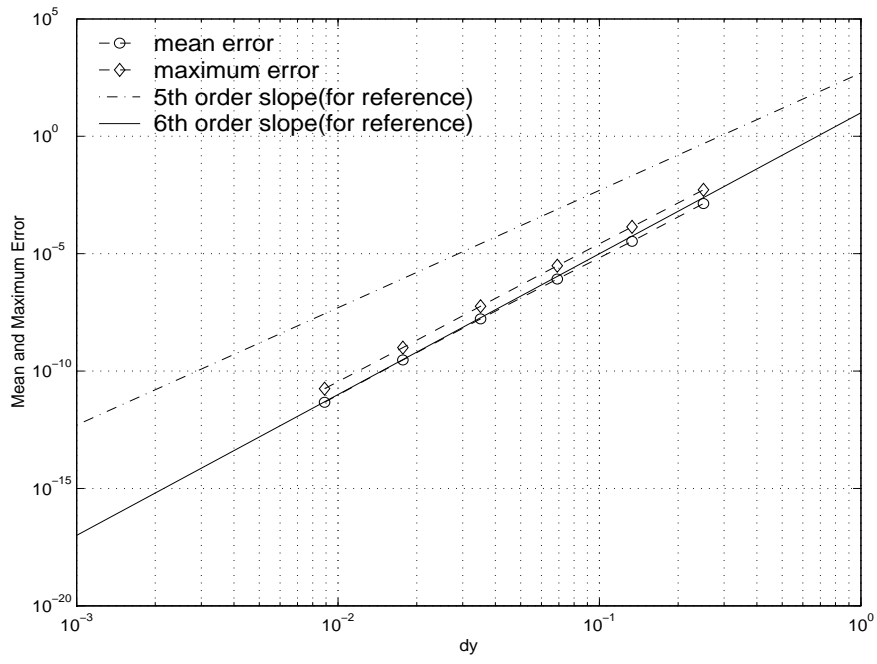


Figure 3: Behavior of the absolute mean and maximum errors  $E_m$  for the  $u$ -velocity.

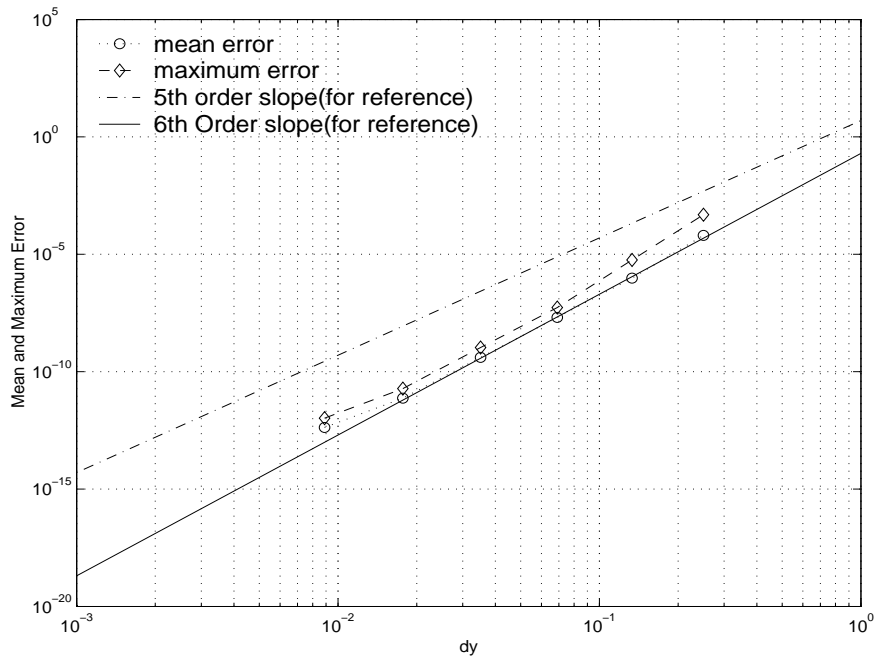


Figure 4: Behavior of the absolute mean and maximum errors  $E_m$  for the  $v$ -velocity.

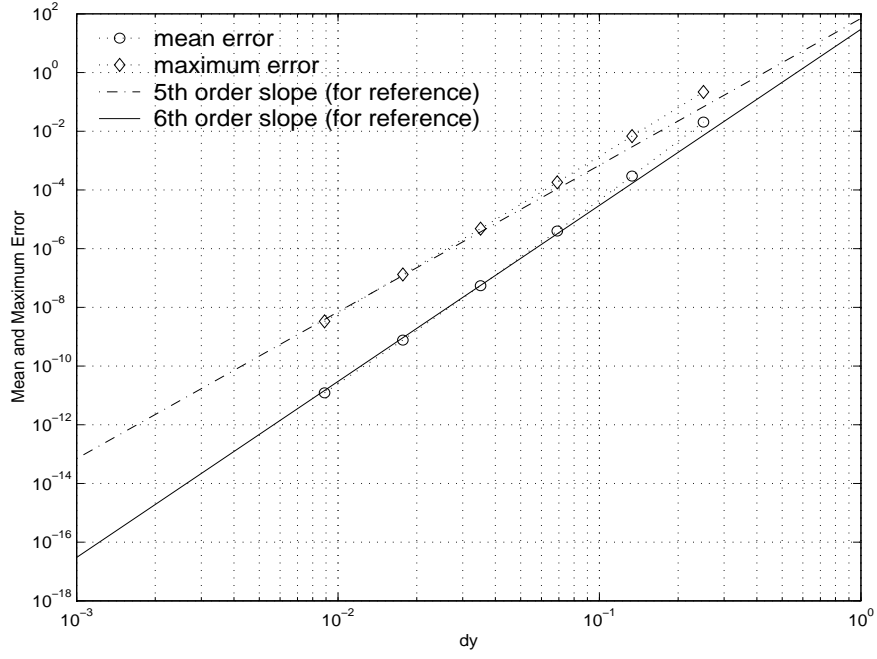


Figure 5: Behavior of the absolute mean and maximum errors  $E_m$  for the  $w_z$ -vorticity.

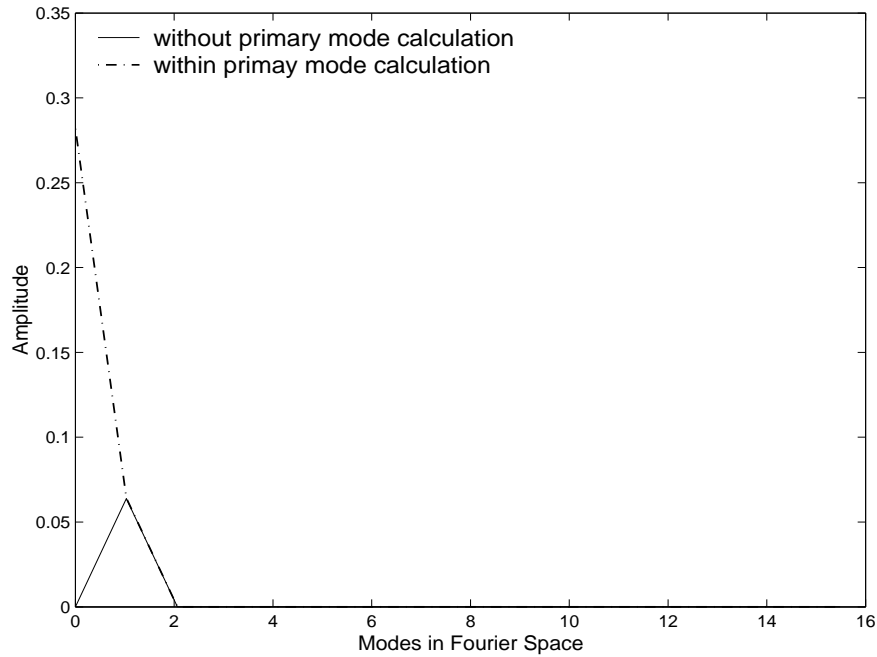


Figure 6: Comparison of the energy spectrum of  $w_z$  where the calculations were performed with (dashed line) and without (solid line) the primary Fourier mode calculation. Note the great amplitude of the primary Fourier mode presented by dashed line. The simulation time was in turn of 90 cycles in time.

## 4. Final Remarks

In the current work a verification test of a numerical code of high order of accuracy was presented. The verification test was done using the Method of Manufactured Solutions. Despite the MMS to be a general and very powerful approach to code verification, it was easy to obtain a manufactured solution that satisfied the boundary conditions yet implemented in the code without simplifications of the governing equations. Consequently, the nonlinear terms in the Navier-Stokes equations were also estimated. Contrary to others verification methods as comparison code-to-code or simulation of benchmark solutions, it was possible to exercise all of the terms in the equations and all of the leading error terms in the discretization scheme. The code verification was done using one test with a small time simulation and another test with a long time simulation. The test with a small simulation time was a mesh refinement test combined with order of accuracy verification. Since the discretization scheme used in this work had different orders of accuracy in the  $y$ -direction, depending on the grid position and on the equation, a lot of attention was required in the analysis of the results. Nevertheless, the resultant order was coherent with the theoretical order of the discretization scheme. For the  $u$  and  $v$ -components of velocity, the obtained order was six. But, to the spanwise vorticity  $\Omega_z$  the obtained order was five, due the discretization error from wall vorticity points was dominant compared with the discretization error from others points of the domain. Another important result was that the use of the average error is not a good technique for the study of the code order. Because it gave order six for the  $\Omega_z$  while that the maximum error technique gave order five. It helped to conclude that in the code order study, the analysis of the maximum error from chosen approximation scheme also is important.

The test with a long simulation time was also performed, using the same manufactured solution that was used to the code order analysis but with different parameters. The parameters  $A$ ,  $Re$ ,  $\alpha$  and  $\omega$  were chosen to the manufactured solution to imitate a neutral Tollmien-Schlichting wave propagating in a Poiseuille flow. It helped to identify if programming error increased in time. The results showed that a numerical instability appeared when the primary Fourier mode was calculated. But, the current numerical code had a good performance for the numerical simulations that despise it.

The numerical code will be used to simulate the temporal nonlinear development of modulated and unmodulated waves in a plane Poiseuille flow. Preliminary studies showed that the primary Fourier modes calculation can not have a strong relevance on the results of these future simulations. Therefore, the paper author's believe that numerical simulations without the primary Fourier modes calculation also will be proper to understand the behavior of the modulated and unmodulated waves in a plane Poiseuille flow in a highly nonlinear stage.

The code validation has been performed comparing the numerical simulation with primary and secondary stabilities results. Preliminary results of validation of the presented code can be found in Crepaldi *et al.*, 2006.

## 5. Acknowledgments

The financial support of the CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) and FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo), grant 04/07507-4 are gratefully acknowledged.

## 6. Responsibility Notice

The authors are the only responsible for the printed material included in this paper.

## 7. References

- AIAA, 1994, Policy Statement on Numerical Accuracy and Experimental Uncertainty, "AIAA Journal", Vol. 32, pp. 3-3.
- ASME, 1994, Journal of Heat Transfer Editorial Policy Statment on Numerical Accuracy, "ASME Journal of Heat Transfer", Vol. 116, pp. 797-798.
- Crepaldi, F., Silva, H. G., Souza, L. F., and Medeiros, M. A. F., 2006, Temporal Numerical Simulation of Disturbances Propagation in a Poiseuille Flow Using Spectral Methods, "Escola de Primavera de Transição e Turbulência-EPTT", Vol. 1, Rio de Janeiro, Brazil.
- Ethier, C. R. and Steinman, D. A., 1994, Exact Fully 3-D Navier/Stokes Solutions for Benchmarking, "Int. J. Num. Meth. Fluids", Vol. 19, pp. 369-375.
- Ferziger, J. H. and Peric, M., 1997, "Computational Methods for Fluid Dynamics", Springer, Berlin, Germany.
- Freitas, C. J., 1993, Editorial Policy Statement on the Control of Numerical Accuracy, "ASME Journal of Fluids Engineering", Vol. 115, pp. 339-340.
- Gresho, P. M. and Taylor, C., 1994, editorial, "Int. J. Num. Meth. Fluids", Vol. 19, pp. 1-2.
- Marxen, O., 1998, "Temporal Numerical Simulation of a Flat Plate Boundary Layer", PhD thesis, University of Arizona, U.S.A.



- Roache, P. J., 1994, Response: To the Comments by Drs. W. Shyy and M. Sindir, "ASME Journal of Fluids Engineering", Vol. 116, pp. 198–199.
- Roache, P. J., 1997, Quantification of Uncertainty in Computational Fluid Dynamics, "Annu. Rev. Fluid. Mech.", Vol. 29, pp. 123–160.
- Roache, P. J., 1998, "Verification and Validation in Computational Science and Engineering", Hermosa Publishers, New México, USA.
- Roache, P. J. and White, F. M., 1986, Editorial Policy Statement on the Control of Numerical Accuracy, "ASME Journal of Fluids Engineering", Vol. 108, pp. 16–42.
- Shih, T. M., 1985, A Procedure to Debug Computer Programs, "Int. J. Num. Meth. Fluids", Vol. 21, pp. 1027–1037.
- Silva, H. G., Medeiros, M. A. F., and Souza, L. F., 2005, A Verification Test for a Direct Numerical Simulation Code that Uses a High Order Discretization Scheme, "18th International Congress of Mechanical Engineering, Proceedings of the COBEM", Ouro Preto, Brazil.
- Souza, L. F., Mendonça, M. T., and Medeiros, M. A. F., 2005, The Advantages of Using High-Order Finite Differences Schemes in Laminar-Turbulent Transition Studies, "Int. J. Num. Meth. Fluids", Vol. 48, pp. 565–582.
- Steinberg, S. and Roache, P. J., 1985, Symbolic Manipulation and Computational Fluid Dynamics, "Journal of Computational Physics", Vol. 30, pp. 251–283.

## GENERACIÓN NUMÉRICA DE MALLAS PARA EL ANÁLISIS DE ESCURRIMIENTOS REACTIVOS EN CÁMARAS DE COMBUSTIÓN

**Pedro Andrés Pacheco Fernandoy**

Departamento de Engenharia Mecânica – COPPE, Universidade Federal do Rio de Janeiro, C.P. 68503 – Rio de Janeiro  
CEP: 21945-970, Brasil.  
e-mail: pedrop@lmt.coppe.ufjf.br

**Albino José Kalab Leiroz**

Departamento de Engenharia Mecânica – COPPE, Universidade Federal do Rio de Janeiro, C.P. 68503 – Rio de Janeiro  
CEP: 21945-970, Brasil.  
e-mail: leiroz@lmt.coppe.ufjf.br

**Resumen.** La creciente demanda observada recientemente en el uso de turbinas a gas ha motivado el estudio numérico de sus principales componentes de forma de obtener un mejor entendimiento de los fenómenos físicos involucrados. En el presente trabajo, técnicas de generación numérica de mallas son aplicadas en geometrías axisimétricas similares a las de las cámaras de combustión de turbinas a gas del tipo tubular. Un cuerpo esférico es ubicado a lo largo del eje de simetría de la cámara, de forma de permitir el estudio del efecto del inyector de combustible. Efectos de inyección de aire secundario son también considerados. Para esta geometría irregular, técnicas de generación numérica de mallas y transformación de coordenadas fueron aplicadas para el mapeamiento del dominio físico utilizado y la solución de las ecuaciones de las variables de interés. Para el presente estudio, el escurrimiento fue considerado incompresible, laminar, y con propiedades termo físicas constantes. El campo de velocidades fue obtenido a través de la formulación Vorticidad-Función Corriente. Las ecuaciones de conservación de energía y especies son resueltas utilizando potenciales de Shvab-Zel'dovich. Los resultados permiten la determinación de la posición y forma de la llama, y la calibración de los parámetros de las mallas generadas.

**Palabras Claves:** Cámara de Combustión, Escurrimiento Reactivo, Generación de Mallas, Vorticidad-Función Corriente, Shvab-Zel'dovich.

### 1. Introducción

El proceso de combustión es uno de los procesos químicos más importantes, debido a que el desarrollo de la civilización actual está basado en el consumo de energía, generalmente proporcionada por el proceso de quema de combustibles. De hecho el proceso de combustión se encuentra aplicado a una gran cantidad de situaciones, como por ejemplo en la generación de calor en calderas para la producción de vapor, y principalmente para la producción de potencia en un eje o empuje. Estos se refieren a los motores de los vehículos de transporte y a los motores de las plantas termoeléctricas en general. En cuanto a los motores de combustión interna, tenemos a los motores alternativos y las turbinas de gas. Los motores alternativos tienen su campo de aplicación principalmente dirigido a la industria automotriz, en cuanto que las turbinas de gas, fueron desarrolladas inicialmente en el campo de la aeronáutica, y últimamente aplicadas al sector industrial. El tratamiento de la combustión requiere sólidas bases de Termodinámica, Mecánica de Fluidos, Transferencia de Calor en todos sus mecanismos, Transferencia de Masa y Cinética Química. La mayor dificultad es que en la mayoría de las situaciones prácticas el escurrimiento es altamente turbulento y confinado a geometrías complejas. Con relación a la complejidad geométrica, como la de las cámaras de combustión de turbinas de gas, diversos trabajos han sido realizados para abarcar esta dificultad (Mare *et al.*, 2004; Selle *et al.*, 2004 y Jurgen *et al.*, 2001) a través de la utilización de Softwares comerciales usando mallas estructuradas, no estructuradas, con inyectores del tipo torbellinador (*swirl*) o de cuerpo estabilizador (*bluff-body*). Estos inyectores tienen como principal objetivo mantener la estabilidad de la llama a través de la creación de una zona de recirculación inducida por este dispositivo (Menom *et al.*, 2001 y Saravanamuttoo *et al.*, 2001).

En el presente estudio, se propone una geometría que procura una semejanza a las cámaras de combustión de turbinas de gas del tipo tubular, usando un inyector del tipo cuerpo estabilizador, cuya geometría utilizada para el presente estudio fue esférica, como se muestra en la Fig. 1. El presente trabajo modela combustión en régimen laminar, en condiciones de convección forzada y con propiedades termo físicas constantes.

## 2. Modelaje Matematico

### 2.1 Generación de Mallas

La técnica de generación numérica de mallas de Thompson (Özisik, 1994) fue adoptada. Entretanto, el esquema iterativo de Hilgenstock (Hofmann, 1992) fue adoptado para el control de las líneas de las mallas a través de las funciones de control de malla,  $P(\xi, \eta)$  y  $Q(\xi, \eta)$ . El método de Hilgenstock establece como condiciones a satisfacer, el espaciamiento de las líneas de la malla adyacente al contorno, y el ángulo de intersección de las líneas de la malla que llegan a este contorno. La condición de ortogonalidad es siempre deseada.

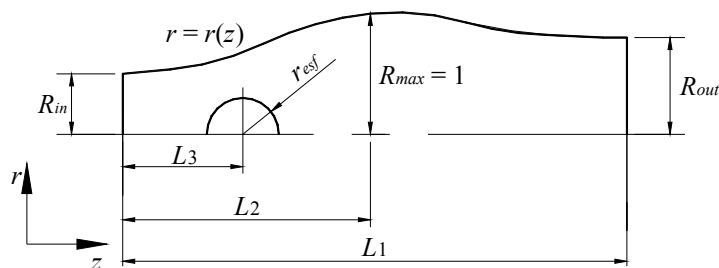


Figura 1. Geometría de la Cámara de Combustión a Modelar.

La malla es generada en el sistema de coordenadas cartesiano  $(x,y)$  a partir de un sistema de coordenadas computacional  $(\xi, \eta)$  ya conocido (Özisik, 1994 y Hofmann, 1992), a partir de las siguientes ecuaciones

$$\alpha \frac{\partial^2 x}{\partial \xi^2} - 2\beta \frac{\partial^2 x}{\partial \eta \partial \xi} + \gamma \frac{\partial^2 x}{\partial \eta^2} + J^2 P(\xi, \eta) \frac{\partial x}{\partial \xi} + J^2 Q(\xi, \eta) \frac{\partial x}{\partial \eta} = 0 \quad (1)$$

$$\alpha \frac{\partial^2 y}{\partial \xi^2} - 2\beta \frac{\partial^2 y}{\partial \eta \partial \xi} + \gamma \frac{\partial^2 y}{\partial \eta^2} + J^2 P(\xi, \eta) \frac{\partial y}{\partial \xi} + J^2 Q(\xi, \eta) \frac{\partial y}{\partial \eta} = 0 \quad (2)$$

donde  $\alpha$ ,  $\beta$  y  $\gamma$  son los coeficientes geométricos, dados por  $\alpha = x_\eta^2 + y_\eta^2$ ,  $\beta = x_\eta x_\xi + y_\eta y_\xi$  y  $\gamma = x_\xi^2 + y_\xi^2$ . Entretanto,  $J$  es el Jacobiano de la transformación de coordenadas, expresado por  $J = x_\xi y_\eta - x_\eta y_\xi$ , y  $x_\eta$ ,  $x_\xi$ ,  $y_\eta$  e  $y_\xi$  son las métricas de la transformación inversa (Özisik, 1994 y Hofmann, 1992). Las funciones de control de malla,  $P(\xi, \eta)$  y  $Q(\xi, \eta)$ , de acuerdo al esquema iterativo de Hilgenstock, son evaluadas a partir de

$$P^{n+1} = P^n + \Delta P^n \quad Q^{n+1} = Q^n + \Delta Q^n \quad (3)$$

donde  $\Delta P^n$  y  $\Delta Q^n$  son las correcciones necesarias para obtener los valores de  $P(\xi, \eta)$  y  $Q(\xi, \eta)$  que proporcionen los valores necesarios para satisfacer las condiciones de espaciamiento y ortogonalidad en los contornos de la malla. El símbolo  $n$  indica el nivel de iteración. Los valores de  $\Delta P^n$  y  $\Delta Q^n$  son determinados de acuerdo con el mapeamiento adoptado, pues los valores de  $P(\xi, \eta)$  tienen influencia en las líneas coordenadas  $\xi$ , y los valores de  $Q(\xi, \eta)$  en las líneas coordenadas  $\eta$ , como observado en la Fig. 2. El mapeamiento adoptado es presentado en la Fig. 3. De la Figura 3 se observa que la malla es generada para toda la geometría, sin hacer uso de su simetría. La técnica de mapeamiento es conocida como Técnica de Ventana (Özisik, 1994 y Hofmann, 1992).

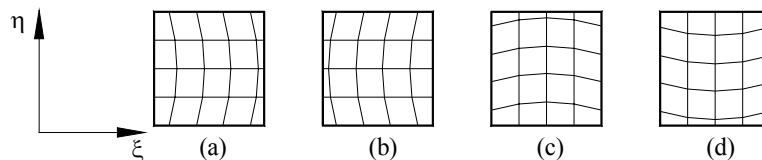


Figura 2. Efectos de  $P(\xi, \eta)$  y  $Q(\xi, \eta)$  en las líneas coordenadas de la malla. (a) y (b)  $Q(\xi, \eta) = 0$  y  $P(\xi, \eta) > 0$  y  $P(\xi, \eta) < 0$ , (c) y (d)  $P(\xi, \eta) = 0$  y  $Q(\xi, \eta) > 0$  y  $Q(\xi, \eta) < 0$ .

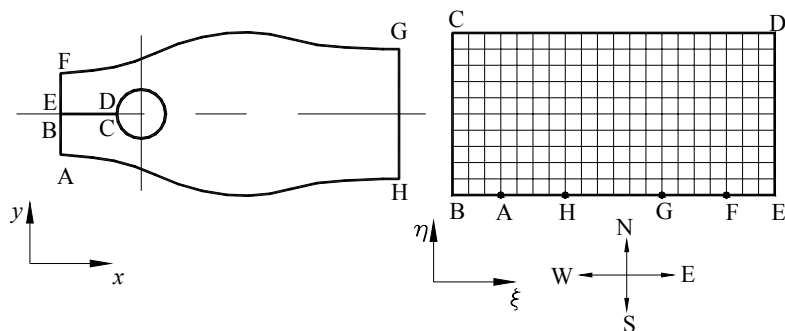


Figura 3. Mapeamento adoptado.

De acuerdo con el mapeamento adoptado, haciendo referencias a las Figs. 2 y 3, el control del espaciamento para el contorno sur del dominio computacional, (Fig. 3), será a través de los valores de  $Q(\xi, \eta)$ . En cuanto que el control del ángulo de intersección de las líneas coordenadas que llegan al contorno norte y sur del dominio computacional, será a través de  $P(\xi, \eta)$ . Control de espaciamento al contorno norte, correspondiente a la superficie de la esfera, no es realizado debido a que el mapeamento adoptado concentra puntos por defecto en esta superficie. Por lo tanto, las correcciones para obtener el espaciamento deseado y el ángulo de intersección deseado, para Ecs. (3), son expresas por

$$\Delta Q^n = + \tan^{-1} \left( \frac{\Delta s^n - \Delta s^*}{\Delta s^*} \right) \quad \Delta P^n = + \tan^{-1} \left( \frac{\alpha^n - \alpha^*}{\alpha^*} \right) \quad (4)$$

donde  $\Delta s^*$  y  $\alpha^*$ , son el espaciamento, y el ángulo de intersección deseado, respectivamente. Entretanto  $\Delta s^n$  y  $\alpha^n$  son los valores reales de estas variables, las cuales son calculadas a través de la formula de distancia entre puntos ( $\Delta s^n$ ), y del producto escalar de vectores a través de la función arco coseno ( $\alpha^n$ ) (Hofmann, 1992). Luego de obtener los valores de  $P(\xi, \eta)$  y  $Q(\xi, \eta)$  en los contornos, los valores en el dominio deben ser calculados. Hilgenstock (Hofmann, 1992) propone funciones de extrapolación para obtener los valores de  $P(\xi, \eta)$  y  $Q(\xi, \eta)$  para los puntos internos del dominio expresadas por

$$P(\xi, \eta) = P(\xi, 1) \exp[-a(\eta-1)/(\eta_{\max} - 1)] + P(\xi, \eta_{\max}) \exp[-b(\eta-1)/(\eta_{\max} - 1)] \quad (5)$$

$$Q(\xi, \eta) = Q(\xi, 1) \exp[-c(\eta-1)/(\eta_{\max} - 1)] + Q(\xi, \eta_{\max}) \exp[-d(\eta-1)/(\eta_{\max} - 1)] \quad (6)$$

donde  $a$ ,  $b$ ,  $c$  y  $d$  son los coeficientes de decaimiento para los valores de  $P(\xi, \eta)$  y  $Q(\xi, \eta)$ , en los contornos, que tienen por objetivo controlar la distribución de los puntos en el interior del dominio. Un coeficiente de decaimiento variable, expresado por Ec. (7), es aplicado para controlar el espaciamento de los puntos internos de la malla

$$c(\eta) = c_1, \quad 1 \leq \eta < \eta_1, \quad c(\eta) = \frac{f c_1}{\sqrt{(f \eta) - \eta_1}}, \quad \eta_1 \leq \eta < \eta_2, \quad c(\eta) = \frac{f c_1}{\sqrt{(f \eta_2) - \eta_1}}, \quad \eta \geq \eta_2 \quad (7)$$

donde  $c_1$  es o valor del coeficiente de decaimiento en el contorno sur, y tiene el valor máximo. Luego  $f$  es un factor de ponderación, y  $\eta_1$  y  $\eta_2$  son las coordenadas donde se aplica la variación del coeficiente. En la sección de resultados será observado el efecto de esta aproximación.

## 2.2. Solución del Ecurrimiento Reactivo

La solución de las variables de flujo es realizada por medio de la formulación Vorticidad-Función Corriente, a fin de evitar el tratamiento numérico de la presión. La Figura 4 presenta la geometría de la cámara de combustión a simular, y las entradas y la salida de masa, a partir de las cuales se definen las condiciones de contorno del problema. El perfil parabólico mostrado en la Fig. 4, en la sección de salida, indica la aplicación de la condición de contorno de flujo difusivo nulo de las variables en la dirección axial. La ecuación de Vorticidad, Ec. (8) y la ecuación de Poisson para función corriente, Ec. (9) son presentadas en la forma adimensional en coordenadas cilíndricas axisimétricas. La vorticidad y la función corriente están relacionadas con el vector de velocidad de acuerdo a las ecuaciones expresadas en la Ec. (10). El procedimiento de obtención de estas ecuaciones, junto con la transformación de las condiciones de

contorno, puede ser visto en (Anderson, 1997; Roache. 1982, y, Pacheco, 2005). La solución del fenómeno reactivo es realizada a través de la formulación de Shvab-Zel'dovich. Esta formulación considera la solución de las variables que definen el fenómeno de combustión a través de una única variable escalar, denominada potencial de acoplamiento, o escalar conservado,  $\phi_{sz}$ , considerando la hipótesis de llama fina y de número de Lewis unitario. La solución de este escalar, Ec. (11), permite la obtención de las fracciones máxicas de combustible, oxidante, y del campo de temperatura, a través de relaciones lineales donde estas variables están acopladas (Pacheco, 2005, y Kuo,1986). La condición de contorno para la corriente de oxidante es  $\phi_{sz} = 0$ , y para la corriente de combustible es  $\phi_{sz} = 1$ . Para los contornos sólidos, la condición de impermeabilidad es aplicada.

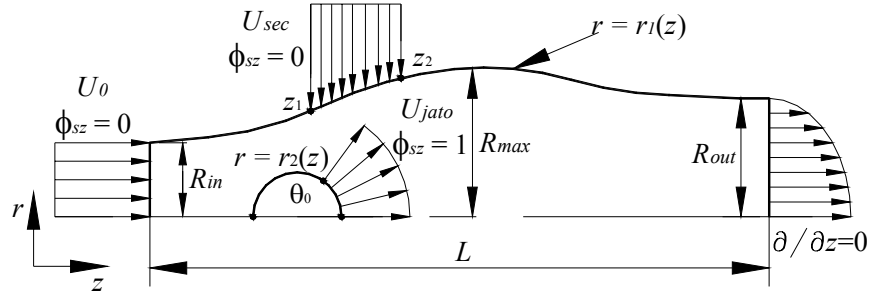


Figura 4. Parámetros del Modelo

$$\frac{\partial \omega_o}{\partial \tau} + u_r \frac{\partial \omega_o}{\partial r} + u_z \frac{\partial \omega_o}{\partial z} - u_r \frac{\omega_o}{r} = \frac{1}{Re} \left[ \frac{\partial^2 \omega_o}{\partial r^2} + \frac{1}{r} \frac{\partial \omega_o}{\partial r} - \frac{\omega_o}{r^2} + \frac{\partial^2 \omega_o}{\partial z^2} \right] \quad (8)$$

$$\omega_o = \frac{1}{r} \frac{\partial^2 \psi}{\partial r^2} - \frac{1}{r^2} \frac{\partial \psi}{\partial r} + \frac{1}{r} \frac{\partial^2 \psi}{\partial z^2} \quad (9)$$

$$\omega_o = \frac{\partial u_r}{\partial z} - \frac{\partial u_z}{\partial r}; \quad u_r = \frac{1}{r} \frac{\partial \psi}{\partial z}; \quad u_z = -\frac{1}{r} \frac{\partial \psi}{\partial r} \quad (10)$$

$$\frac{\partial \phi_{sz}}{\partial \tau} + u_r \frac{\partial \phi_{sz}}{\partial r} + u_z \frac{\partial \phi_{sz}}{\partial z} = \frac{1}{Pe} \left[ \frac{\partial^2 \phi_{sz}}{\partial r^2} + \frac{1}{r} \frac{\partial \phi_{sz}}{\partial r} + \frac{\partial^2 \phi_{sz}}{\partial z^2} \right] \quad (11)$$

$$Re = \frac{\rho U_0 R_{max}}{\mu}; \quad Pe = Re Pr = \frac{U_0 R_{max}}{\alpha} \quad (12)$$

El número de Prandtl para la mezcla es fijado a 0,7 (Lienhard, 2003). Como el número de Peclet es dependiente del número de Reynolds, Ec. (12), se tiene que el problema se torna dependiente únicamente del número de Reynolds. El método numérico empleado es el Método de Diferencias Finitas, con el uso de diferencias centradas de segunda orden para todas las derivadas espaciales, y Crank-Nicolson para la integración temporal.

### 3. Resultados

#### 3.1. Malla

La Figura 5 presenta una ligera modificación realizada en los cantos del dominio físico, puntos A, H, G, F, de la Fig. 3. En la Figura 5.(a) se observa que las líneas de la malla poseen derivadas discontinuas. Eso dificulta el cálculo natural de las variables, lo que implica un tratamiento especial en la línea coordenada que llega al canto. La solución adoptada consiste en aproximar ese canto por una curva, como mostrado en la Fig. 5.(b), de esta forma las derivadas son continuas, y permiten que la solución de las variables de interés sea normal. La Figura 6 presenta la calidad de los elementos de la malla en la superficie de la esfera y del cilindro. Las Figuras 7.(a)-(b) presentan el efecto de la distribución de puntos en la región posterior de la esfera, a través de las ecuaciones mostradas en (3)-(7). La Figura 7.(a) presenta la malla obtenida con la aplicación de un coeficiente de decaimiento constante y la Fig. 7.(b) presenta la malla obtenida a través de la aplicación de las relaciones mostradas en las Ecs. (7), donde se observa un cambio considerable en la distribución de los puntos internos en la región comprendida entre la esfera y la salida.

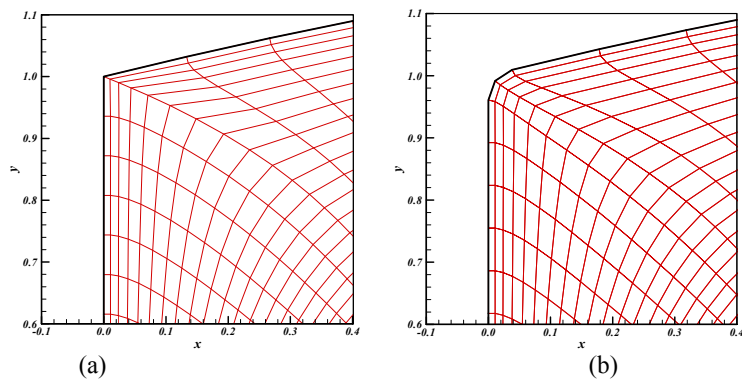


Figura 5: Aproximación de los cantos suaves (a) arista recta, (b) arista curva.

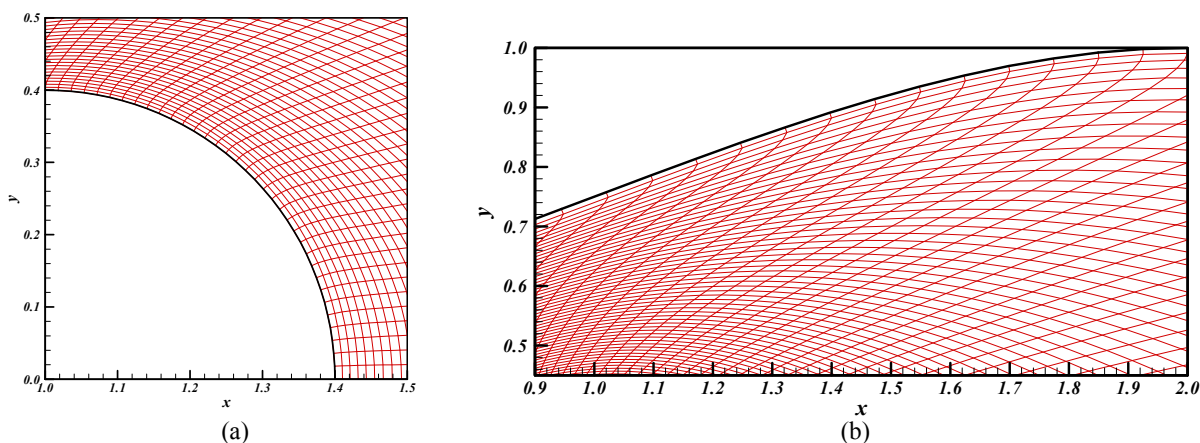


Figura 6. Detalle de la ortogonalidad de los elementos de malla en los contornos, y el grado de distorsión,  $r_{esf} = 0,4$ .

(a) Superficie de la esfera, (b) superficie del cilindro.

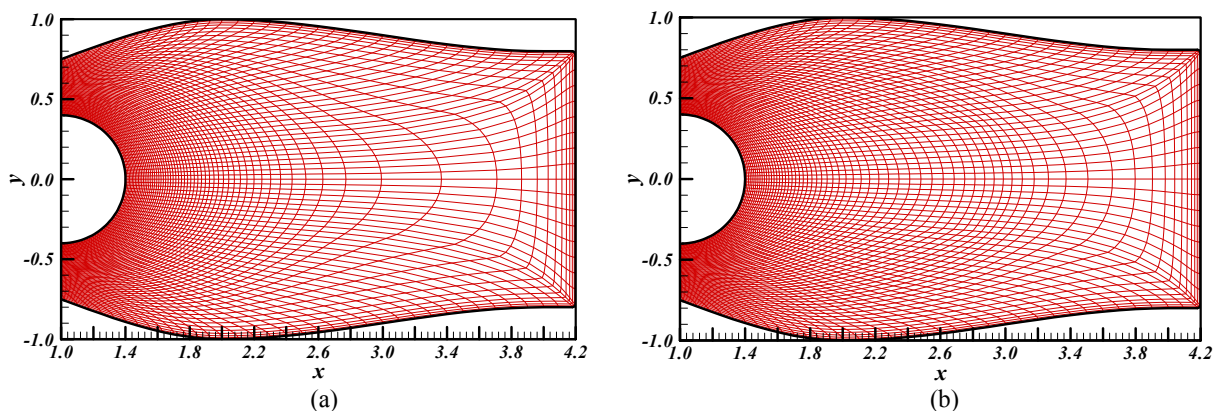


Figura 7. Malla de 141x66 nodos,  $r_{esf} = 0,4$ . (a) Coeficiente de decaimiento  $c$  constante,  $a = b = 25$ ,  $c = 35$

(b) Coeficiente de decaimiento variable,  $a = b = 25$ ,  $c_1 = 35$ ,  $f = 1,5$ ,  $\eta_1 = 0,3 \eta_{max}$ ,  $\eta_2 = 0,75 \eta_{max}$ ,  $\eta_{max} = 66$

### 3.2. ANÁLISIS DE CONVERGENCIA DE MALLA

La Tabla 1 presenta las condiciones de contorno fijadas, y los flujos másicos de operación expresados en términos del caudal volumétrico. El flujo másico sobre las cuales fueron realizadas las simulaciones numéricas, no guarda relación a las condiciones de operación de turbinas a gas. Esto puede ser observado en la Tab. 1, a través de la relación  $Q_p/Q_f$  y  $Q_T/Q_f$ , que es la relación de flujo másico de aire primario sobre combustible, y aire total sobre combustible, respectivamente. En condiciones reales, estas relaciones son del orden de 15 y 60 respectivamente.

El análisis de convergencia de malla de los resultados numéricos es realizado en forma gráfica a través de secciones representativas del dominio, debido a la irregularidad de la malla para evaluar los valores de las variables en puntos de referencia fijos. Las secciones analizadas fueron escogidas por ser las secciones donde se esperan los mayores gradientes. Los números de Reynolds simulados fueron  $Re = 1, 50, 100, 250, 500$  y  $750$  para  $r_{esf} = 0,2$  y  $0,4$ . El análisis de convergencia de malla fue realizado para tres mallas: una malla gruesa de (71x66) nodos, media de (106x86) nodos, y fina de (141x101) nodos.

**Tabla 1.** Condiciones de contorno.

$r_{esf}$	$U_0$	$Q_{pr}$	$U_{sec}$	$z_1$	$z_2$	$Q_{sec}$	$U_{jato}$	$\theta_0$	$Q_f$	$Q_p/Q_f$	$Q_T/Q_f$
0,2	1	0,393	1,35	1,25	1,55	1,131	2	1	0,1155	3,40	13,2
0,4	1	0,393	1,35	1,25	1,55	1,131	2	1	0,4621	0,85	3,30

El caso presentado a continuación resulto ser el más crítico. Esto es, el caso  $Re = 750$  para  $r_{esf} = 0,4$ . La Figura 9 presenta los perfiles de vorticidad y función corriente obtenida con las tres mallas en la sección  $z = 1$ . Se observa la concordancia cualitativa de los perfiles mostrando la evolución de una solución convergente para ambas variables a medida que la malla se refina. La diferencia relativa máxima observada en los resultados de vorticidad, para las mallas más refinadas es de 15% en  $r = 0,4$ , entretanto que la malla gruesa (71x66) con la malla media presentan diferencias de 100%. En  $r = 0,74$ , la diferencia relativa resulta prohibitiva de aplicar, pues los valores de vorticidad, para la malla fina (141x101), se encuentran próximos de cero. La diferencia absoluta es de 0.1 entre las mallas más refinadas, y de 3 para las mallas media (106x86) y gruesa (71x66). La función corriente presenta una diferencia relativa de 3% para las mallas mas refinadas, y de 50% para mallas gruesa y media en  $r = 0,55$ .

La Figura 10 presenta los resultados de las mallas simuladas para vorticidad y función corriente en la sección  $z = 1,4$ , donde se encuentra la entrada del aire secundario. Para vorticidad, se observa que para  $r > 0,6$  los resultados de las tres mallas presentan grandes desvíos. La concordancia cualitativa de los resultados de las mallas más refinadas se ve comprometida por el comportamiento anómalo en  $r = 0,8$ . Evaluando cuantitativamente la diferencia relativa, en los puntos donde se observa la mayor diferencia, se tiene que en  $r = 0,65$  las mallas más refinadas presentan un 8% de diferencia relativa, en cuanto que en  $r = 0,8$  la diferencia relativa es de 150%. Entretanto, los resultados de mallas gruesa y media presentan un 25% y 180% en  $r = 0,65$  y  $r = 0,8$ , respectivamente. Por otro lado, la significativa diferencia presentada en la vorticidad no afecta seriamente a los resultados de función corriente, presentando una diferencia relativa de 8% en  $r = 0,5$  para las mallas más refinadas. La significativa diferencia en la vorticidad es atribuida a la calidad de los elementos de la malla, presentando una elevada distorsión en sus elementos, como también a la discretización de los términos convectivos de la ecuación de vorticidad. El comportamiento anómalo observado en los resultados de vorticidad, es típico de soluciones con presencia de dispersión numérica, atribuida al hecho que el esquema de Diferencias Centradas fue aplicado a la discretización de los términos convectivos. Por lo tanto, se aplica un esquema de alta orden que considera la dirección del vector velocidad como el esquema QUICK con corrección atrasada de Hayase (Versteeg y Malalasekere, 1995), utilizando la malla fina, para luego comparar los resultados de este esquema QUICK con el resultado de Diferencias Centradas (DC) en esta malla. De esta forma se cuantifica el efecto de los esquemas de discretización en los términos convectivos. La Figura 11 muestra como la aplicación del esquema QUICK atenúa fuertemente las oscilaciones numéricas, además de confirmar que los resultados de la función corriente presentan una evidente concordancia para ambos esquemas de discretización. Por lo tanto, se confirma que los resultados de función corriente son satisfactoriamente concordantes, y como consecuencia, la obtención del vector de velocidad resulta satisfactoria.

Para la sección  $z = 2$  (Fig. 12), los resultados de vorticidad y función corriente obtenidos con las mallas simuladas presentan mejores comportamientos que aquellos observados para el caso mostrado en la Fig. 10. Por lo tanto, la intensidad de los gradientes en esta región son menores, pues los resultados de las mallas simuladas, para vorticidad y función corriente, en esta sección presentan concordancia cualitativa para las tres mallas simuladas, siendo que las que muestran los menores desvíos son los resultados de las mallas más refinadas. La mayor diferencia relativa de los resultados de vorticidad entre las mallas más refinadas es de 4% en  $r = 0,65$  y para la función corriente de 1% en  $r = 0,7$ .

En lo que se refiere al comportamiento de la variable escalar, la Figura 13 presenta el perfil del potencial de acoplamiento en  $z = 1,4$ ,  $z = 2$  y  $z = 4$ . En  $z = 1,4$  se observa el intenso gradiente de esta variable, debido a que la salida de combustible se ubica en esta posición. La concordancia cualitativa de las mallas refinadas indica que esta variable esta bien resuelta. La evaluación de la diferencia relativa en  $r = 0,4$  indica un 8% de diferencia relativa, debido al intenso gradiente observado. Entretanto, en  $z = 2$  los gradientes son menos intensos, y la solución de la variable escalar es convergente para las tres mallas, dentro de la precisión de los gráficos. Por último, en  $z = 4$  el

comportamiento de la malla más grosera presenta una oscilación numérica, con valores no físicos,  $\phi_{sz} < 0$ , en  $r = 0,46$ , atribuido al hecho de que la malla en la sección de salida del dominio físico tiene elementos de tamaño más groseros.

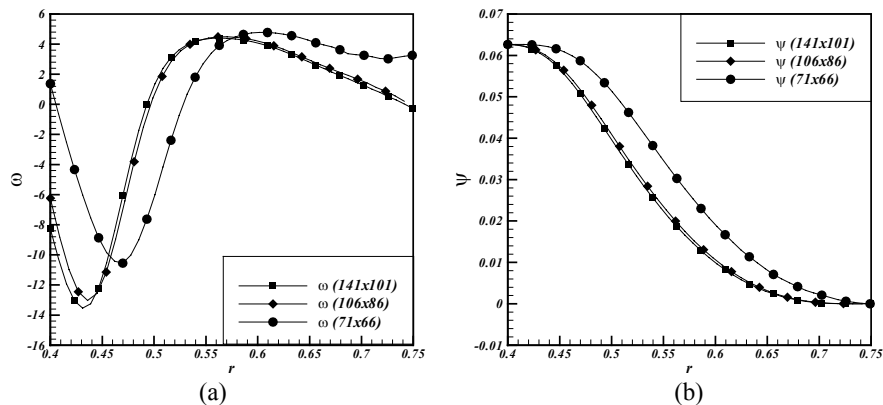


Figura 9. Vorticidad (a) y Función Corriente (b), en  $z = 1$ ,  $Re = 750$ ,  $r_{esf} = 0,4$ .

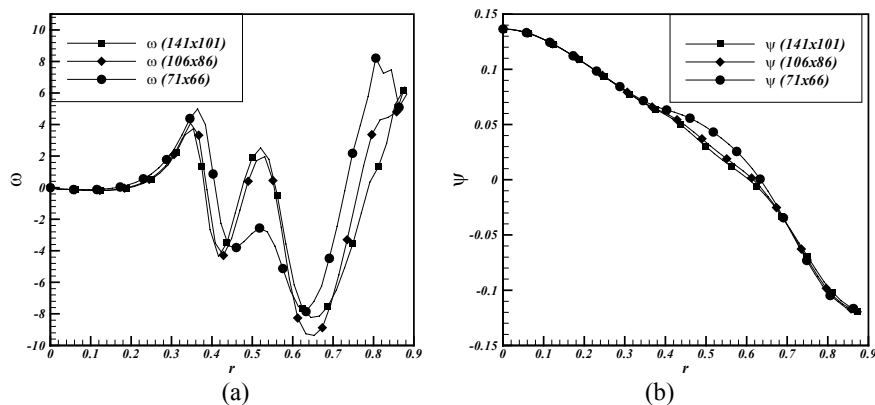


Figura 10. Vorticidad (a) y Función Corriente (b), en  $z = 1,4$ ,  $Re = 750$ ,  $r_{esf} = 0,4$ .

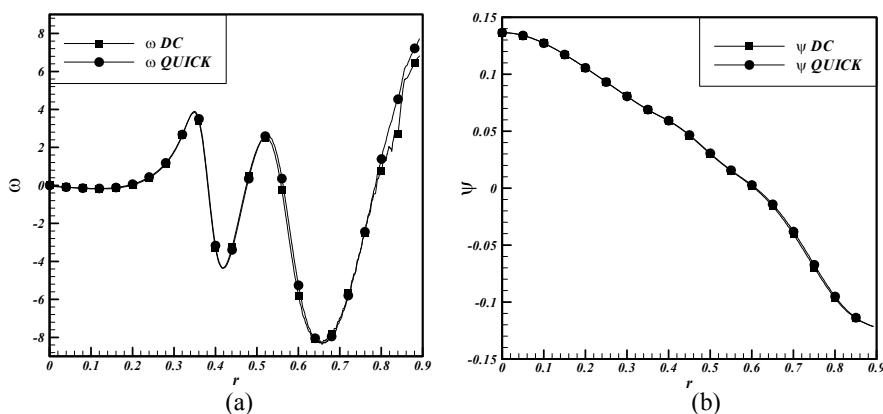


Figura 11. Vorticidad (a) y Función Corriente (b), Caso Figura 10, Malla Fina, QUICK-DC.

Finalmente, la Fig. 14 presenta un resultado global y local para  $r_{esf} = 0,4$ ,  $Re = 750$ , a fin de mostrar el potencial de la herramienta numérica desarrollada en el presente trabajo. El resultado global presenta la solución del Potencial de Acoplamiento,  $\phi_{sz}$ , Fig. 14.(a), en conjunto con los vectores de velocidad en secciones representativas del dominio, para mostrar la evolución del campo de velocidades a lo largo del dominio. Entretanto, la Fig. 14.(b) presenta las líneas de



corriente en la región de mayor interés, denominada de región primaria, que es la región donde interactúan las corrientes de aire primario, aire secundario y el chorro de combustible.

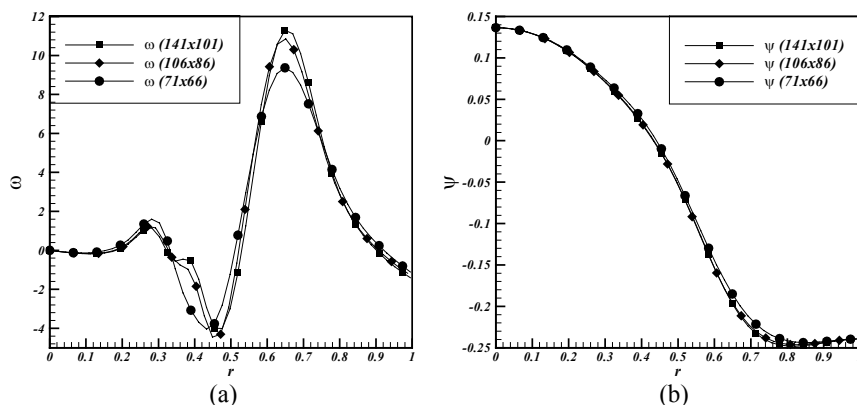


Figura 12. Vorticidad (a) y Función Corriente (b), en  $z = 2$ ,  $Re = 750$ ,  $r_{esf} = 0,4$ .

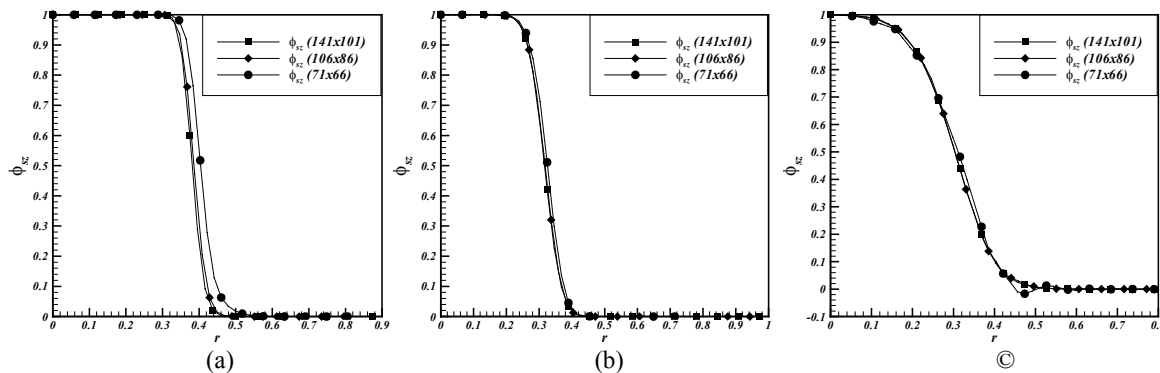


Figura 13. Escalar Conservado, en (a)  $z = 1,4$ , (b)  $z = 2$ , y (c)  $z = 4$ ,  $Re = 750$ ,  $r_{esf} = 0,4$ .

#### 4. Conclusiones

Fue realizado un estudio numérico del escurrimiento de fluido en una geometría axisimétrica con una superficie cilíndrica irregular, que procura una semejanza con las cámaras de combustión de turbinas a gas del tipo tubular, y que tiene como finalidad la simulación de este dispositivo. La primera aproximación consistió en resolver las ecuaciones de conservación de cantidad de movimiento, conservación de energía y especies, a través de la formulación Vorticidad-Función Corriente, y la formulación de Shvab-Zel'dovich. A fin de confirmar la potencialidad de la malla generada, un análisis de convergencia de malla fue realizado. Fueron simuladas tres mallas de (71x66) nodos, (106x86) nodos y (141x101) nodos, denominadas malla gruesa, media y fina. El análisis de convergencia de malla para los campos de función corriente y el potencial de acoplamiento presentaron soluciones satisfactorias para las mallas más refinadas, en cuanto que la malla gruesa mostró resultados no físicos para el escalar conservado en la región de salida. Entretanto, la vorticidad se comporta como la variable de convergencia más pobre, en particular para la sección  $z = 1,4$ . En el resto de las secciones analizadas los resultados de vorticidad obtenidos con las mallas más refinadas fueron satisfactorios, mostrando la tendencia de una solución convergente. Las diferencias en  $z = 1,4$ , son atribuidas a la calidad de los elementos de malla, presentando una elevada distorsión en sus elementos, y principalmente a la entrada del aire secundario, la cual impone un fuerte efecto convectivo, y como consecuencia, soluciones con dispersión numérica fueron encontradas. La aplicación del esquema QUICK atenúa fuertemente las oscilaciones numéricas observadas cuando diferencias centradas fueron aplicadas a los términos convectivos de la ecuación de vorticidad. Sin embargo, los resultados de función corriente, para ambos esquemas de discretización, utilizando la malla fina, presentan una solución satisfactoriamente convergente. De esta forma, la obtención del campo de velocidades a través del campo de función corriente es bien sucedida, y esto es confirmado con la solución satisfactoriamente convergente, dentro de la precisión de los gráficos, de la variable escalar que define el fenómeno reactivo.

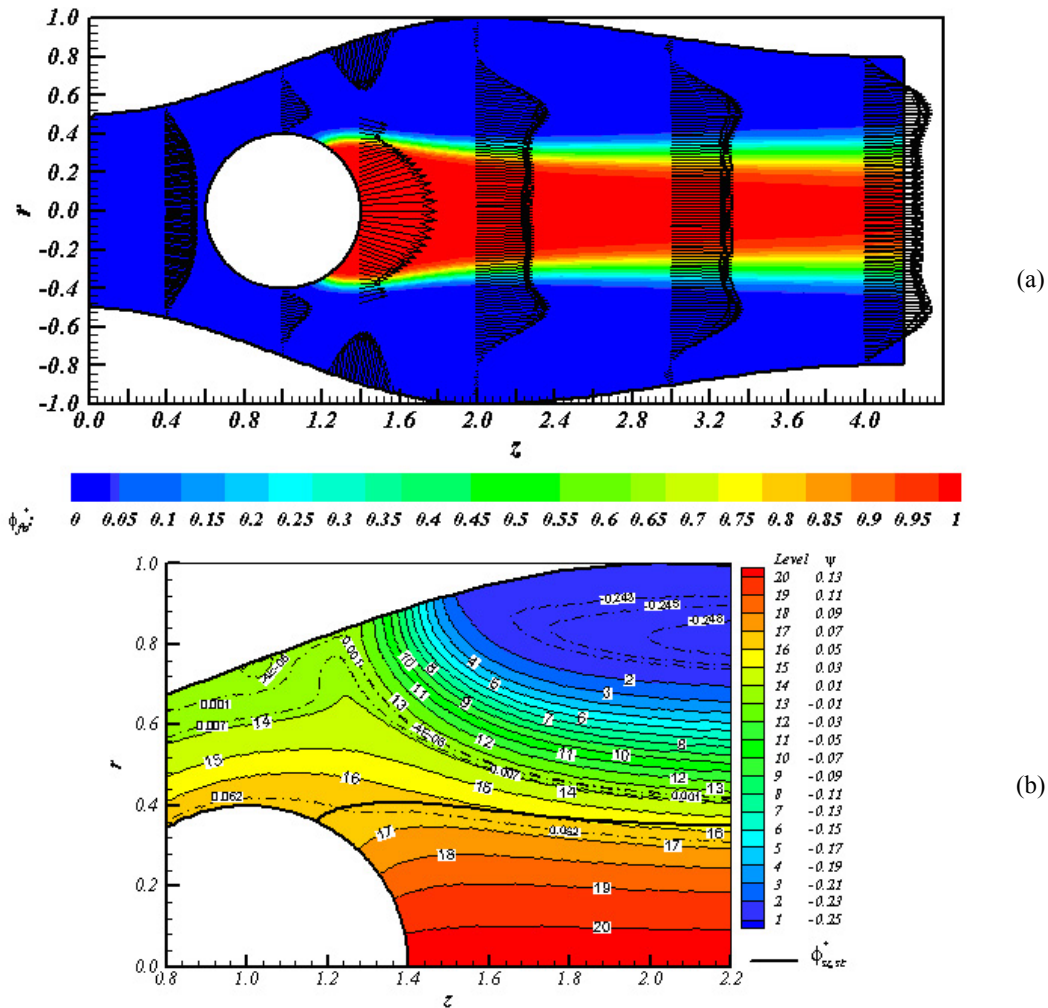


Figura 14: (a) Potencial de Acoplamiento y (b) Líneas de Corriente,  $Re = 750$ ,  $r_{esf} = 0,4$ .

### 5. Agradecimientos

Los autores agradecen a CAPES por el apoyo financiero proporcionado para el desarrollo del presente trabajo.

Los recursos computacionales fueron facilitados por el Laboratorio de Máquinas Térmicas LMT de la Universidad Federal de Rio de Janeiro UFRJ.

### 6. Referencias

Anderson, Dale A., Tannehill, John C., Pletcher. Richard H. "Computational Fluid Mechanics and Heat Transfer". Taylor & Francis Group 2º Edición, 1997.

Hofmann Joe D. "Numerical Method for Engineers and Scientists". McGraw-Hill inc. 1992.

Jurgen J.J. Louis, Jim B. W. Kok, y Sikke A. Klein. "Modelling and Measurements of a 16-kW Turbulent Nonadiabatic Diffusion Flame in a Cooled Cylindrical Combustion Chamber", Combustion and Flame vol.125, pp. 1012-1031, 2001.

Kuo Kenneth k. "Principles of Combustion". John Wiley & Sons Inc., 1º Edición, 1986.

Lienhard IV, John H., Lienhard V, John H. "A Heat Transfer TextBook". Cambridge MA: Phlogiston Press, 3º Edition 2003.

Mare F. di, Jones W.P., y Menzies K. R. "Large Eddy simulation of a model gas turbine combustor", Combustion and Flame vol.137, pp. 278-294, 2004.

Menom S., Sankaran V., y Stone C. "Dynamics of Swirling Premixed and Spray Flames", AIAA 2001-1092, 2001.

Özisik, M. Necati. "Finite Difference Methods in Heat Transfer". Editora CRC Press, Boca Raton, 1º Edición 1994.

- Pacheco F, Pedro A. “Escoamentos Reativos em Geometrias-Modelo de Câmaras de Combustão de Turbinas a Gás”, Tese M. Sc. PEM/COPPE/UFRJ, 2005.
- Roache, Patrick J. “Computational Fluid Dynamics”. Hermosa Publisher, 1982.
- Saravanamutto H. I. H, Rogers G. F. C., Cohen H. “Gas Turbine Theory”. Pearson Education Limited, 5° Edición, 2001.
- Selle L., Lartigue G., Poinso T., Koch R., K.-U.Schildmacher, Krebs W., Prade B., Kaufmann P., y Veynante D. “Compressible large eddy simulation of turbulent combustion in complex geometry on unstructured meshes”, Combustion and Flame vol.137, pp. 489-505, 2004.
- Versteeg H. K., Malalasekera W. “An introduction to Computational Fluid Dynamics The Finite Volume Method”. Longman Group Ltd, 1995.

## NUMERICAL GRID GENERATION FOR THE ANALYSIS OF REACTIVES FLOW IN COMBUSTOR

### **Pedro Andrés Pacheco Fernandoy**

Department of Mechanical Engineering–COPPE, Universidade Federal do Rio de Janeiro, P.O: Box 68503 – Rio de Janeiro CEP: 21975-970, Brazil.  
e-mail: pedrop@lmt.coppe.ufrj.br

### **Albino José Kalab Leiroz**

Department of Mechanical Engineering–COPPE, Universidade Federal do Rio de Janeiro, P.O: Box 68503 – Rio de Janeiro CEP: 21975-970, Brazil  
e-mail: leiroz@ufrj.br

**Abstract.** *The increasing interest recently observed in the use of gas turbines has motivated the numerical study of its principal components in order to obtain a better understanding of physical phenomena involved. In the present work, numerical grid generation techniques are applied in axisymmetrical geometries similar to can-type combustor of gas turbine. A spherical bluff-body is positioned to deliver of axis symmetry of combustor, of way to allow the studying effect of the fuel injector. Effects of secondary air injection are considered also. For this irregular geometry, techniques of numerical grid generation and coordinate transformation are applied for the mapping of physical dominion and the solution the equations of interest variables. For the present study, the flow field was considered incompressible, laminar, and with constant thermophysical properties. The flow field was obtained through the Stream Function-Vorticity formulation. The solution of the conservation equations for energy, and chemical species is accomplished through of Shvab-Zel'dovich potentials. Result of model allows the determination the position and shape of flame, and calibration of parameters of grid generated.*

**Keywords:** *Gas Turbine Combustor, Grid Generation, Stream Function-Vorticity, Shvab-Zel'dovich.*

## A NEW MESHFREE APPROACH FOR SIMULATION OF FLUID FLOW

**Fernanda Olegario dos Santos**

feroleg@icmc.usp.br

**Antonio Castelo Filho**

castelo@icmc.usp.br

**Luis Gustavo Nonato**

gnonato@icmc.usp.br

Instituto de Ciências Matemáticas e de Computação  
Departamento de Matemática Aplicada e Estatística,  
Universidade de São Paulo,  
Av. Trabalhador São-Carlense 400, Cxp 668, 13560-970, São Carlos-SP, Brasil

**Abstract.** Meshfree fluid flow simulation has achieved large popularity in the last years. Meshfree Galerkin Methods and Smooth Particle Hydrodynamics are typical examples of meshfree techniques, whose ability to handle complex problems has motivated the CFD community. In this work we present a new meshfree approach that uses moving least square (MLS) to discretize the model equations. A mesh is only employed to manage the neighborhood relationship of points spread within the domain, avoiding thus the problem of keeping a good quality mesh. This strategy was implemented in the integrated simulation system, called UmFlow-2D, whose purpose is to simulate incompressible fluid flow on unstructured mesh. The system is divided in three modules: modeling, simulation and visualization modules. The simulation module implements the Navier-Stokes equation, which are discretized by a Generalized Finite Difference Method. In particular convective terms are discretized by a semi-lagrangian technique. A projection method is employed to uncouple the velocity components and pressure. Results of numerical simulations proving the effectiveness of our approach in two-dimensional fluid flow simulations are presented and discussed.

**keywords:** Numerical simulation. Generalized finite difference method. Unstructured mesh. Meshfree Discretization.

### 1. Introduction

The need for new techniques for the solution of problems where the classical numerical methods fail or are prohibitively expensive has motivated the development of new approaches, such as meshfree methods. Aiming at avoiding difficulties as the generation of good quality meshes and mesh distortions in large deformation problems, the meshfree methods try to construct approximation functions in terms of a set of nodes.

The literature has presented a set of different meshfree methods, according to computational modeling, the meshfree methods may be put into two different classes, [Li and Liu, 2004] those that approximate the strong form of a partial differential equation (PDE), this techniques, in general, discretize the PDE by a collocation technique. Examples of such methods are generalized finite difference method (GFDM) [Liszka and Orkisz, 1980] and smoothed particle hydrodynamics (SPH) [Monaghan, 1988]. The methods in the second class, i.e., serving as approximations of the weak form of a PDE, are often Galerkin weak formulations (meshfree Galerkin methods). Examples of such an approach are element-free Galerkin method (EFGM) [Belytschko, 1994], diffuse element method (DEM)[Nayroles et al., 1992], reproducing kernel particle methods (RKPM)[Liu et al., 1995], and partition of unit method (PUM) [Melenk and Babuska, 1996].

In this work we present a new meshfree method that approximates the strong form of a PDE. Our approach estimates the derivatives involved in a PDE from a polynomial approximation conducted in each discretized node. Different from GFDM methods, which use the classical Taylor series expansion to calculate the polynomial from which the derivatives are extracted, our strategy adopts a more flexible scheme to compute the polynomial approximation, namely the moving least square (MLS)[Levin, 1998]. The moving least square presents some advantages over Taylor series expansion. For example, the weight assignment, usually employed to control the

contribution of neighbor nodes to the polynomial approximation, can be accomplished in a more straight way by MLS. Furthermore, MLS can be combined with partition of unity in order to tackling the problem of the number of neighbor nodes properly. In order to show the effectiveness of the proposed technique, we present an integrated simulation system, called UmFlow-2D, wich aims at simulating two-dimensional incompressible fluid flow using unstructured mesh. The system is divided in three modules: modeling, simulation and visualization modules. The modeler module is used to define a fluid, boundary conditions, and others simulation's datas. The simulation module implements the Navier-Stokes equation and boundary bonditions. The visualization module is a system that permit the visualization of result and implements several visualization method.

In this paper we presents the simulation module. This module implements the Navier-Stokes equation in a computacional code. The governing equations are discretized by a generalized finite difference method with the convective terms discretized by a semi-Lagrangian scheme. A projection method is employed to uncouple the velocity componentes and pressure. The strategy employed to solve the Poisson's equation generated from our discretization strategy is another novelty of this work. The details of such a modeling is also presented.

The work is organized as follows: Section 2 presents the least square discretization method proposed in this work. A description of how to employ such a discretization method in Navie-Stokes equations is discussed in section 3. The scheme adopted to define the boundary conditions is presented in section 3.1. Section 4 presents some results obtained from the proposed approach. Conclusions and future work are in section 5.

## 2. Least Square Approximation

In this section we present some basic definitions and notation employed in the remaining of the text and the Least Square Approximation.

### 2.1. Computational Node Arrangement

Let  $V = \{v_1, v_2, \dots, v_n\}$  be a set of discrete nodes representing a domain  $D \subset \mathbb{R}^2$ . For each node  $v_i \in V$  we define the *local coordinate system of  $v_i$*  by writing any point  $\mathbf{r} = (x, y) \in D$  as  $\bar{\mathbf{r}}_i = \mathbf{r} - \mathbf{r}_i$ , where  $\mathbf{r}_i = (x_i, y_i)$  are the coordinates of  $v_i$ . We denote by  $\bar{\mathbf{r}}_{k,i} = \mathbf{r}_k - \mathbf{r}_i$  the coordinates of a node  $v_k \in V$  written in the local coordinate system of  $v_i$ .

Let  $S_i \subset V$  be a non-empty subset of nodes,  $S_i$  is a *star of  $v_i$* , if the two conditions bellow are satisfied:

1. if  $\|\bar{\mathbf{r}}_{s,i}\| \leq \|\bar{\mathbf{r}}_{k,i}\|, \forall v_k \in V, k \neq s$  then  $v_s \in S_i$
2. if  $v_s$  is in the convex hull of  $S$  then  $v_s \in S_i$

The *local minimum length* of a star  $S_i$  is defined as:

$$h_i = \min_{v_s \in S_i} \|\bar{\mathbf{r}}_{s,i}\| \quad (1)$$

Notice that the local minimum length is the same for all stars of  $v_i$ . From the definition of local minimum length we can define the *global minimum length* with respect to  $V$ :

$$h = \min_{v_i \in V} h_i \quad (2)$$

in another words, the global minimum length  $h$  is the shortest distance of the nodes representing  $D$ .

### 2.2. Least Square Approximation

Let  $v_i \in V$  be a node in the domain  $D$  and  $S_i$  be a star of  $v_i$ . Suppose that  $f : D \rightarrow \mathbb{R}$  is a real function defined in  $D$ . We aim at approximating  $f$  in a neighborhood of  $v_i$  by a function  $\bar{f}$  of the form:

$$\bar{f}_i(\bar{\mathbf{r}}) = f(\mathbf{r}_i) + W_i(\bar{\mathbf{r}}) \quad (3)$$

where  $W_i$  is a polynomial of degree  $d$  that can be written as:

$$W_i(\bar{\mathbf{r}}) = \sum_{j=1}^N c_j P^{(j)}. \quad (4)$$

The terms  $P^{(j)}$  in Eq. (4) forms a basis of monomial  $\{x, y, x^2, xy, y^2, \dots\}$ , which can be numbered as in Fig. (1). Notice that the constant monomial is not considered, as the polynomial will be employed to approximate derivatives, thus the constant term can be neglected.

Monomial Basis	
monomial	Degree
$P^{(1)} = x$	1
$P^{(2)} = y$	1
$P^{(3)} = x^2$	2
$P^{(4)} = xy$	2
$P^{(5)} = y^2$	2
$P^{(6)} = x^3$	3
$P^{(7)} = x^2y$	3
$P^{(8)} = xy^2$	3
$P^{(9)} = y^3$	3

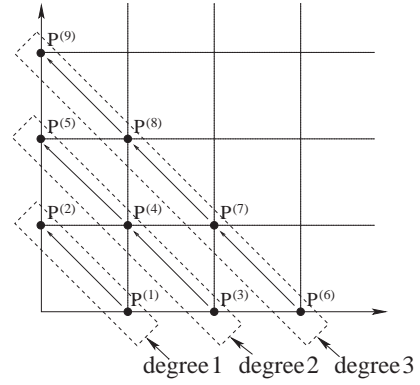


Figure 1: Monomial basis and numbering scheme.

Given the values of  $f$  in each node  $v_k \in S_i$ , we can compute the coefficients  $c_j$  of  $W_i$  by solving the linear system  $Ac = B$ :

$$\begin{bmatrix} a_{11} & \cdots & a_{1N} \\ \vdots & & \vdots \\ a_{N1} & \cdots & a_{NN} \end{bmatrix} \begin{bmatrix} c_1 \\ \vdots \\ c_N \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_N \end{bmatrix} \quad (5)$$

where the elements  $a_{ij}$  of the matrix  $A$  and the elements  $b_i$  of vector  $b$  are given by:

$$a_{ij} = \sum_{v_k \in S_i} P^{(i)}(\bar{\mathbf{r}}_k) P^{(j)}(\bar{\mathbf{r}}_k) w_k; \quad i, j = 1, \dots, N \quad (6)$$

$$b_i = \sum_{v_k \in S_i} (f(\mathbf{r}_k) - f(\mathbf{r}_i)) P^{(i)}(\bar{\mathbf{r}}_k) w_k \quad (7)$$

As can be seen from Eq. (6) and (7), using the MLS we are assigning weights  $w_k$  for the node  $v_k \in S_i$ . Such weights can depend on the distance between  $v_k$  and  $v_i$  or they can be a Gaussian in  $v_i$ . It is important to point out that the rank of  $A$  depends on the number of elements in  $S_i$ . For example, for a quadratic polynomial approximation there will be needed at least five nodes in  $S_i$ . The higher the degree of  $W_i$  the more nodes are needed.

Once the coefficients  $c_j$  have been computed, the derivatives of  $f$  can be approximated in  $v_i$  by the derivatives of  $\bar{f}_i$ . Furthermore, if  $\bar{f}_i$  is a quadratic polynomial then the second order derivatives are given directly from the coefficients  $c_j$ , i.e.,

$$\begin{aligned} \frac{\partial^2 \bar{f}_i}{\partial \bar{x}^2} &= \frac{\partial^2 W_i}{\partial \bar{x}^2} = 2c_3 \\ \frac{\partial^2 \bar{f}_i}{\partial \bar{x} \partial \bar{y}} &= \frac{\partial^2 W_i}{\partial \bar{x} \partial \bar{y}} = c_4 \\ \frac{\partial^2 \bar{f}_i}{\partial \bar{y}^2} &= \frac{\partial^2 W_i}{\partial \bar{y}^2} = 2c_5 \end{aligned} \quad (8)$$

It can be shown that the discretization strategy presented above is consistent if the nodes in  $S_i$  are distributed properly. Details about this theoretical result can be found in Peña's master dissertation [Peña, ]. In order to verify the effectiveness of the scheme above in numerical simulations, we apply the proposed strategy in an incompressible fluid flow simulation problem. How to conduct the discretization of the Navier-Stokes equations from our approach is the subject of the next section.

### 3. Discretizing Navier-Stokes Equations

Although the discretization technique presented in the last section has been developed for meshfree domain decompositions, we prefer using a mesh to make the access to the neighborhood of a node easier. To this end, the set of nodes representing a domain  $D$  has been input in a Delaunay mesh generator. It is not difficult to show that Delaunay meshes guarantee the first condition of the definition of a star. Without any post-processing a Delaunay mesh satisfy the second condition in almost every node. Steiner points can be inserted if it is strongly necessary to respect condition 2 of the definition of a star.

Pressure discretization will also be making use of the mesh, as we are storing the pressure on the triangular cells. It is worth mentioning that the velocity field is stored on the nodes as in Fig (2). Such a scheme has been adopted in order to make velocity and pressure decoupling easier.

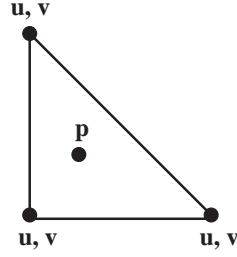


Figure 2: Computational cell.

Consider the Navier-Stokes and continuity equations:

$$\frac{D\mathbf{u}}{Dt} = -\nabla p + \frac{1}{Re}\nabla^2\mathbf{u} + \frac{1}{Fr^2}\mathbf{g}, \quad (9)$$

$$\nabla \cdot \mathbf{u} = 0. \quad (10)$$

where  $\mathbf{u}$  is the velocity and  $p$  is the pressure of the fluid,  $Re$  is the Reynolds number and  $Fr$  is the Froude number.

The material derivative  $\frac{D\mathbf{u}}{Dt}$  is discretized by the semi-Lagrangian method:

$$\frac{D\mathbf{u}}{Dt} = \frac{\mathbf{u}(\mathbf{x}, t + \delta t) - \mathbf{u}(\mathbf{x} - \delta\mathbf{x}, t)}{\delta t}. \quad (11)$$

where  $\delta\mathbf{x} = \mathbf{u}\delta t$ .

Using the fractionary step method (projection method), we obtain the set of equations:

$$\frac{\tilde{\mathbf{u}}(\mathbf{x}, t + \delta t) - \mathbf{u}(\mathbf{x} - \delta\mathbf{x}, t)}{\delta t} = \frac{1}{Re}\nabla^2\mathbf{u} + \frac{1}{Fr^2}\mathbf{g}, \quad (12)$$

$$\frac{\mathbf{u}(\mathbf{x}, t + \delta t) - \tilde{\mathbf{u}}(\mathbf{x}, t + \delta t)}{\delta t} = -\nabla p^{n+1}, \quad (13)$$

$$\nabla^2 p^{n+1} = \frac{1}{\delta t}\nabla \cdot \tilde{\mathbf{u}}(\mathbf{x}, t + \delta t), \quad (14)$$

From the above equations, the velocity and pressure fields can be computed, for each time step, as follows:

1. Intermediate velocity

$$\tilde{\mathbf{u}} = \mathbf{u}(\mathbf{x} - \delta\mathbf{x}, t) + \delta t \left( \frac{1}{Re}\nabla^2\mathbf{u} + \frac{1}{Fr^2}\mathbf{g} \right) \quad (15)$$

2. Intermediate pressure

$$\nabla^2 p^{n+1} = \frac{1}{\delta t}\nabla \cdot \tilde{\mathbf{u}}, \quad (16)$$

using the following boundary conditions

- Homogeneous Neumann condition for rigid contours, given by

$$\frac{\partial p^{n+1}}{\partial n} = 0.$$

This condition is used also on the inflows.

- Homogeneous Dirichlet condition for outflow, i.e.,  $p^{n+1} = 0$ .

### 3. New velocity

$$\mathbf{u}^{n+1} = \tilde{\mathbf{u}} - \delta t \nabla p \quad (17)$$

The term  $\mathbf{u}(\mathbf{x} - \delta \mathbf{x}, t)$  in Eq. (15) is computed by linear interpolation of the velocity  $\mathbf{u}$  on the nodes  $v_i, v_j$  and  $v_k$  closest to  $\mathbf{x} - \delta \mathbf{x}$ . The Laplacian term  $\nabla^2 \mathbf{u}$  is computed from a least square approximation as described in (9).

After estimating  $\tilde{\mathbf{u}}$ , we must solve Poisson's equation (16). In fact, this is the hardest step of the scheme. Using a quadratic polynomial for the least square approximation, a  $5 \times 5$  linear system is obtained:

$$\begin{bmatrix} a_{11} & \cdots & a_{15} \\ \vdots & & \\ a_{51} & \cdots & a_{55} \end{bmatrix} \begin{bmatrix} c_1 \\ \vdots \\ c_5 \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_5 \end{bmatrix} \quad (18)$$

where the elements  $a_{ij}$  and  $b_i$  are given by Eq. (6) and Eq. (7) respectively.

Using Gaussian elimination we can re-write the system (18) as:

$$\begin{bmatrix} \hat{a}_{11} & \cdots & \hat{a}_{15} \\ & \ddots & \\ & & \hat{a}_{55} \end{bmatrix} \begin{bmatrix} c_1 \\ \vdots \\ c_5 \end{bmatrix} = \begin{bmatrix} \hat{b}_1 \\ \vdots \\ \hat{b}_5 \end{bmatrix}. \quad (19)$$

By backward substitution one can obtain the coefficients  $c_5$  and  $c_3$  that are involved in the discretization of  $\nabla^2 p$ , and they can be written as:

$$c_3 = \sum_{v_k \in S_i} \alpha_k p(r_k) + \alpha_i p(r_i) \quad (20)$$

$$c_5 = \sum_{v_k \in S_i} \beta_k p(r_k) + \beta_i p(r_i) \quad (21)$$

where  $\alpha_k$  and  $\beta_k$  are constants obtained from the Gaussian elimination process.

In that way, the Poisson matrix is sparse and non-symmetric. In our implementation we employ the bi-conjugate gradient method [Saad, 2003] to solve the resulting linear system.

Once  $p$  has been calculated, moving least square can be employed to approximate  $\nabla p$ , making it possible to solve Eq. (17).

#### 3.1. Boundary Conditions

The boundary conditions employed in our discretization scheme have been discussed. In fact, we must handle three different types of boundary: rigid contours, inflow and outflow.

For rigid contours two different boundary conditions have been implemented in our code: no slip and free slip. In the first case the velocity is set to zero in all nodes defining the rigid contours. The free slip condition imposes that the velocity in the normal direction be zero and the derivative of the tangential velocity with respect to the normal direction is also zero.

On the inflows, the velocity is given in the normal direction, being zero in the tangential direction.

On the outflows, the derivative of the normal component of the velocity with respect to the normal direction is zero and the derivative of the tangential velocity with respect to the normal direction is also zero.

The boundary conditions for pressure is introduced for each triangular element at contour an "imaginary" triangular element, increasing a neighborhood in the cell of approach of the derivatives as in Fig. (3).

- Homogeneous Neumann condition for rigid contours, given by

$$\frac{p^{(n+1)}(r_f) - p^{(n+1)}(r_0)}{\text{dist}(r_f \rightarrow r_0)} = 0.$$

The value of  $p$  in the triangular element imaginary is equal to the value of  $p$  in the triangular element of contour.

$$b_i = \sum_{v_k \in C_0} (p^{(n+1)}(\mathbf{r}_k) - p^{(n+1)}(\mathbf{r}_0)) P_k^{(j)} \omega_k \quad (22)$$



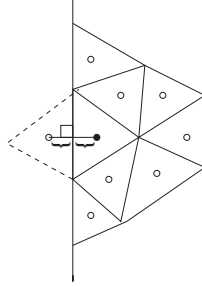


Figure 3: Boundary condition for pressure.

The independent term  $b_i$  is not modified.

- Homogeneous Dirichlet condition for outflow, i.e.,

$$p = 0.$$

In this case the value of  $p$  in the element imaginary is

$$p(r_f) = -p(r_i).$$

and the independent term is modified

$$b_i = \sum_{v_k \in C_0} (p^{(n+1)}(\mathbf{r}_k) - p^{(n+1)}(\mathbf{r}_0)) P_k^{(j)} \omega_k$$

Getting  $2p_0 P_f^i$  where  $P_f^i$  is the monomial  $i$  of the triangular element imaginary.

## 4. Results

In order to illustrate the effectiveness of our discretization technique, we present two examples of simulations. The first example shows the classical fluid flow simulation in a channel. The second example aims at illustrating the behavior of our approach in a flow over a circular cylinder.

### 4.1. Flow in a Channel

The well known *Hagen-Poiseuille* flow has been chosen to validate our numerical method, as an analytical solution is available. This simulation consists of a flow between two parallel plates, as illustrated in Fig. (4).

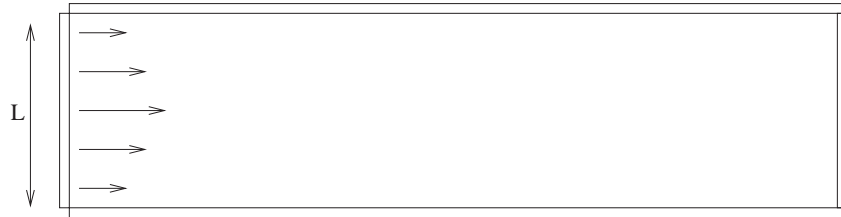


Figure 4: *Hagen-Poiseuille* flow.

The analytical solution for Hagen-Poiseuille flow, which can be found in Batchelor [Batchelor, 1970], is given by:

$$u(y) = -\frac{1}{2\mu} \frac{\partial p}{\partial x} (yL - y^2), \quad (23)$$

where  $\mu$  is the viscosity and the velocity  $u$  is a function of the distance  $y$  to the wall. Considering  $L$  to be the width of the channel, the pressure gradient can be written as:

$$\frac{\partial p}{\partial x} = -12 \frac{\mu Q}{L^3}, \quad (24)$$

where  $Q$  is defined by:

$$Q = \int_0^L u(y) dy. \tag{25}$$

Considering  $u(y) = U$  on the inflow, where  $U$  is the reference velocity, and choosing  $L = U = 1$ , the analytical solution is:

$$u(y) = -6y(y - 1), \tag{26}$$

Three different meshes have been employed to show the convergence of our method: a course mesh with 193 cells (M1), an intermediate mesh containing 728 cells (M2), and a refined mesh with 2853 cells (M3). The parameters of the simulation have been set as: domain:  $3m \times 1m$ ; Viscosity:  $0.10Ns/m^2$ ; Density:  $0.10Kg/m^3$ ; Reynolds:  $Re = 1$ ; Froude:  $Fr = 0.319275$ . Figure ( 5) shows the intermediate mesh and Fig. ( 6) presents a qualitative map of the velocity in  $x$ .

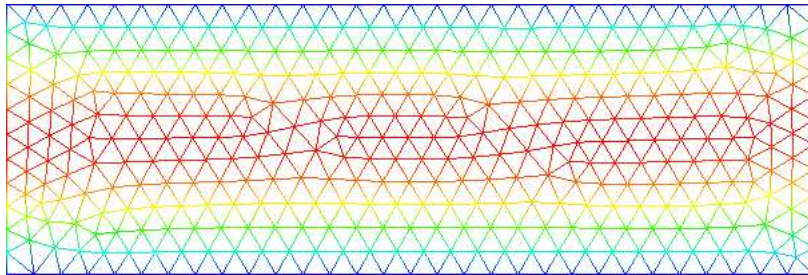


Figure 5: Intermediate mesh with 728 cells.

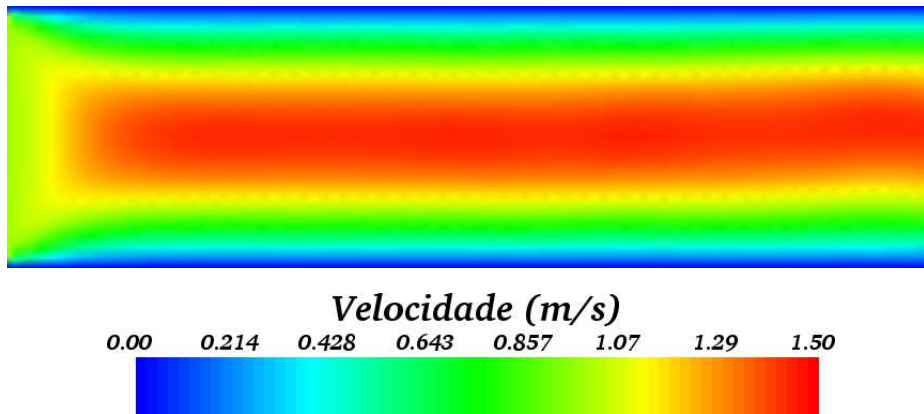


Figure 6: Velocity field in  $x$  direction.

Figure ( 7) shows a comparison between the analytical and numerical solution on a line in the middle of the channel.

One can observe that in the refined mesh it is difficult to distinguish the analytical from the numerical solution.

#### 4.2. Flow over a circular cylinder

We finish this section with an example illustrating the simulation of bidimensional around a circular cylinder. For numbers of Reynolds in the band  $40 < Re < 1000$ , the draining around the cylinder cause the detachment of vortices, being this laminar until  $Re = 300$ .

The boundary conditions were imposed in such a way that the flow was from the bottom up toward the top of the domain. A Newman boundary condition was used on the lateral boundaries. A constant velocity profile  $U$  was specified at the domain entrance.

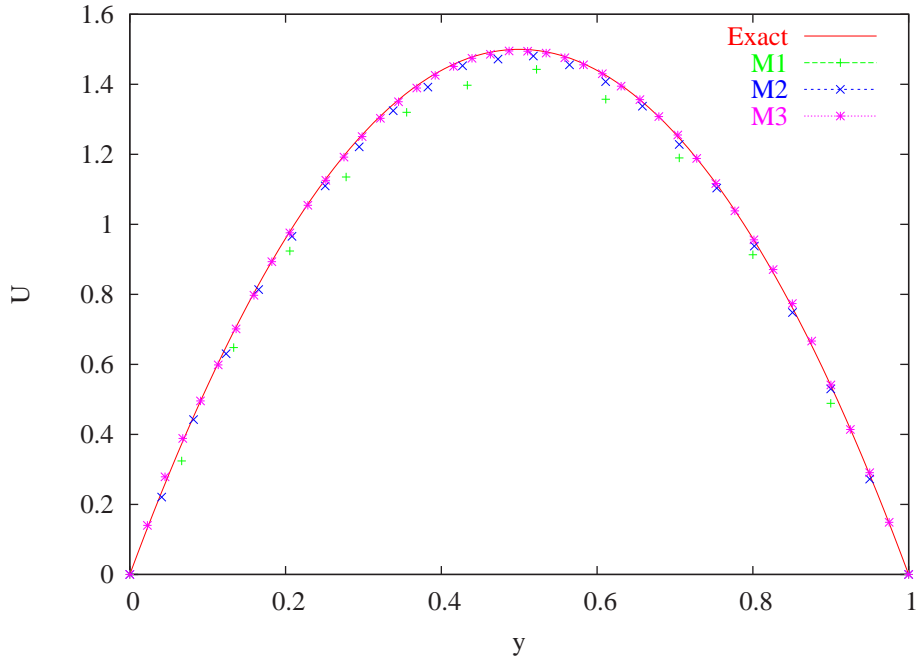


Figure 7: Comparing analytical and numerical results.

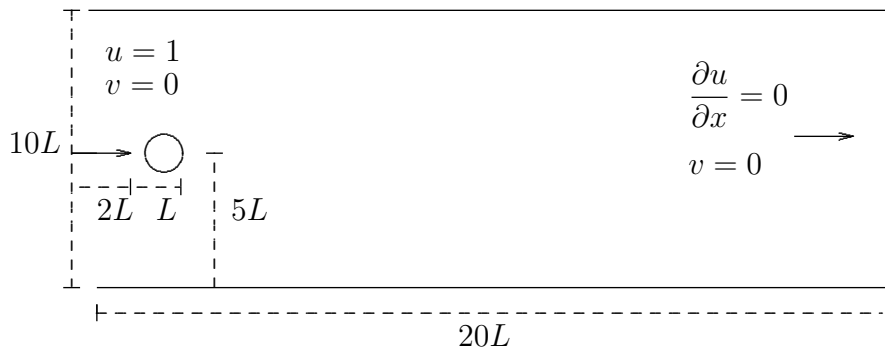


Figure 8: Computacional domain and boundary condition for fluid around a cylinder.

The simulation was in a fixed mesh with elements and  $h = 0.1$ , shown in the Fig. (9). The used Reynolds of number was  $Re = 100$ , sufficient for the detachment of vortices. The Fig. (10) - (13) shows the vorticity of fluid in some time different.

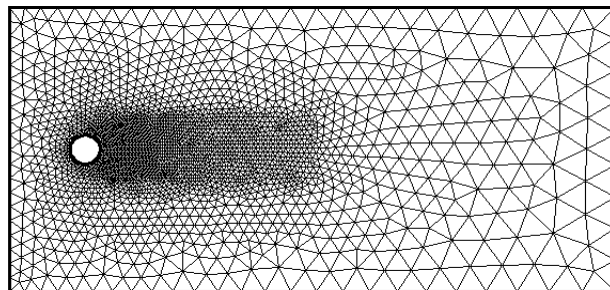


Figure 9: Mesh used in simulation around a cylinder.

Notice from Fig. (10) - (13) vorticity of fluid is in accordance with what we expected.

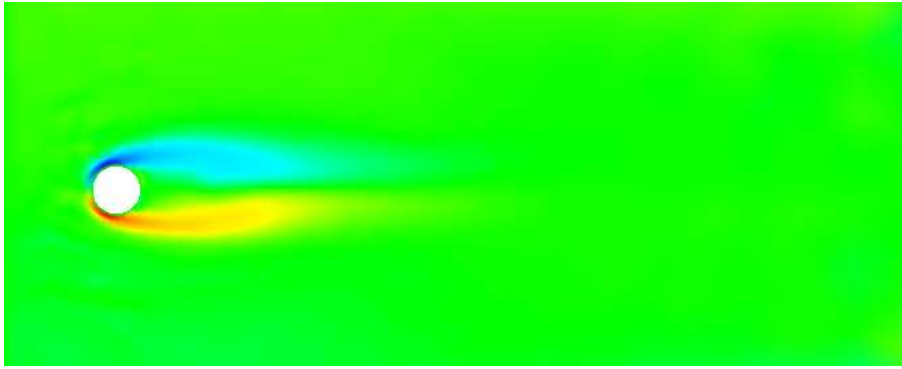


Figure 10: Vorticity field,  $t = 10s$ .

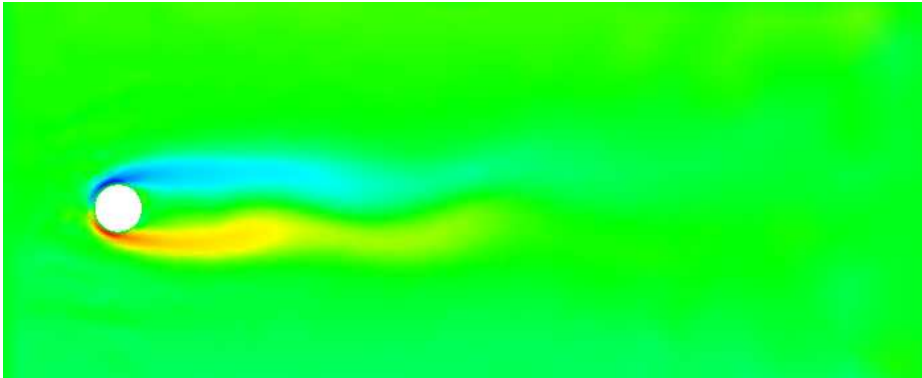


Figure 11: Vorticity field,  $t = 20s$ .

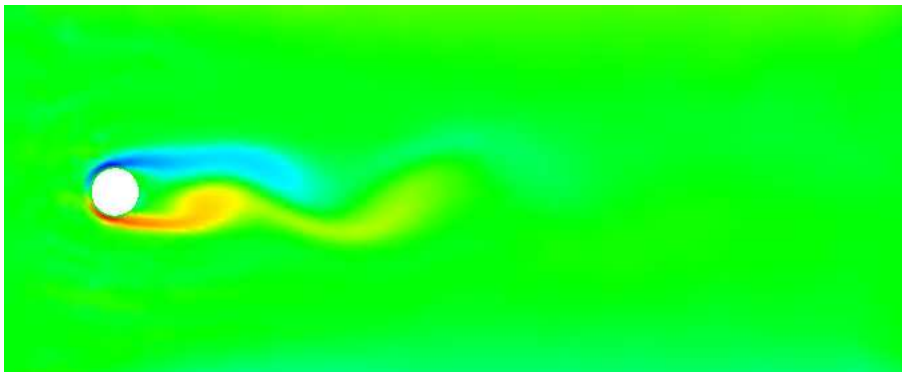


Figure 12: Vorticity field,  $t = 30s$ .

## 5. Conclusions and Future Work

In this work we present a new discretization technique that makes use of least square approximation to estimate derivatives. Such an approach has turned out to be very robust in fluid flow simulation, being thus a new alternative for handling these kind of problems. The strategy adopted to build the Poisson's matrix by Gaussian decomposition of the least square matrix is another contribution of this work.

The results of applying the proposed approach in the well known Hagen-Poiseuille flow and in a fluid flow simulation are very consistent, confirming thus the effectiveness of our method.

Although this new methodology has been developed envisioning a complete meshfree discretization scheme, we make use of a triangular mesh to improve the access to nodes neighborhood. In order to get rid of the mesh we are developing a set of data structures devoted to access neighborhood of nodes. A new scheme for discretizing the pressure on the nodes has also been investigated.

Another aspect we are considering is to employ high order semi-Lagrangian schemes, making it possible to deal with higher Reynolds number.

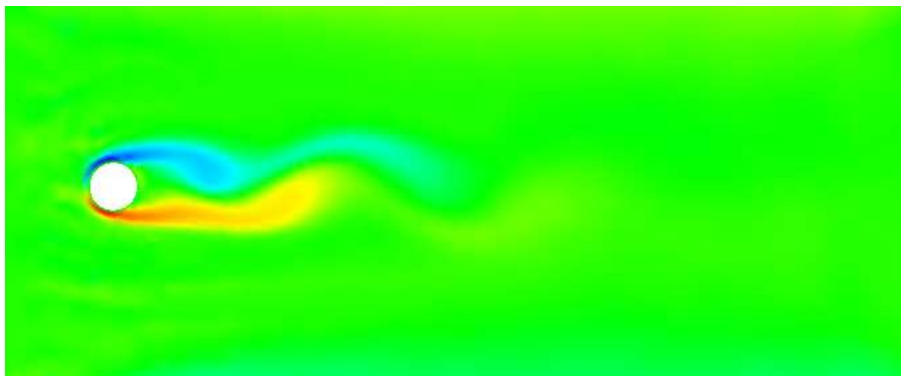


Figure 13: Vorticity field,  $t = 33s$ .

### Acknowledgments

We acknowledge the financial support of FAPESP - the State of São Paulo Research Funding Agency (Grant # 02/11686-6) and (Grant # 2004/16064-5)

### References

- [Batchelor, 1970] Batchelor, G., 1970, "An Introduction to Fluid Dynamics", Cambridge University Press, Cambridge.
- [Belytschko, 1994] Belytschko, T., 1994, Element-Free Galerkin Method, "Int. J. Numer. Meth. Engrg.", Vol. 37, pp. 229–256.
- [Levin, 1998] Levin, D., 1998, The approximation power of moving least-squares, "Math. Comp.", Vol. 67, No. 224.
- [Li and Liu, 2004] Li, S. and Liu, W., 2004, "Meshfree Particle Methods", Springer.
- [Liszka and Orkisz, 1980] Liszka, T. and Orkisz, J., 1980, The Finite Difference Method at Arbitrary Irregular Grids and Its Application in Applied Mechanics, "Computers and Structures", Vol. 11, pp. 83–95.
- [Liu et al., 1995] Liu, W., Jun, S., Li, S., Adee, J., and Belytschko, T., 1995, Reproducing kernel particle methods for structural dynamics, "Int. J. Numer. Meth. Engrg.", Vol. 38, pp. 1655–1679.
- [Lizier et al., 2003] Lizier, M. A. S., Nonato, L. G., Castelo, A., and Oliveira, M. C. F., 2003, A Topological Approach for Detecting and Extracting 2D Object Models from Images, "In: WICCGPI, SIBGRAPI 2003, São Carlos - SP", pp. 10–17.
- [Melenk and Babuska, 1996] Melenk, J. and Babuska, I., 1996, The partition of unity finite element method, "Comp. Meth. Appl. Mech. Engrg.", Vol. 139, pp. 289–314.
- [Monaghan, 1988] Monaghan, J., 1988, An introduction to SPH, "Computer Physics Communications", Vol. 48, pp. 89–96.
- [Nayroles et al., 1992] Nayroles, B., Touzot, G., and Villon, P., 1992, Generalizing the finite element method: diffuse approximation and diffuse elements, "Computational Mechanics", Vol. 10, pp. 307–318.
- [Oliveira, 1999] Oliveira, J., 1999, "Desenvolvimento de um Sistema de Simulação de escoamentos de Fluidos com Superfícies livres Bidimensionais."
- [Oliveira, 2000] Oliveira, J., 2000, "Manual do FreeFlow2D", Relatório Técnico, ICMC-USP-São Carlos, Brasil.
- [Peña, ] Peña, D., "Método de diferenças finitas generalizadas por mínimos quadrados", Dissertação de mestrado, ICMC-USP.
- [Pressman, 1994] Pressman, R. S., 1994, "Software Engineering: a practitioner's approach", European Edition, Rio de Janeiro.
- [Saad, 2003] Saad, Y., 2003, "Iterative methods for sparse linear systems", Philadelphia : SIAM.
- [Santos, 2005] Santos, F. O., 2005, "Simulação numérica de escoamento de fluido utilizando diferenças finitas generalizadas", Universidade de São Paulo, São Carlos, Dissertação de mestrado, ICMC-USP.

## KINETIC MODELING THROUGH THE INVERSE PROBLEM METHODOLOGY APPLIED TO SALICYLIC ACID ADSORPTION IN POLYMERIC AND ACTIVATED CHARCOAL ADSORBENTS

### Leônio Diógenes Tavares Câmara

Instituto Politécnico - IPRJ, Universidade do Estado do Rio de Janeiro - UERJ,  
P.O. Box 97282, 28601-970, Nova Friburgo, RJ, Brazil.  
dcamara@iprj.uerj.br

### Cesar Costapinto Santana

Faculdade de Engenharia Química, Universidade Estadual de Campinas - UNICAMP,  
P.O. Box 6066, 13081-970, Campinas, SP, Brazil.  
santana@feq.unicamp.br

### Antônio José da Silva Neto

Instituto Politécnico - IPRJ, Universidade do Estado do Rio de Janeiro - UERJ,  
P.O. Box 97282, 28601-970, Nova Friburgo, RJ, Brazil.  
ajsneto@iprj.uerj.br

**Abstract.** The understanding of adsorption mechanisms of compounds utilized in many medications and prescription drugs represents an important step for the development of different separation processes of such substances. The adsorption kinetic study of such chemical compounds through the application of new models permits a better comprehension of the evolved mechanisms, leading to the estimation of important parameters that will be incorporated in the equipment design. In this work a kinetic model of irreversible adsorption is implemented, which is dependent on both solute and site concentrations. It showed to be very effective correlating the simulated results with the adsorption experiments of salicylic acid through three adsorbents in different temperature conditions. The application of an inverse problem approach, with the minimization of the squared residues cost function, was successful in determining the kinetic parameters. The increase in the kinetic constants obtained with the temperature increase was followed by a reduction in the adsorption capacity of the adsorbents. Thermodynamic parameters were also estimated, leading to the comprehension about the equilibrium states of the adsorption systems considered.

**Keywords.** Kinetic modeling, Inverse Problem, Salicylic acid, Thermodynamic Analysis

### 1. Introduction

The comprehension about the adsorption mechanisms of chemical compounds utilized in medicines is an important step for the development and improvement of several adsorption processes evolved with such substances, such as separation, purification, decontamination of wastewaters etc.

The agitated batch process of adsorption is an important method used for equilibrium parameters estimation, which are applied in the processes modeling such as chromatography and simulated moving bed (SMB) separation. The hydrodynamic aspects of these processes become the kinetic modeling an interesting tool for the process modeling in obtaining parameters that will be incorporated in the equipment design.

The application of an inverse problem methodology for chromatographic systems is a new promising area since the solution of inverse problems has several relevant applications in the engineering field (Vasconcellos *et al.*, 2003). Other results can be found in Vasconcellos *et al.* (2002a, 2002b), Lugon *et al.* (2001), Denisov (2000), Felinger *et al.*, (2003), and Folly *et al.* (2005). In the work of Vasconcellos *et al.* (2003) the inverse problem formulation with the minimization of a cost function of squared residues was applied in the mass transfer parameters estimation of protein adsorption (Bovine serum albumin-BSA). The optimization method utilized is the same applied by Folly *et al.* (2005), the Levenberg-Marquardt. The method in both applications was successful in determining the parameters of non-linear equations.

Some contributions in the application of adsorption kinetic models for the liquid phase can be encountered through the following publications: Thomas (1944), Chase (1984), Sarkar and Chatteraj (1993), Hamadi *et al.* (2001, 2004), Otero *et al.* (2004), Gulen *et al.* (2005) and Aroguz (2006). An important contribution comes from the work of Chase (1984), which implemented semi-analytical expressions to model the adsorption phenomenon in agitated tanks and chromatographic columns. He considered the kinetic concepts to model the adsorption process as a reversible system with an overall rate of second-order. In a general point of view, the above publications, with exception of the Chase model (Chase, 1984), use simplified or empiric expressions for the kinetic models. The advantage of utilizing the

concepts of kinetic theory to develop new models is that the stoichiometric and order, related to the compounds in the adsorption system considered, can be varied and analyzed independently, leading to a better comprehension of the evolved kinetic phenomenology.

In this work an implemented kinetic model of adsorption is applied in the modeling of salicylic acid adsorption onto three different adsorbents, an activated carbon (F400) and two polymeric resins (F206 and F207), in three different temperature conditions. The model adjustment through the experimental data is done with the application of an inverse problem approach that minimize the square residues of a cost function.

## 2. Formulation of the kinetic model with inverse approach

The agitated adsorption techniques to measure adsorption parameters are modeled with the following expression for batch processes:

$$r_j = \frac{1}{V} \frac{dn_j}{dt} \quad (1)$$

in which  $r_j$ , that corresponds to the adsorption rate of component  $j$ , is proportional to the variation in the moles number of solute  $j$ . The tank volume ( $V$ ) is assumed constant.

The adsorption stoichiometry considered is represented in Fig. 1. It is related to an irreversible adsorption kinetic with a kinetic constant  $k_i$ . This adsorption mechanism depends both on the solute concentration (liquid phase) and the active surface concentration (site concentration in solid phase).

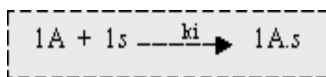


Figure 1. Representation of the adsorption mechanism.

The adsorption mechanism of Fig. 1 considers the adsorption of 1(one) mol of solute A in 1(one) mol of active site (s). The kinetic modeling, in terms of consumption rate of solute  $j$  ( $r_j$ ), is written in the following form.

$$(-r_j) = k_i C_j^n \cdot C_s^m \quad (2)$$

Where  $k_i$ ,  $C_j$  and  $C_s$  represent the kinetic constant, the concentration of solute  $j$  in the liquid phase and the concentration of sites of adsorption in the solid phase, respectively. For a first order elementary adsorption, the exponents  $n$  and  $m$  are equal to 1, which corresponds to an overall rate of second order. The irreversible adsorption is an adequate hypothesis, since in some experimental studies the desorption procedures are necessary to return the original adsorbent properties without solute traces (Pereira (1999), Silva (2000)). This is done with elution and washing steps.

With the considerations just described, Eq. (2) can be solved analytically through the expression (1), applying a balance in the moles number of active sites of adsorption, i.e.

$$C_t = C_s + C_{A,s} \quad (3)$$

in which  $C_t$  corresponds to the maximum concentration of adsorption sites, that is the sum through the concentration of vacant sites ( $C_s$ ) and occupied sites by solute A ( $C_{A,s}$ ). Another important balance is related to the concentration of solute A. In the balance of solute A, the initial concentration in the solution ( $C_{A0}$ ) corresponds to the sum of the final solute concentration in the solution ( $C_A$ ) and the adsorbed solute concentration in the solid phase ( $C_{A,s}$ ), i.e.

$$C_{A0} = C_A + C_{A,s} \quad (4)$$

The Combination of Eqs.(1-4) leads to.

$$\int \frac{dC_A}{C_A(a + C_A)} = \int -k_i dt \quad (5)$$

in which  $a = C_t - C_{A0}$ . Performing the integrations in Eq. (5) results.

$$\ln \left( \frac{C_A}{a + C_A} \right) = -a.k_i.t + c \quad (6)$$

We obtain the final expression of the irreversible adsorption kinetic model of overall order of 2 (IKM2) using the following contour conditions.

$$t = 0 \Rightarrow \begin{cases} C_A = C_{A0} \\ C_t = C_s \end{cases}, \quad t = \infty \Rightarrow \begin{cases} C_{eq} = C_A \\ C_t = C_{A,s} \end{cases} \quad (7)$$

The balance of the solute (Eq. 4) in the above conditions (Eq. 7) leads to the following form in the equilibrium.

$$C_{A0} = C_t + C_{eq} \quad (8)$$

The equations below (Eq. 9) present the final expressions for the concentration of solute A, with  $C_t$ ,  $C_{A0}$  and  $k_i$  in the time domain.

$$\frac{C_A}{a + C_A} = \left( \frac{C_{A0}}{a + C_A} \right) e^{-a.k_i.t} \quad \text{or} \quad C_A = \frac{a.C_{A0}}{(a + C_{A0})e^{-a.k_i.t} - C_{A0}} \quad (9)$$

Note that the implemented IKM2 expression comes from the moles balance following the moles relation shown in Fig. 1, which can be calculated independently of the volume of each phase. The parameter  $a$  in the IKM2 (Eq. 9) can be substituted by the term  $-C_{eq}$ , becoming the model dependent on the liquid phase parameters.

The correlations with the kinetic model implemented in this work (IKM2) were obtained through the experimental data of Otero *et al.* (2004). The correlation results from the kinetic model applied by Otero *et al.* (2004) were also analyzed for comparison purposes.

In order to obtain estimates for the unknown parameters we formulate the inverse problem implicitly, as an optimization problem in which we look for the minimum value of the cost function. The last one is given by the summation of the squared residues between experimental ( $C_{exp}$ ) and calculated ( $C_{cal}$ ) values of the solute concentration.

$$Q = \sum_{i=1}^N (C_{exp_i} - C_{cal_i})^2 \quad (10)$$

### 3. Results and discussion

The simulation results of IKM2 model are compared and correlated to three different adsorbent materials used in the adsorption of salicylic acid (Otero *et al.*, 2004).

The simulation results were obtained assuming an equilibrium concentration ( $C_{eq}$ ) that corresponds to the final concentration of the solute in the solution. The equilibrium concentration is an important parameter in the adsorption process once it determines the concentration of adsorption sites (solid phase concentration) from a moles balance with the initial concentration of the solute (see Eq. 8). In all simulations were considered an implicit error in the equilibrium concentration ( $C_{eq}$ ). The  $C_{eq}$  was varied in the range between -10 to 10% of its original value. It led to optimized simulation results with better fit between the model and the experimental data. This was confirmed by significant reductions in the cost function of squared residues ( $Q$ ).

The Figs. (2-4) present the correlation results between the IKM2 model and the experimental data from Otero *et al.* (2004). As can be observed from the Figs. (2-4), the IKM2 model showed high fit correlating the experimental points of all adsorbents in all temperature conditions.

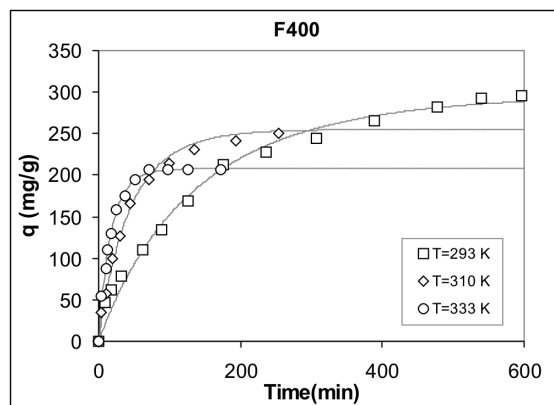


Figure 2. Correlation between the IKM2 model (lines) and the experimental adsorption data (points) of salicylic acid onto the F400 adsorbent.



The IKM2 model was highly satisfactory correlating the experimental data both in the initial period of time and in long times. It provided better correlation results, according to best fits, than those obtained by Otero *et al.*, 2004, which applied a linear driving force (LDF) model for the adsorption kinetic.

An interesting characteristic of the implemented model (IKM2) is the very small computational effort in obtaining the simulation results. It is related to the analytical form of the mathematical expression (Eq. 9). Besides the good agreement with the real experimental data, the kinetic model described (IKM2) requires only two parameters ( $C_{A0}$  and  $C_i$  or  $C_{eq}$ ) to obtain the rate kinetic constants ( $k_i$ ).

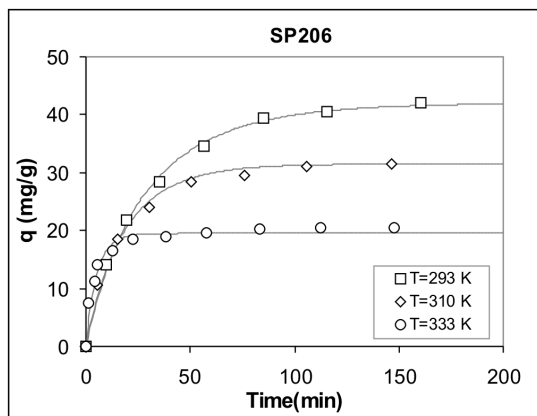


Figure 3. Correlation between the IKM2 model (lines) and the experimental adsorption data (points) of salicylic acid onto the F206 adsorbent.

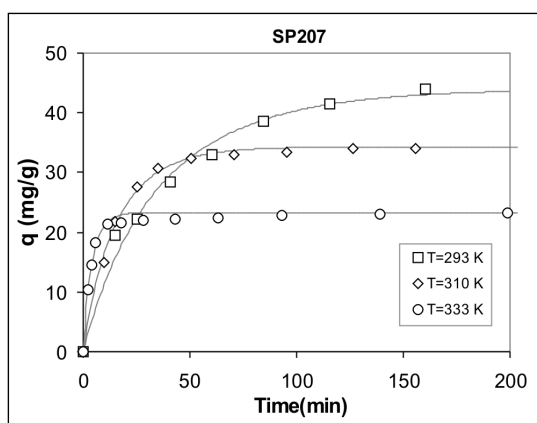


Figure 4. Correlation between the IKM2 model (lines) and the experimental adsorption data (points) of salicylic acid onto the F207 adsorbent.

The Table 1 presents the kinetic constants of adsorption estimated from the IKM2 model. As can be seen from the Table 1, there is an increase in the kinetic constants with the increase in the temperature. This increase in the kinetic constant is accompanied by a reduction in the adsorption capacity of the adsorbent, as can be verified from the Figs. (2-4). The estimated kinetic constants from the polymeric adsorbents (F206 and F207) were greater than those from the activated carbon (F400), although the last one presented the high adsorption capacity (vide Fig. 2).

For the processes design, the optimization between retention capacity and adsorption rate (proportional to the kinetic constants) must be carried out to reach the desired temperature for the maximum production with the minimum time for the processing.

Table 1. Estimated kinetic constants of adsorption for each adsorbent.

	Kinetic constants of adsorption (mL/mg.min)*10 <sup>3</sup>		
Temp. (K)	F400	F206	F207
293 K	1.578	6.278	8.052
310 K	4.877	14.148	11.993
333 K	12.194	52.642	41.394

Fig. 5 presents the behavior and the minimum of the cost function ( $Q$ ) by varying the kinetic constant ( $k_i$ ) for three different values of equilibrium concentration ( $C_{eq}$ ). These results correspond to the adsorption experiments through the adsorbent F400 in the temperature of 333 K. The cases studied correspond to the  $C_{eq}$  values of 76 mg/L, 83.1 mg/L and 79.2 mg/L. The first and second one correspond to the values obtained by Otero *et al.* (2004), from the Langmuir model correlation and the equilibrium kinetic experiments, respectively. The last one was obtained from the optimization procedure of  $C_{eq}$  (varied in the range of -10 to 10 % of its original). As can be seen, the minimum cost function presented smooth profiles that led to absolute minimums in some determined kinetic constants values. The optimized condition ( $C_{eq} = 79.2$  mg/L) showed the great absolute minimum. The  $C_{eq}$  cases from the Langmuir isotherm and the equilibrium kinetic experiments presented the high residue value if compared to the optimized condition. The last procedure can leads to erroneous estimation of parameters.

The results of the Fig. 5 showed that an improvement in the model correlation with experimental data can be obtained, what is related to the reduction in the cost function. It was reached optimizing the equilibrium concentration in a pre-determined range of error.

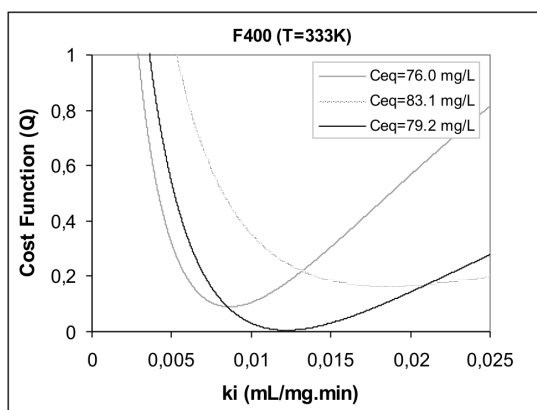


Figure 5. Cost function residues ( $Q$ ) of three equilibrium concentration values varying the kinetic constants.

The estimated kinetic constants (Table 1) were applied into the Arrhenius equation (Eq. 11), making possible the determination of the adsorption activation energies for the adsorbents studied.

$$k_i = A.e^{\frac{E_a}{R.T}} \tag{11}$$

where  $A$ ,  $E_a$  and  $T$  represents the frequency factor ( $\text{min}^{-1}$ ), The activation energy ( $\text{J/mol}$ ) and the temperature ( $\text{K}$ ), respectively, utilizing the constant  $R$  equals to  $8.314 \text{ J/mol.K}$ . The application of the logarithmic in both equation sides becomes the Arrhenius equation linear. The results are plotted in the Fig. 6. The activation energy comes from the angular term of the straight equation.

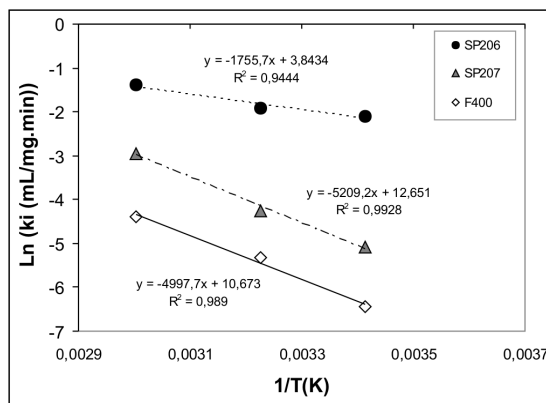


Figure 6. Linear form of the Arrhenius equation for the determination of the adsorption activation energy.

Other thermodynamics parameters were determined through the expression of the free energy of Gibbs ( $\Delta G$ ),

$$\Delta G = \Delta H - T.\Delta S \quad (12)$$

in which  $\Delta H$ ,  $\Delta S$  and  $T$  correspond to the variation of enthalpy and entropy and the temperature, respectively. The free energy of Gibbs, in the standard state ( $\Delta G^o$ ), is represented by the Eq. (13).

$$\Delta G^o = -R.T.Ln(K_c) \quad (13)$$

The term  $K_c$  represents the equilibrium constant. It is obtained through the ratio of the solid phase and the liquid phase concentration in the equilibrium (Aroguz, 2006). The Eqs. (12) and (13) combined lead to the linear form of the Eq. (14).

$$Ln(K_c) = -\left(\frac{\Delta H^o}{R.T}\right) + \left(\frac{\Delta S^o}{R}\right) \quad (14)$$

The parameters  $\Delta S_o$  and  $\Delta H_o$  were obtained, respectively, from the linear and angular coefficient of the Eq. (14). The thermodynamics parameters, from Eqs. (11) and (14) are presented in the Table 2.

Table 2. Estimated kinetic constants of adsorption for each adsorbent..

	F400	F206	F207
$E_a$ (KJ/mol)	41.55	14.6	43.31
$\Delta H^o$ (KJ/mol)	-9.75	-21.31	-17.34
$\Delta S^o$ (J/mol)	-40.44	-77.9	-65.24

Analyzing the data of Table 2 can be verified values of activation energies that correspond to mechanisms of physisorption (Aroguz, 2006). These values are in the high limit of physisorption, in which after it there is the chemisorption domain. The negative variation of enthalpy and entropy is related to the adsorption that is exothermic and to the decrease in the system disorder, respectively. An increase in the temperature leads to a decrease in the equilibrium constant, as the adsorption process is exothermic. This is verified from the Figs 2-4, in which the retention capacity decrease with the increase in the temperature. So, the adsorption is not spontaneous with the increase in the temperature.

#### 4. Conclusions

The analytical kinetic model of adsorption implemented (IKM2) has proved to be satisfactory due to a number of aspects. Firstly, it provided better agreements with experimental data when compared to other kinetic models, such as the kinetic model of linear driving force (Otero *et al.*, 2004). Other relevant aspects are related to the necessity of a small number of parameters in the model and the straightforward procedure obtaining the solution. It turned out to be very useful when associated to the square residues inverse modeling.

The consideration of an acceptable error domain for the equilibrium concentration ( $C_{eq}$ ) provided good results by reductions in the residues cost function, which led to a better experimental correlation with an increase in the accuracy of the parameters estimated. In the procedures evolved with the correlation of experimental equilibrium data with isotherm models, such as Langmuir model, a special treatment must be considered since implicit errors can lead to erroneous estimation of parameters.

The simulated results indicated an increase in the kinetic parameters ( $k_i$ ) with an increase in the temperature for all the adsorbents studied. The increase in the kinetic constants or adsorption velocity was followed by a reduction in the retention capacity of the solid phase. It was confirmed by the thermodynamic parameters obtained.

The observed decrease in the capacity of adsorbent with the increase in the temperature must be analyzed and optimized in terms of adsorption velocity (related to the kinetic constants) as the last one decrease the time for the processing.

#### 5. Acknowledgement

The authors acknowledgment the financial support provided by CNPq (Conselho Nacional de Desenvolvimento Científico e Regional), FAPERJ (Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro) and FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo).

## **6. References**

- Aroguz, A.Z., 2006, "Kinetics and Thermodynamics of Adsorption of Azinphosmethyl from Aqueous Solution onto Pyrolyzed (at 600° C) Ocean Peat Moss (*Sphagnum* sp.)", *Journal of Hazardous Materials*. (In press)
- Chase, H.A., 1984, "Prediction of the Performance of Preparative Affinity Chromatography", *J. Chromatography*, Vol. 297, pp. 179-202.
- Denisov, A.M., 2000, "Inverse Problems of Absorption Dynamics, Proc. Minisymposium on Inverse Problems in Medicine", *Engineering and Geophysics-Proc. XXIII Brazilian Congress on Applied and Computational Mathematics*, Santos- Brazil, pp. 24-32. (Invited Lecture)
- Felinger, A., Zhou, D., Guiochon, G., 2003, "Determination of the Single Component and Competitive Adsorption Isotherms of the 1-Indanol Enantiomers by Inverse Method", *Journal of Chromatography A*, Vol. 1005, pp. 35-49.
- Folly, F.M., Silva Neto, A.J., Santana, C.C., 2005, "An Inverse Mass Transfer Problem for the Characterization of Simulated Moving Bed Adsorption Columns", *Proceedings 5th Intern. Conf. on Inverse Problems in Eng*, Cambridge, UK, 11-15th July..
- Gulen, J., Aroguz, A.Z., Dalgin, D., 2005, "Adsorption Kinetics of Azinphosmethyl from Aqueous Solution onto Pyrolyzed Horseshoe Sea Crab Shell from the Atlantic Ocean", *Bioresource Technology*, Vol. 96, pp. 1169-1174.
- Hamadi, N.K., Chen, X.D., Farid, M.M., Lu, M.G.Q., 2001, "Adsorption Kinetics for the Removal of Chromium(VI) from Aqueous Solution by Adsorbents Derived from Used Tires and Sawdust", *Chemical Engineering Journal*, Vol. 84, pp. 95-105.
- Hamadi, N.K., Swaminathan, S., Chen, X.D., 2004, "Adsorption of Paraquat Dichloride From Aqueous Solution by Activated Carbon Derived from Used Tires", *Journal of Hazardous Materials B*, Vol. 112, pp. 133-141.
- Lugon Jr, J., Silva Neto, A.J., Santana, C.C., 2001, "An Inverse Mass Transfer Problem in Gas-Liquid Adsorption in Columns of Bubble and Foam", *Inverse Problems in Engineering Symposium*, College Station, Texas, EUA.
- Otero, M., Grande, C.A., Rodrigues, A.E., 2004, "Adsorption of Salicylic Acid onto Polymeric Adsorbents and Activated Charcoal", *Reactive and Functional Polymers*, Vol. 60, pp. 203-213.
- Pereira, J.A.M., 1999, "Adsorption of  $\beta$ -Galactosidase from *Scopulariopsis* sp in Ion Exchange Resin with Purification and Scaling-up objective", D.Sc. Thesis, Universidade Estadual de Campinas, São Paulo, Brazil. (In Portuguese)
- Sarkar, D., Chattoraj, D.K., 1993, "Activation Parameters for Kinetics of Protein Adsorption at Silica-Water Interface", *Journal of Colloid and Interface Science*, Vol. 157, pp. 219-226.
- Silva, F.R.C., 2000, "Study of Inulinases Adsorption in Columns with Ion Exchange Resin: Experimental Parameters and Modeling", D.Sc. Thesis, Universidade Estadual de Campinas, São Paulo, Brazil. (In Portuguese)
- Thomas, H., 1944, "Heterogeneous Ion Exchange in Flowing System", *J. Am. Chem. Soc.*, Vol. 66, pp. 1664-1668.
- Vasconcellos, J.F.V., Silva Neto, A.J., Soeiro, F.J.P., Santana, C.C., 2002a, "Parameter Estimation in Adsorption Columns with a Stochastic Global Method", *Proc. 4th Internat. Conf. on Inverse Problems*. Rio de Janeiro, Brazil.
- Vasconcellos, J.F.V., Silva Neto, A.J., Santana, C.C., 2002b, "An Inverse Mass Transfer Problem of Parameter Estimation in Adsorption Columns", *Proc. 5th World Congress on Computational Mechanics*. Vienna, Austria.
- Vasconcellos, J.F.V., Silva Neto, A.J., Santana, C.C., 2003 "An Inverse Mass Transfer Problem in Solid-Liquid Adsorption Systems", *Inverse Problems in Eng.*, Vol. 11, No. 5, pp. 391-408.

## THERMALLY DEVELOPING NON-NEWTONIAN LAMINAR FLOW IN ECCENTRIC ANNULAR DUCTS

### Evaldiney Ribeiro Monteiro

Mining Engineering Department, CSSPA, Universidade Federal do Pará, UFPA  
Campus II, Folha 17, Quadra 04, Lote Especial, 68505-080, Marabá, PA, Brazil  
monteiro@ufpa.br

### Emanuel Negrão Macêdo

### João Nazareno Nonato Quaresma

Chemical and Food Engineering Department, CT, Universidade Federal do Pará, UFPA  
Campus Universitário do Guamá, 66075-110, Belém, PA, Brazil  
quaresma@ufpa.br

### Renato Machado Cotta

Mechanical Engineering Department – POLI/COPPE, Universidade Federal do Rio de Janeiro, UFRJ  
Cx. Postal 68503 – Cidade Universitária, 21945-970, Rio de Janeiro, RJ, Brazil  
cotta@mecanica.coppe.ufrj.br

**Abstract.** Thermally developing laminar flow in eccentric annular ducts involving non-Newtonian power-law fluids is analyzed by using the Generalized Integral Transform Technique (GITT) to solve the associated energy equation. The mathematical formulation is constructed based on the cylindrical coordinates system in such a way that the solid surfaces are described in terms of internal and external radii as functions of the angular coordinate, thus avoiding discontinuities in the boundary conditions. This thermal problem is here analyzed under the boundary condition of prescribed wall temperature. Numerical results for the temperature field and Nusselt numbers were produced for different values of the governing parameters, i.e., eccentricity and aspect ratio along the thermal entry region, which were critically compared with previously reported ones, in order to illustrate the usefulness of the employed integral transform approach with automatic error control.

**Keywords.** Eccentric annular ducts, Thermally developing flow, Power-law fluids, Integral transforms.

### 1. Introduction

The study of thermally developing flow in eccentric annular ducts is important mainly due to its frequent appearance in several industrial applications, such as in heat exchange devices including the most common double-pipe configuration. In this type of equipment, due to imperfections and tolerances in the manufacturing steps, the eccentricity eventually caused, may or not be important. However, there are few typical applications where this effect is more pronounced and produced on purpose. Oil and gas drilling wells, polymer and plastic extrusion processes and nuclear reactors are some of the situations that reflect the importance of eccentricity in annular passages. In addition, when dealing with purely viscous non-Newtonian fluids, the essential heat and fluid flow analysis is much less available in the open literature, while commonly required in different industries, namely, chemical, food processing and pharmaceutical. There, the power-law model can adequately describe the rheology of a wide variety of fluids.

The literature survey brings up several works that dealt with thermally developing or fully developed flow in this geometric configuration, such as the pioneering works of Piercy *et al.* (1933), Stevenson (1949), Snyder and Goldstein (1965) and Jonsson and Sparrow (1965), which concentrated their analyses in the fluid flow, while Cheng and Hwang (1968), Trombetta (1971) and Suzuki *et al.* (1991) analyzed the heat transfer problem under different sets of boundary conditions. Details of such earlier works can be found in the compilation of Shah and London (1978). These studies have regained interest in the more recent works of Manglik and Fang (1995), Fang *et al.* (1999), Manglik and Fang (2002) and Escudier *et al.* (2002) in which the effects of eccentricity and duct rotation were investigated for the flow and heat transfer of non-Newtonian fluids. Escudier *et al.* (2002), in addition, recently offered an excellent literature review for flow and heat transfer in eccentric annular ducts involving Newtonian and non-Newtonian fluids.

With the advance of a structured hybrid analytical-numerical approach for convection-diffusion problems along the last two decades, named the Generalized Integral Transform Technique (GITT), Cotta (1993) & Cotta (1994), including the solution of elliptic diffusion-type problems defined within irregular domains (Aparecido *et al.*, 1989), it was possible to apply this analysis to fully developed laminar flow within ducts of various shapes, such as trapezoidal, triangular, and hexagonal ducts (Aparecido and Cotta, 1987; Aparecido *et al.*, 1989; Aparecido and Cotta, 1990; Barbuto and Cotta, 1997). Fully developed laminar flow and heat transfer of non-Newtonian fluids inside irregular ducts of different geometric configurations was also treated (Chaves *et al.*, 2001a; 2001b; 2004, Monteiro *et al.*, 2004), again through extension of the GITT approach, yielding accurate numerical results for quantities of practical interest such as the Fanning friction factor and Nusselt numbers, within a wide range of the governing parameters.

The present study is aimed at applying the so-called Generalized Integral Transform Technique (GITT) to solve the energy equation for thermally developing laminar flow of power-law fluids inside eccentric annular ducts subjected to

constant temperatures either at the inner or outer duct walls. The cylindrical coordinates system is used in the mathematical formulation, so that the solid surfaces are described in the form of internal and external radii as functions of the angular coordinate, and thus avoiding more involved formulation in other coordinates systems and cumbersome domain transformation approaches. An analysis of convergence of the eigenfunction expansion is performed and a set of benchmark results is established for quantities of practical interest, such as dimensionless average temperature and local Nusselt numbers, within a wide range of the dimensionless axial coordinate, different power-law indices, aspect ratios and dimensionless eccentricities. Comparisons are then critically performed with previously reported results from direct numerical approaches along both, fully developed and thermally developing regions.

## 2. Analysis

Thermally developing flow of a non-Newtonian fluid inside eccentric annular ducts is considered according to Fig. (1). The non-Newtonian viscosity expression  $\eta = K\dot{\gamma}^{n-1}$  is described according to the Ostwald-de Waele model or power law model (Bird *et al.*, 1987), where  $K$  is the fluid consistency index (given in  $N.s^n/m^2$ ),  $n$  is the power-law index (dimensionless) and  $\dot{\gamma}$  is the rate-of-deformation tensor. The power-law fluid classification is (according to the  $n$  value): pseudoplastic ( $n < 1$ ), Newtonian ( $n = 1$  and  $K = \mu$ ) and dilatant fluids ( $n > 1$ ). This rheological model was employed to obtain the fully developed velocity profile in eccentric annular ducts in a previous work by Monteiro *et al.* (2004), which is used as input for the present thermal problem formulation. In addition, the duct walls are subjected to different situations of prescribed temperature. Therefore, the dimensionless energy equation for constant properties flow, neglecting axial conduction and viscous dissipation, is written as:

$$W(R,\theta)\frac{\partial T}{\partial Z} = \frac{1}{R}\frac{\partial}{\partial R}\left(R\frac{\partial T}{\partial R}\right) + \frac{1}{R^2}\frac{\partial^2 T}{\partial \theta^2}, \quad \text{in } Z > 0, R_1(\theta) < R < R_2(\theta), 0 < \theta < \pi \quad (1a)$$

with inlet and boundary conditions given, respectively, as follows:

$$T(R,\theta,0) = 0, \quad R_1(\theta) \leq R \leq R_2(\theta), \quad 0 \leq \theta \leq \pi \quad (1b)$$

$$\left. \begin{aligned} T(R_1(\theta),\theta,Z) = 1, \quad \frac{\partial T(R_2(\theta),\theta,Z)}{\partial n} = 0, \quad Z > 0, \quad \text{Case 1A} \\ \frac{\partial T(R_1(\theta),\theta,Z)}{\partial n} = 0, \quad T(R_2(\theta),\theta,Z) = 1, \quad Z > 0, \quad \text{Case 1B} \end{aligned} \right\} \quad (1c-f)$$

$$\frac{\partial T(R,0,Z)}{\partial \theta} = 0, \quad \frac{\partial T(R,\pi,Z)}{\partial \theta} = 0, \quad Z > 0 \quad (1g,h)$$

where  $\partial/\partial n$  represents the normal derivative to the channel wall surface, in the sense leaving the medium.

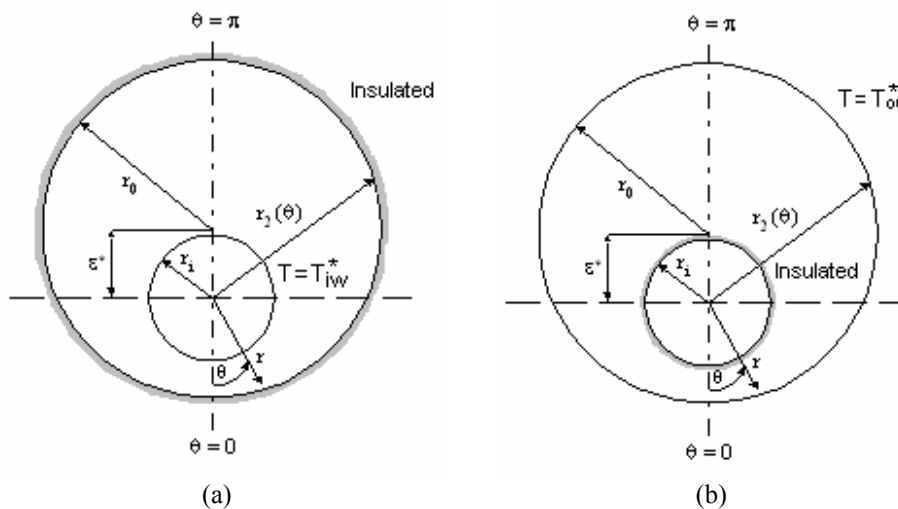


Figure 1 - Geometry and coordinates system for thermally developing flow in eccentric annular ducts: (a) prescribed temperature at the inner wall; (b) prescribed temperature at the outer wall.

The following dimensionless groups were employed in Eqs. (1) above:

$$R = \frac{r}{r_o}, R_1(\theta) = \frac{r_1(\theta)}{r_o}, R_2(\theta) = \frac{r_2(\theta)}{r_o} = \sqrt{1 - \varepsilon^2 (1 - \gamma)^2 \sin^2 \theta} - \varepsilon(1 - \gamma) \cos \theta, \gamma = \frac{r_i}{r_o}, \varepsilon = \frac{\varepsilon^*}{r_o - r_i}, Z = \frac{z}{D_h Pe} \quad (2a-f)$$

$$V_z(R, \theta) = \frac{v_z(r, \theta)}{\left[ \left( -\frac{dP}{dZ} \right) \frac{D_h^{n+1}}{K} \right]^{1/n}}, W(R, \theta) = \frac{V_z(R, \theta)}{4(1 - \gamma)^2 V_{z,m}}, Pe = Re Pr = \frac{v_{z,m} D_h}{\alpha} = \frac{\rho c_p}{k} v_{z,m} D_h \quad (2h-i)$$

$$\left. \begin{aligned} Re = \frac{\rho v_{z,m}^{2-n} D_h^n}{K}, Pr = \frac{K}{\rho v_{z,m}^{1-n} D_h^{n-1} \alpha}, T(R, \theta, Z) = \frac{T^*(r, \theta, z) - T_e}{T_{iw}^* - T_e}, \text{ Case 1A} \\ T(R, \theta, Z) = \frac{T^*(r, \theta, z) - T_e}{T_{ow}^* - T_e}, \text{ Case 1B} \end{aligned} \right\} \quad (2j-m)$$

The main dimensionless groups in Eqs. (2) above are:  $R_2(\theta)$  (dimensionless function that describes the outer surface),  $\gamma$  (aspect ratio),  $\varepsilon$  (dimensionless eccentricity),  $Pe$  (Péclet number),  $Re$  (Reynolds number) and  $Pr$  (Prandtl number).  $D_h$  is the hydraulic diameter defined as  $D_h = 2r_o(1 - \gamma)$ .

In order to homogenize Eqs. (1c) and (1f), and thus improve the computational performance, a filter is defined as:

$$T(R, \theta, Z) = 1 + \Phi(R, \theta, Z) \quad (3)$$

After introducing Eq. (3) into Eqs. (1), the following problem for the potential  $\Phi(R, \theta, Z)$  results:

$$W(R, \theta) \frac{\partial \Phi}{\partial Z} = \frac{1}{R} \frac{\partial}{\partial R} \left( R \frac{\partial \Phi}{\partial R} \right) + \frac{1}{R^2} \frac{\partial^2 \Phi}{\partial \theta^2}, \text{ in } Z > 0, R_1(\theta) < R < R_2(\theta), 0 < \theta < \pi \quad (4a)$$

$$\Phi(R, \theta, 0) = -1, \quad R_1(\theta) \leq R \leq R_2(\theta), \quad 0 \leq \theta \leq \pi \quad (4b)$$

$$\left. \begin{aligned} \Phi(R_1(\theta), \theta, Z) = 0, \quad \frac{\partial \Phi(R_2(\theta), \theta, Z)}{\partial n} = 0, \quad Z > 0, \text{ Case 1A} \\ \frac{\partial \Phi(R_1(\theta), \theta, Z)}{\partial n} = 0, \quad \Phi(R_2(\theta), \theta, Z) = 0, \quad Z > 0, \text{ Case 1B} \end{aligned} \right\} \quad (4c-f)$$

$$\frac{\partial \Phi(R, 0, Z)}{\partial \theta} = 0, \quad \frac{\partial \Phi(R, \pi, Z)}{\partial \theta} = 0, \quad Z > 0 \quad (4g,h)$$

The dimensionless velocity profile is given from the solution of the flow equations by the application of the GITT approach itself, for a non-Newtonian power-law fluid flowing within eccentric annular ducts, as an infinite series in the form (Monteiro *et al.*, 2004):

$$W(R, \theta) = \sum_{i=1}^{\infty} \Psi_i(R, \theta) \bar{V}_{z,i}(\theta), \quad \bar{V}_{z,i}(\theta) = \frac{1}{L_i(\theta)} \int_{R_1(\theta)}^{R_2(\theta)} \frac{\Psi_i(R, \theta) V_z(R, \theta)}{R} dR \quad (5a,b)$$

$$\Psi_i(R, \theta) = \sin \left\{ \lambda_i(\theta) \ln \left[ \frac{R_2(\theta)}{R} \right] \right\}, \quad \lambda_i(\theta) = \frac{i\pi}{\ln \left[ \frac{R_2(\theta)}{R_1(\theta)} \right]}, \quad L_i(\theta) = \frac{1}{2} \ln \left[ \frac{R_2(\theta)}{R_1(\theta)} \right], \quad i = 1, 2, 3, \dots \quad (5c-e)$$

In Eqs. (5), the quantities  $\bar{V}_{z,i}(\theta)$  represent the transformed potentials for the velocity field, which were numerically obtained through appropriate subroutines such as DBVFPD from the IMSL Library (1991), with local error control.

Due to the non-separable nature of the velocity profile given in Eq. (4a) and consequently, of the related eigenvalue problem needed to solve the energy equation through well-known analytical methods such as the classical integral transform technique (Mikhailov and Özisik, 1984), an exact solution of problem (4) is not possible. On the other hand, with the advances on the so-called GITT approach for the hybrid analytical-numerical solution of this class of non-transformable problem, it is possible to avoid these difficulties as now demonstrated (Aparecido and Cotta, 1992; Cotta, 1993, Chaves *et al.*, 2002). For this purpose, in order to alleviate the difficulties related to the eigenvalue problem and to progress with the application of the generalized integral transform technique, the following auxiliary eigenvalue problems are chosen:

R-direction:

$$\frac{1}{R} \frac{\partial}{\partial R} \left[ R \frac{\partial \Gamma_i(R, \theta)}{\partial R} \right] + \mu_i^2(\theta) \frac{1}{R^2} \Gamma_i(R, \theta) = 0, \quad R_1(\theta) < R(\theta) < R_2(\theta) \quad (6a)$$

$$\left. \begin{aligned} \Gamma_i[R_1(\theta), \theta] = 0, \quad \frac{\partial \Gamma_i[R_2(\theta), \theta]}{\partial R} = 0, \quad \text{Case 1A} \\ \frac{\partial \Gamma_i[R_1(\theta), \theta]}{\partial R} = 0, \quad \Gamma_i[R_2(\theta), \theta] = 0, \quad \text{Case 1B} \end{aligned} \right\} \quad (6b-e)$$

$\theta$ -direction:

$$\frac{d^2 \varphi_\ell(\theta)}{d\theta^2} + \beta_\ell^2 \varphi_\ell(\theta) = 0, \quad 0 < \theta < \pi \quad (7a)$$

$$\frac{d\varphi_\ell(0)}{d\theta} = 0, \quad \frac{d\varphi_\ell(\pi)}{d\theta} = 0 \quad (7b,c)$$

which are readily solved to yield eigenfunctions and eigenvalues, as follows:

$$\left. \begin{aligned} \Gamma_i(R, \theta) = \cos \left\{ \mu_i(\theta) \ln \left[ \frac{R_2(\theta)}{R} \right] \right\}, \quad \text{Case 1A} \\ \Gamma_i(R, \theta) = \sin \left\{ \mu_i(\theta) \ln \left[ \frac{R_2(\theta)}{R} \right] \right\}, \quad \text{Case 1B} \end{aligned} \right\}, \quad \varphi_\ell(\theta) = \cos(\beta_\ell \theta) \quad (8a-c)$$

$$\mu_i(\theta) = \frac{(2i-1)\pi}{2} \frac{1}{\ln \left[ \frac{R_2(\theta)}{R_1(\theta)} \right]}, \quad i = 1, 2, 3, \dots, \quad \beta_\ell = \ell - 1, \quad \ell = 1, 2, 3, \dots \quad (8d,e)$$

Also, the eigenfunctions above enjoy the following orthogonality properties:

$$\int_{R_1(\theta)}^{R_2(\theta)} \frac{\Gamma_i(R, \theta) \Gamma_j(R, \theta)}{R} dR = \begin{cases} 0, & i \neq j \\ M_i(\theta), & i = j \end{cases}, \quad M_i(\theta) = \int_{R_1(\theta)}^{R_2(\theta)} \frac{\Gamma_i^2(R, \theta)}{R} dR = \frac{1}{2} \ln \left[ \frac{R_2(\theta)}{R_1(\theta)} \right] \quad (9a,b)$$

$$\int_0^\pi \varphi_\ell(\theta) \varphi_m(\theta) d\theta = \begin{cases} 0, & \ell \neq m \\ N_\ell, & \ell = m \end{cases}, \quad N_\ell = \int_0^\pi \varphi_\ell^2(\theta) d\theta = \begin{cases} \pi, & \ell = 1 \\ \frac{\pi}{2}, & \ell > 1 \end{cases} \quad (9c,d)$$

Eigenvalue problems (6) and (7) allow the development of the following integral transform pair:

$$\tilde{\Phi}_{i\ell}(Z) = \int_0^\pi \int_{R_1(\theta)}^{R_2(\theta)} \frac{\Gamma_i(R, \theta) \varphi_\ell(\theta) \Phi(R, \theta, Z)}{R M_i(\theta) N_\ell} dR d\theta, \quad \text{transform} \quad (10a)$$

$$\Phi(R, \theta, Z) = \sum_{i=1}^{\infty} \sum_{\ell=1}^{\infty} \Gamma_i(R, \theta) \varphi_\ell(\theta) \tilde{\Phi}_{i\ell}(Z), \quad \text{inverse} \quad (10b)$$

The next step in the GITT approach is the integral transformation process itself, when all independent variables are eliminated from the partial differential formulation but one, in this case the dimensionless axial coordinate. To obtain the resulting system of ordinary differential equations for the transformed potentials  $\tilde{\Phi}_{i\ell}(Z)$ , the partial differential equation (4a) is multiplied by  $\Gamma_i(R, \theta) \varphi_\ell(\theta) / R$ , integrated over the domains  $[R_1(\theta), R_2(\theta)]$  in the R-direction and  $[0, \pi]$  in the  $\theta$ -direction, and the inverse formula, Eq. (10b), is employed in place of the filtered temperature distribution  $\Phi(R, \theta, Z)$ , resulting in the following transformed ordinary differential system:

$$\sum_{j=1}^{\infty} \sum_{m=1}^{\infty} F_{ij\ell m} \frac{d\tilde{\Phi}_{jm}(Z)}{dZ} + \sum_{j=1}^{\infty} \sum_{m=1}^{\infty} G_{ij\ell m} \tilde{\Phi}_{jm}(Z) = 0, \quad Z > 0; \quad j = 1, 2, \dots \quad m = 1, 2, \dots \quad (11a)$$

$$\tilde{\Phi}_{i\ell}(0) = \tilde{h}_{i\ell} \quad (11b)$$

where,



$$F_{ij\ell m} = \int_0^\pi \int_{R_1(\theta)}^{R_2(\theta)} \frac{R\Gamma_i(R,\theta)\Gamma_j(R,\theta)W(R,\theta)\varphi_\ell(\theta)\varphi_m(\theta)}{M_i(\theta)N_\ell} dRd\theta, \quad G_{ij\ell m} = \beta_\ell^2 \delta_{ij} \delta_{\ell m} + G1_{ij\ell m} - G2_{ij\ell m} - 2G3_{ij\ell m} \quad (11c,d)$$

$$\tilde{h}_{i\ell} = - \int_0^\pi \int_{R_1(\theta)}^{R_2(\theta)} \frac{\Gamma_i(R,\theta)\varphi_\ell(\theta)}{RM_i(\theta)N_\ell} dRd\theta, \quad G1_{ij\ell m} = (p-1) \int_0^\pi \frac{\varphi_\ell(\theta)\varphi_m'(\theta)}{M_i(\theta)N_\ell} \frac{\varepsilon(1-\gamma)\sin\theta}{\sqrt{1-\varepsilon^2(1-\gamma)^2\sin^2\theta}} d\theta + \delta_{ij} \int_0^\pi \mu_i^2(\theta) \frac{\varphi_\ell(\theta)\varphi_m(\theta)}{N_\ell} d\theta \quad (11e,f)$$

$$G2_{ij\ell m} = \int_0^\pi \int_{R_1(\theta)}^{R_2(\theta)} \left[ \frac{\Gamma_i(R,\theta)}{R} \frac{\partial^2 \Gamma_j(R,\theta)}{\partial \theta^2} \frac{\varphi_\ell(\theta)\varphi_m(\theta)}{M_i(\theta)N_\ell} \right] dRd\theta, \quad G3_{ij\ell m} = \int_0^\pi \int_{R_1(\theta)}^{R_2(\theta)} \left[ \frac{\Gamma_i(R,\theta)}{R} \frac{\partial \Gamma_j(R,\theta)}{\partial \theta} \frac{\varphi_\ell(\theta)\varphi_m'(\theta)}{M_i(\theta)N_\ell} \right] dRd\theta \quad (11g,h)$$

$$p = \begin{cases} 0, & \text{Case 1A} \\ 1, & \text{Case 1B} \end{cases} \quad (11i)$$

In Eq. (11a) each summation is associated with the eigenfunction expansion in a corresponding spatial coordinate, for computational purposes, the series solution given by Eq. (10b) is, in general, truncated to a finite number of terms in the summation, in order to compute the potential  $\Phi(R,\theta,Z)$ . The solution convergence is verified by comparing the values for the potential obtained with the truncated series for different numbers of retained terms. The coupled system of ordinary differential equations (11a) is solved by efficient numerical algorithms for initial value problems, such as in subroutine DIVPAG from the IMSL package (1991), with high accuracy and automatic control along the solution procedure. Then, after the transformed potentials are numerically obtained, quantities of practical interest are determined from the analytical inverse formula (10b), such as the dimensionless average temperature

$$T_{av}(Z) = 1 + \frac{2}{A_t(\gamma)} \int_0^\pi \int_{R_1(\theta)}^{R_2(\theta)} \frac{RV_Z(R,\theta)}{V_{Z,m}} \Phi(R,\theta,Z) dRd\theta \quad (12a)$$

or,

$$T_{av}(Z) = 1 + \frac{2}{A_t(\gamma)V_{Z,m}} \sum_{i=1}^\infty \sum_{\ell=1}^\infty I_{i\ell} \tilde{\Phi}_{i\ell}(Z) \quad (12b)$$

where  $A_t(\gamma)$  is the cross-section area of the annular passage and,

$$I_{i\ell} = \sum_{k=1}^\infty \int_0^\pi \int_{R_1(\theta)}^{R_2(\theta)} R\Gamma_i(R,\theta)\Psi_k(R,\theta)\varphi_\ell(\theta)\bar{V}_{Z,k}(\theta) dRd\theta \quad (12c)$$

The local Nusselt number can be calculated by making use of the temperature gradients at the walls integrated over the perimeter, or utilizing the axial gradient of the average temperature,

$$Nu_w(Z) = \frac{h(z)D_h}{k} = - \frac{2(1-\gamma)}{\pi} \frac{\int_0^\pi \left( \frac{\partial T}{\partial n} \right)_{R_w(\theta)} d\theta}{1-T_{av}(Z)}, \quad w = \begin{cases} 1 & \text{to Case 1A} \\ 2 & \text{to Case 1B} \end{cases} \quad (13a,b)$$

or,

$$Nu_1(Z) = \frac{1}{4} \frac{(1+\gamma)}{\gamma} \frac{dT_{av}}{dZ} \quad (\text{Case 1A}) \quad Nu_2(Z) = \frac{(1+\gamma)}{4} \frac{dT_{av}}{dZ} \quad (\text{Case 1B}) \quad (13c,d)$$

### 3. Results and discussion

Numerical results for thermally developing laminar flow of power-law fluids inside eccentric annular ducts were obtained by codes developed in the FORTRAN 90 programming language. The system given by Eqs. (11) was handled through the subroutine DIVPAG from the IMSL Library (1991) with a user-prescribed subroutine relative error target of  $10^{-8}$ . These codes were implemented on a PENTIUM – IV 1.3 GHz microcomputer and the complete solution was computed with  $NT \leq 400$  in the expansions. The results are presented in terms of dimensionless average temperatures and local Nusselt numbers along the dimensionless axial coordinate, within the range  $Z = 10^{-3}$  to 1, for different values of power-law indices, dimensionless eccentricity and aspect ratios.

Table (1) illustrates the convergence behavior of the present approach in terms of the local Nusselt number in the thermal entry region (i. e.,  $Z = 10^{-3}$ ,  $10^{-1}$  and 1) for different power-law indices and dimensionless eccentricity,  $\varepsilon = 0$ . It is observed an excellent convergence behavior, with practically three converged digits for all positions considered.

Table 1. Convergence of the local Nusselt number for an eccentric annular duct with dimensionless eccentricity  $\epsilon = 0$ .

Nu(Z)							
$\gamma$	NT	Case 1A			Case 1B		
		n = 0.5	n = 1.0	n = 1.5	n = 0.5	n = 1.0	n = 1.5
<b>Z = 0.001</b>							
0.2	25	36.190	33.814	32.954	10.201	9.9638	9.8819
	100	43.855	34.798	32.183	17.433	17.077	16.946
	255	40.141	33.676	31.832	23.077	22.450	22.212
	400	40.371	33.715	31.792	26.489	25.416	25.009
0.5	25	23.043	22.344	22.035	13.365	13.073	12.969
	100	34.203	31.453	30.180	22.846	22.215	21.958
	255	32.835	29.008	27.519	28.115	26.762	26.196
	400	31.464	28.436	27.239	29.091	27.010	26.171
0.8	25	18.830	18.346	18.149	15.794	15.429	15.291
	100	30.150	28.544	27.825	26.404	25.417	24.995
	255	31.800	27.837	27.460	30.357	28.206	27.305
	400	29.931	27.051	25.973	29.532	26.854	25.825
<b>Z = 0.1</b>							
0.2	25	10.714	10.022	9.8155	5.7154	5.5408	5.4891
	100	10.610	9.9157	9.7023	5.7034	5.4887	5.4225
	255	10.625	9.9290	9.7149	5.6843	5.4736	5.4088
	400	10.619	9.9244	9.7107	5.6798	5.4702	5.4058
0.5	25	7.5796	7.2741	7.1247	6.0831	5.8550	5.7676
	100	7.5289	7.2443	7.1025	5.9938	5.7682	5.6832
	255	7.5334	7.2479	7.1055	5.9876	5.7638	5.6794
	400	7.5311	7.2462	7.1041	5.9856	5.7623	5.6781
0.8	25	6.8205	6.5377	6.4088	6.3728	6.1141	6.0043
	100	6.7671	6.5101	6.3928	6.3005	6.0623	5.9627
	255	6.7678	6.5105	6.3935	6.3005	6.0607	5.9615
	400	6.7661	6.5097	6.3924	6.2964	6.0596	5.9605
<b>Z = 1</b>							
0.2	25	8.3698	8.1313	8.0802	4.2288	4.1945	4.2089
	100	8.3681	8.1297	8.0788	4.2287	4.1944	4.2087
	255	8.3681	8.1296	8.0787	4.2287	4.1944	4.2087
	400	8.3681	8.1296	8.0783	4.2287	4.1944	4.2087
0.5	25	5.7702	5.7385	5.7101	4.4536	4.4296	4.4292
	100	5.7696	5.7381	5.7099	4.2562	4.4293	4.4293
	255	5.7696	5.7381	5.7106	4.4535	4.4293	4.4293
	400	5.7696	5.7381	5.7100	4.4534	4.4293	4.4293
0.8	25	5.1098	5.0822	5.0641	4.7098	4.6850	4.6760
	100	5.1095	5.0821	5.0640	4.7097	4.6848	4.6760
	255	5.1095	5.0820	5.0640	4.7090	4.6850	4.6760
	400	5.1095	5.0820	5.0640	4.7097	4.6850	4.6760

Table (2) shows a comparison of the asymptotic Nusselt numbers for the case 1A against those of Manglik and Fang (2002). It can be noticed an excellent agreement among the results furnishing a direct validation of the computational code developed in the present work.

Table 2. Comparison of the asymptotic Nusselt number for an eccentric annular duct for the case 1A.

Nu							
$\gamma$	$\epsilon$	n = 0.5		n = 1		n = 1.5	
		Present	Manglik and Fang (2002)	Present	Manglik and Fang (2002)	Present	Manglik and Fang (2002)
0.2	0	8.3681	8.1093	8.1296	8.0651	8.0783	8.0151
	0.2	6.5178	6.5244	6.4796	6.4861	6.4509	6.4574
	0.6	3.9721	3.9761	3.9875	3.9915	3.9812	4.0012
0.5	0	5.7696	5.7621	5.7385	5.7316	5.7100	5.7040
	0.2	3.5630	3.5666	3.7349	3.7386	3.8073	3.8264
	0.6	2.0318	2.0338	2.1018	2.1039	2.1396	2.1504
0.8	0	5.1095	5.1051	5.0820	5.0785	5.0640	5.0604
	0.2	2.5675	2.5701	2.8312	2.8369	2.9296	2.9443
	0.6	1.2897	1.2923	1.4173	1.4216	1.4725	1.4874

Figure (2) presents the evolution of the dimensionless average temperature along the thermal entry region for a Newtonian fluid with a dimensionless eccentricity  $\varepsilon = 0$  and different aspect ratios. One can see that the aspect ratio causes an opposite behavior on this evolution; while for the case 1A the fully developed region is reached more rapidly for higher aspect ratios, for the case 1B this region is reached faster for lower values of this parameter. This can be explained by the effect of the adopted thermal boundary conditions and to a decrease in the annular gap; for the case 1A the insulated wall is the outer surface, so that the inner surface is subjected to higher heat fluxes but at lower aspect ratios, the average velocity is also lower, this way the heat exchange is less intensified and it is increased as the aspect ratio increases; while the opposite is observed for the case 1B.

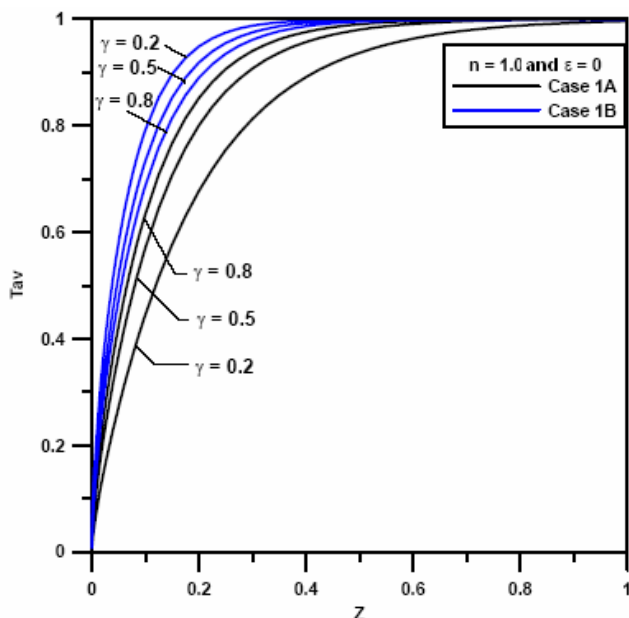


Figure 2. Evolution of the dimensionless average temperature along the thermal entry region.

Figure (3) presents the effect of the power-law index on the temperature field as a function of the normalized radial coordinate, for the two cases of thermal boundary conditions analyzed at the angular positions  $\theta = 0^\circ$  and  $180^\circ$ , and considering fixed values of axial position, aspect ratio and dimensionless eccentricity ( $Z = 1$ ,  $\gamma = 0.2$  and  $\varepsilon = 0.6$ ). It is verified that for the case 1A the power-law index does not affect the temperature field for both angular positions. This can be explained by the fact of satisfying the energy equation leading to a compensation evidenced by flattened temperature distributions, once the heat transfer process is limited by the nearest adiabatic wall. For the second case (case 1B), the power-law index slightly affects the temperature profile, for lower values of the normalized axial coordinate at  $\theta = 180^\circ$ , due to an increase of the surface area near the adiabatic wall.

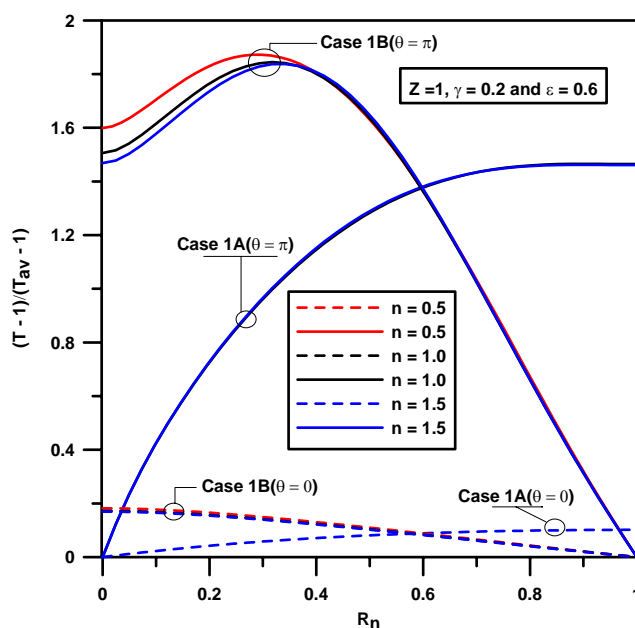


Figure 3. Effect of the power-law index on the temperature field for an eccentric annular duct.

Similarly, Fig. (4) brings the effect of the dimensionless eccentricity on the temperature field as a function of the normalized radial coordinate. It is noted that an increase of this parameter results in an increase of the stagnation zone; as a result it higher temperature gradients are experienced in this region. For the case where the adiabatic wall is far from this zone, the temperature gradient is lower, this way resulting in higher peaks of temperature for lower dimensionless eccentricities.

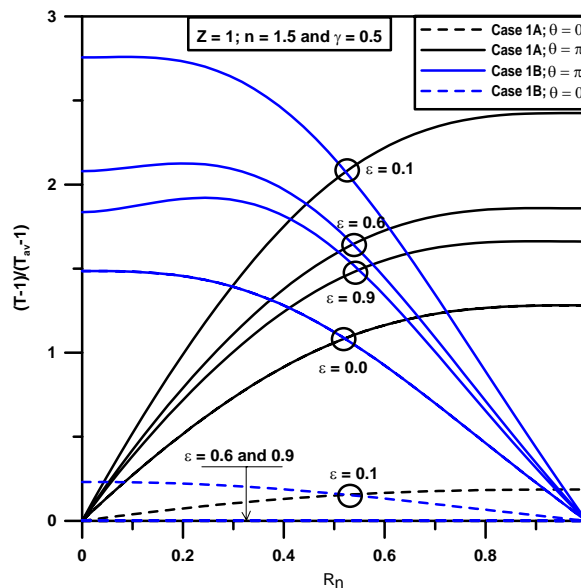


Figure 4. Effect of the dimensionless eccentricity on the temperature field for an eccentric annular duct.

Finally, Fig. (5) now considers the effect of the aspect ratio. It is evidenced that an increase of the aspect ratio results in higher peaks of temperature, as a direct consequence of a decrease on the annular gap width, so that higher temperature gradients are experienced mainly at the stagnation zones.

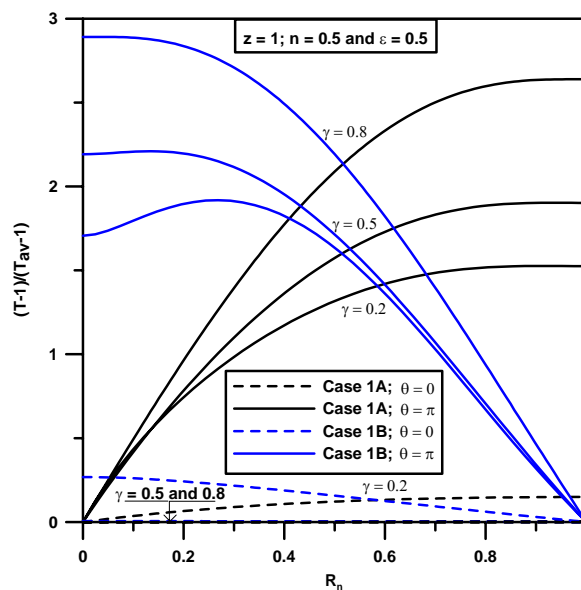


Figure 5. Effect of the aspect ratio on the temperature field for an eccentric annular duct.

#### 4. Conclusions

The present study was intended to further demonstrate the usefulness of the Generalized Integral Transform Technique (GITT) as a benchmarking and covalidation tool in the simulation of convection-diffusion problems, while the codes created with this approach were found to be relatively cost-effective, within the range of truncations orders considered. Numerical results for convection heat transfer in eccentric annular ducts were tabulated and graphically presented providing a reliable source of benchmark for the local Nusselt numbers and dimensionless average temperature.

## 5. Acknowledgement

The authors wish to acknowledge the financial support provided by CNPq and FAPERJ.

## 6. References

- Aparecido, J. B. and Cotta, R. M., 1987, "Fully Developed Laminar Flow in Trapezoidal Ducts", Proceedings of the 9th Brazilian Congress of Mechanical Engineering – COBEM 87, Vol. 1, Florianópolis, Brazil, pp. 25-28.
- Aparecido, J. B., Cotta, R. M. and Özişik, M. N., 1989, "Analytical Solutions to Two-dimensional Diffusion Type Problems in Irregular Geometries", J. Franklin Institute, Vol. 326, pp. 421-434.
- Aparecido, J. B. and Cotta, R. M., 1990, "Laminar Flow inside Hexagonal Ducts", Computational Mechanics, Vol. 6, pp. 93-100.
- Barbuto, F. A. A. and Cotta, R. M., 1997, "Integral Transformation of Elliptic Problems within Irregular Domains – Fully Developed Channel Flow", Int. J. Num. Meth. Heat & Fluid Flow, Vol. 7, pp. 778-793.
- Bird, R. B., Armstrong, R. C. and Hassager, O., 1987, "Dynamics of Polymeric Liquids", Vol. 1, 2<sup>nd</sup> Ed., John Wiley, New York.
- Chaves, C. L., Quaresma, J. N. N., Macêdo, E. N., Pereira, L. M. and Lima, J. A., 2001a, "Hydrodynamically Developed Laminar Flow of Non-Newtonian Fluids inside Triangular Ducts", Proceedings of the 16th Brazilian Congress of Mechanical Engineering, Vol. 9, Uberlândia, Brazil, pp. 105-114 (on CD-ROM).
- Chaves, C. L., Quaresma, J. N. N., Macêdo, E. N., Pereira, L. M. and Lima, J. A., 2001b, "Hydrodynamically Developed Laminar Flow of Non-Newtonian Fluids inside Double-sine Ducts", Hybrid Methods in Engineering, Vol. 3, pp. 1-14.
- Chaves, C. L., Quaresma, J. N. N., Macêdo, E. N., Pereira, L. M. and Lima, J. A., 2004, "Forced Convection Heat Transfer to Power-Law Non-Newtonian Fluids inside Triangular Ducts", Heat Transfer Eng., V. 25, pp. 23-33.
- Cotta, R. M., 1993, "Integral Transforms in Computational Heat and Fluid Flow", CRC, Boca Raton, FL, USA.
- Cotta, R. M., 1994, "Benchmark Results in Computational Heat and Fluid Flow: - The Integral Transform Method", Int. J. Heat Mass Transfer (Invited Paper), Vol. 37 (Suppl. 1), pp. 381-393.
- Cheng, K. C. and Hwang, G. J., 1968, "Laminar Forced Convection in Eccentric Annuli", AIChE J., Vol. 14, pp. 510-512.
- Escudier, M. P., Oliveira, P. J. and Pinho, F. T., 2002, "Fully Developed Laminar Flow of Purely Viscous Non-Newtonian Liquids through Annuli, Including the Effects of Eccentricity and Inner-cylinder Rotation", Int. J. Heat Fluid Flow, Vol. 23, pp. 52-73.
- Fang, P. P., Manglik, R. M. and Jog, M. A., 1999, "Characteristics of Laminar Viscous Shear-thinning Fluid Flows in Eccentric Annular Ducts", J. Non-Newtonian Fluid Mech., Vol. 84, pp. 1-17.
- IMSL Library, MATH/LIB, Houston, TX, 1991.
- Jonsson, V. K. and Sparrow, E. M., 1965, "Results of Laminar Flow Analysis and Turbulent Flow Experiments for Eccentric Annular Duct", AIChE J., Vol. 11, pp. 1143-1145.
- Manglik, R. M. and Fang, P. P., 1995, "Effect of Eccentricity and Thermal Boundary Conditions on Laminar Fully Developed Flow in Annular Ducts", Int. J. Heat Fluid Flow, Vol. 16, pp. 298-306.
- Manglik, R. M. and Fang, P. P., 2002, "Thermal Processing of Viscous Non-Newtonian Fluids in Annular Ducts: Effects of Power-law Rheology, Duct Eccentricity, and Thermal Boundary Conditions", Int. J. Heat Mass Transfer, Vol. 45, pp. 803-814.
- Mikhailov, M.D. and Özişik, M.N., 1984, "Unified Analysis and Solutions of Heat and Mass Diffusion", John Wiley, New York, USA.
- Monteiro, E. R., Macêdo, E. N., Quaresma, J. N. N. and Cotta, R. M., 2004, "A Solution through Integral Transforms for Fully Developed Flow in Doubly Connected Ducts", Proceedings of the 10th Brazilian Congress of Thermal Sciences and Engineering -- ENCIT 2004, Rio de Janeiro, Brazil, paper # CIT04-0528 (on CD-ROM).
- Piercy, N. A. V., Hooper, M. S. and Winny, H. F., 1933, "Viscous Flow through Pipes with Cores", London Edinburgh Dublin Philos. Mag. J. Sci., Vol. 15, pp. 647-676.
- Shah, R. K. and London, A. L., 1978, "Laminar Flow Forced Convection in Ducts", In Advances in Heat Transfer (Supplement 1), New York, Academic Press.
- Snyder, W. T. and Goldstein, G. A., 1965, "An Analysis of Fully Developed Laminar Flow in an Eccentric Annulus", AIChE J., Vol. 11, pp. 462-467.
- Stevenson, C., 1949, "The Centre of Flexure of a Hollow Shaft", Proc. London Math. Soc., Vol 50, pp. 536.
- Suzuki, K., Szmyd, J. S. and Ohtsuk, H., 1991, "Laminar Forced Convection Heat Transfer in Eccentric Annuli", Heat Transfer-Jpn. Res., Vol. 20, pp. 169-183.
- Trombetta, M. L., 1971, "Laminar Forced Convection in Eccentric Annuli", Int. J. Heat Mass Transfer, Vol 14, pp. 1161-1173.

## 7. Copyright Notice

The author is the only responsible for the printed material included in his paper.

## ANALYSIS OF HYPERBOLIC HEAT CONDUCTION IN FINS THROUGH THE GENERALIZED INTEGRAL TRANSFORM TECHNIQUE

Carlos Alexandre Moreira da Silva

Emanuel Negrão Macêdo

João Nazareno Nonato Quaresma

Chemical and Food Engineering Department, CT, Universidade Federal do Pará, UFPA  
Campus Universitário do Guamá, 66075-110, Belém, PA, Brazil; quaresma@ufpa.br

Renato Machado Cotta

Mechanical Engineering Department – POLI/COPPE, Universidade Federal do Rio de Janeiro, UFRJ  
Cx. Postal 68503 – Cidade Universitária, 21945-970, Rio de Janeiro, RJ, Brazil; cotta@serv.com.ufrj.br

**Abstract.** *The Generalized Integral Transform Technique (GITT) is employed in the hybrid solution of the energy equation to describe the behavior of a rectangular fin undergoing non-Fourier heat conduction. The employment of the GITT approach in the hyperbolic heat conduction equation leads to a coupled system of second order ordinary differential equations in time. The resulting system is then numerically solved by Gear's method for stiff problems, available in the subroutine DIVPAG from the IMSL Library. Numerical results for the temperature field are computed for different values of the governing parameters and dimensionless thermal relaxation times, which are then compared with results previously reported in the literature for special cases.*

**Keywords.** *Hyperbolic heat conduction, Rectangular fins, Integral transforms, Hybrid methods.*

### 1. Introduction

Non-Fourier hyperbolic models have been proposed to investigate conduction heat transfer phenomenon in a few special applications, such as in some situations related to combustion engines, pulsating laser heating, rapidly contacting surfaces in electronic devices and heat transfer in nanosystems. For these situations, the classical model based on the Fourier law is not adequate to describe the physical mechanism of heat conduction, in which the speed of propagation of thermal waves is infinite, this way leading to a non-realistic notion of energy diffusion. The technological advances in the micro- and nano-fabrication fields, including the construction of micro heat exchange devices, has contributed to the appearance of investigations towards the evaluation of performance and efficiency of fins, when heat propagates in extremely short time intervals with a finite speed of propagation. In order to take into account the effect of finite speed thermal transfers, Cattaneo (1958) and Vernotte (1958) independently modified the Fourier model to include a relaxation time that reflects the temporary delay of the thermal wave. In their models, the Fourier law is a particular case when the relaxation time is zero, and thus an infinite speed of thermal wave propagation is achieved, as foreseen by this classical theory.

Many researchers have devoted attention to heat transfer in fast processes along the years. More recently, Tsai et al. (2005) analyzed the non-Fourier effect on the thermal behavior of spherical media (hollow and bi-layered composite spheres) subjected to suddenly changes of the surface temperature. These authors focused their analysis in using the Laplace transform in conjunction with the Riemann-sum approximation method to solve the related energy equation and to investigate the influence of parameters such as relaxation time, temperature and diffusivities ratio. Quaresma et al. (2001), Cruz et al. (2001) and Macêdo et al. (2005) solved the hyperbolic heat conduction problem for a slab subjected to various prescribed heat flux forms at one of the boundaries. Different methodologies were employed to solve the hyperbolic heat conduction equation, respectively, Laplace Transforms with numerical inversion (Quaresma et al., 2001), Finite Volume Method with Gear Approach (Cruz et al., 2001) and the Generalized Integral Transform Technique (Macêdo et al., 2005). Numerical results for the temperature field were presented in order to analyze the influence of the governing parameters (relaxation times and Biot numbers) on this physical problem. In dealing with heat flow in fins, Lin (1998) studied the effect of the relaxation time on the performance of a convective fin of constant cross-sectional area subjected to periodic thermal conditions, by employing a hybrid scheme involving the Laplace transform and the finite volume method.

In this context, the main objective of the present work is to develop a hybrid numerical-analytical solution based on the Generalized Integral Transform Technique (GITT) (Cotta, 1993; Cotta, 1994; Cotta & Mikhailov, 1997; Cotta & Mikhailov, 2006) to analyze the non-Fourier thermal response of a fin of constant cross-sectional area, such as the one studied by Lin (1998). The employment of the GITT approach to solve the related hyperbolic partial differential equation produces a fast and efficient solution, with a considerable analytical involvement, but presenting some advantages when compared to purely numerical schemes. The characteristic of automatic global error control inherent to this technique allows for the computation of benchmark results as well. In this case, the integral transformation of the hyperbolic partial differential equation yields an infinite system of coupled ordinary differential equations of second order, which is then solved through well-established routines appropriate for handling initial value problems with stiff characteristics, such as the DIVPAG routine from the IMSL Library (1991). Numerical results are then produced for the temperature field within representative ranges of the governing parameters, and critically compared with those previously presented in the literature.

## 2. Analysis

One-dimensional hyperbolic heat conduction in a fin of constant cross-sectional area is considered, initially at the uniform temperature  $T_i$ , assuming constant thermophysical properties  $k$  and  $\alpha$ , and no internal heat generation. The fin tip is maintained insulated, while its larger surfaces are exchanging heat by convection with a fluid kept at a constant temperature,  $T_\infty$  and uniform heat transfer coefficient,  $h$ . The fin base is subjected to a periodic temperature variation in the form:

$$T_b(t) = \bar{T}_b + (\bar{T}_b - T_i)A \cos(\hat{\omega}t), \quad t > 0 \quad (1)$$

where  $\bar{T}_b$  is the average fin base temperature,  $A$  is the dimensionless amplitude of base temperature difference oscillation and  $\hat{\omega}$  is the frequency of base temperature oscillation. The heat transfer by radiation is neglected and the heat transfer coefficient,  $h$ , is varying with the spatial coordinate as:

$$h(x) = h_0 H(x/L) \quad (2)$$

where  $h_0$  is a reference heat transfer coefficient, which is taken as  $h_0 = bk/(2L^2)$  and  $H$  is a function only of the spatial coordinate.

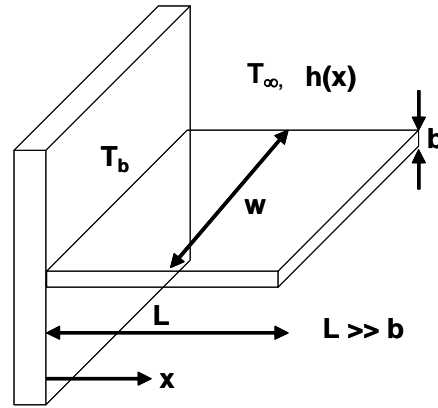


Figure 1. Schematic representation of the fin geometry.

The one-dimensional energy equation for the fin shown in Fig. (1) is written in the following form:

$$\rho c_p \frac{\partial T(x,t)}{\partial t} = -\frac{\partial q''(x,t)}{\partial x} - 2 \frac{h(x)}{b} [T(x,t) - T_\infty] \quad (3)$$

For the phenomenon involving finite speed of propagation of the thermal waves, the classical Fourier model must be modified. Therefore, Cattaneo (1958) and Vernotte (1958) independently derived a model of heat flow in the form:

$$\tau_r \frac{\partial q''(x,t)}{\partial t} + q''(x,t) = -k \frac{\partial T(x,t)}{\partial x} \quad (4)$$

where  $\tau_r$  is the relaxation time of the material in which the heat conduction process is occurring. Combining Eq. (3) with Eq. (4), it then results the partial differential equation that governs the hyperbolic heat conduction in a convective fin, as:

$$\rho c_p \tau_r \frac{\partial^2 T(x,t)}{\partial t^2} + \rho c_p \frac{\partial T(x,t)}{\partial t} + 2 \tau_r \frac{h(x)}{b} \frac{\partial [T(x,t) - T_\infty]}{\partial t} = k \frac{\partial^2 T(x,t)}{\partial x^2} - 2 \frac{h(x)}{b} [T(x,t) - T_\infty] \quad (5)$$

Equation (5) describes the heat propagation in a convective fin with speed  $v = (\alpha/\tau_r)^{1/2}$ . Equation (5) is then written in dimensionless form as:

$$\beta \frac{\partial^2 \theta(\eta, \xi)}{\partial \xi^2} + [1 + \beta H(\eta)] \frac{\partial \theta(\eta, \xi)}{\partial \xi} = \frac{\partial^2 \theta(\eta, \xi)}{\partial \eta^2} - H(\eta) \theta(\eta, \xi) + H(\eta) \theta_e \quad (6a)$$

subjected to the following initial and boundary conditions:

$$\theta(\eta, 0) = 0; \quad \frac{\partial \theta(\eta, 0)}{\partial \xi} = 0, \quad 0 \leq \eta \leq 1 \quad (6b,c)$$

$$\theta(0, \xi) = 1 + A \cos(\omega \xi); \quad \frac{\partial \theta(1, \xi)}{\partial \eta} = 0, \quad \xi > 0 \quad (6d,e)$$

The following dimensionless groups were employed in obtaining Eqs. (6):

$$\eta = x/L; \quad \xi = \alpha t/L^2; \quad \beta = \alpha \tau_r/L^2; \quad \omega = \hat{\omega} L^2/\alpha; \quad \theta(\eta, \xi) = [T(x, t) - T_1]/[\bar{T}_b - T_1]; \quad \theta_e = [T_\infty - T_1]/[\bar{T}_b - T_1] \quad (7)$$

Equations (6) constitute a non-homogeneous problem, which have to be filtered in order to obtain a better computational performance in the integral transform method. For this purpose, the dimensionless temperature  $\theta(\eta, \xi)$  is written in a separated form as follows:

$$\theta(\eta, \xi) = \theta_f(\eta; \xi) + \theta_h(\eta, \xi) \quad (8)$$

Introducing Eq. (8) into Eqs. (6), one obtains the following equations for calculating the filtered potential  $\theta_f(\eta, \xi)$ :

$$\frac{\partial^2 \theta_f(\eta; \xi)}{\partial \eta^2} + H(\eta) \theta_e = 0 \quad (9a)$$

$$\theta_f(0; \xi) = 1 + A \cos(\omega \xi), \quad \xi > 0 \quad (9b)$$

$$\frac{\partial \theta_f(1; \xi)}{\partial \eta} = 0, \quad \xi > 0 \quad (9c)$$

Equation (9a) is readily integrated to yield

$$\theta_f(\eta; \xi) = 1 + A \cos(\omega \xi) + \theta_e \int_0^\eta \int_{\eta'}^1 H(\eta'') d\eta'' d\eta' \quad (9d)$$

For the particular case where  $H(\eta) = e^\eta$  (the functional form adopted in the illustration of results), Eq. (9d) becomes

$$\theta_f(\eta; \xi) = 1 + A \cos(\omega \xi) + \theta_e (1 + e\eta - e^\eta) \quad (9e)$$

Then, the partial differential equation for the filtered potential  $\theta_h(\eta, \xi)$  is obtained as

$$\beta \frac{\partial^2 \theta_h(\eta, \xi)}{\partial \xi^2} + [1 + \beta H(\eta)] \frac{\partial \theta_h(\eta, \xi)}{\partial \xi} = \frac{\partial^2 \theta_h(\eta, \xi)}{\partial \eta^2} - H(\eta) \theta_h(\eta, \xi) - \left[ \beta \frac{\partial^2 \theta_f(\eta; \xi)}{\partial \xi^2} + [1 + \beta H(\eta)] \frac{\partial \theta_f(\eta; \xi)}{\partial \xi} + H(\eta) \theta_f(\eta; \xi) \right] \quad (10a)$$

$$\theta_h(\eta, 0) = -\theta_f(\eta; 0) = -[1 + A + \theta_e (1 + e\eta - e^\eta)], \quad 0 \leq \eta \leq 1 \quad (10b)$$

$$\frac{\partial \theta_h(\eta, 0)}{\partial \xi} = -\frac{\partial \theta_f(\eta; 0)}{\partial \xi} = 0, \quad 0 \leq \eta \leq 1 \quad (10c)$$

$$\theta_h(0, \xi) = 0, \quad \xi > 0 \quad (10d)$$

$$\frac{\partial \theta_h(1, \xi)}{\partial \eta} = 0, \quad \xi > 0 \quad (10e)$$

## 2.1. Solution methodology

The next step is to find a solution for the potential  $\theta_h(\eta, \xi)$ , and for this purpose we follow the ideas in the GITT (Cotta, 1993; Cotta, 1994; Cotta & Mikhailov, 1997; Cotta & Mikhailov, 2006), first by selecting an appropriate auxiliary eigenvalue problem, which shall provide the basis for the eigenfunction expansion. Therefore, the following simple eigenvalue problem is here proposed:

$$\frac{d^2 \psi_i(\eta)}{d\eta^2} + \mu_i^2 \psi_i(\eta) = 0, \quad \text{in } 0 < \eta < 1 \quad (11a)$$

$$\psi_i(0) = 0; \quad \frac{d\psi_i(1)}{d\eta} = 0 \quad (11b,c)$$

Equations (11) can be analytically solved to yield, respectively, the eigenfunctions and eigenvalues as

$$\psi_i(\eta) = \sin(\mu_i \eta); \quad \mu_i = (2i - 1)\pi/2, \quad i = 1, 2, 3, \dots \quad (12a,b)$$

It can be shown that the eigenfunctions  $\psi_i(\eta)$  present the following orthogonality property:

$$\int_0^1 \psi_i(\eta) \psi_j(\eta) d\eta = \begin{cases} 0, & i \neq j \\ N_i, & i = j \end{cases} \quad (12c,d)$$



where  $N_i$  is the normalization integral. Also, we define  $\tilde{\psi}_i(\eta)$  as being the normalized eigenfunctions.  $N_i$  and  $\tilde{\psi}_i(\eta)$  are computed respectively as

$$N_i = \int_0^1 \psi_i^2(\eta) d\eta = 1/2; \quad \tilde{\psi}_i(\eta) = \psi_i(\eta) / \sqrt{N_i} \quad (12e,f)$$

Equations (11) and (12) together with the above properties allow the definition of the integral transform pair for the potential  $\theta_h(\eta, \xi)$  as:

$$\bar{\theta}_{h,i}(\xi) = \int_0^1 \tilde{\psi}_i(\eta) \theta_h(\eta, \xi) d\eta, \quad \text{transform} \quad (13a)$$

$$\theta_h(\eta, \xi) = \sum_{i=1}^{\infty} \tilde{\psi}_i(\eta) \bar{\theta}_{h,i}(\xi), \quad \text{inverse} \quad (13b)$$

To obtain the resulting system of differential equations for the transformed potentials  $\bar{\theta}_{h,i}(\xi)$ , the partial differential equation (10a) is multiplied by  $\tilde{\psi}_i(\eta)$ , integrated over the domain  $[0,1]$  in the  $\eta$ -direction, and the inverse formula, Eq. (13b), is employed in place of the potential  $\theta_h(\eta, \xi)$ , resulting in the following transformed ordinary differential system:

$$\frac{d^2 \bar{\theta}_{h,i}(\xi)}{d\xi^2} + \sum_{j=1}^{\infty} A_{ij} \frac{d \bar{\theta}_{h,j}(\xi)}{d\xi} + \sum_{j=1}^{\infty} B_{ij} \bar{\theta}_{h,j}(\xi) = \bar{g}_i(\xi), \quad i = 1, 2, 3, \dots \quad (14a)$$

The same operation can be performed over the initial conditions given by Eqs. (10b,c), to furnish

$$\bar{\theta}_{h,i}(0) = \bar{f}_i; \quad \frac{d \bar{\theta}_{h,i}(0)}{d\xi} = 0 \quad (14b,c)$$

where the coefficients in Eqs. (14) are defined as follows:

$$A_{ij} = \frac{1}{\beta} \delta_{ij} + \int_0^1 H(\eta) \tilde{\psi}_i(\eta) \tilde{\psi}_j(\eta) d\eta; \quad B_{ij} = \frac{1}{\beta} \left[ \delta_{ij} \mu_i^2 + \int_0^1 H(\eta) \tilde{\psi}_i(\eta) \tilde{\psi}_j(\eta) d\eta \right]; \quad \bar{g}_i(\xi) = \bar{g}_{i,1} \cos(\omega \xi) + \bar{g}_{i,2} \sin(\omega \xi) + \bar{g}_{i,3} \quad (15a-c)$$

$$\bar{g}_{i,1} = A \int_0^1 \tilde{\psi}_i(\eta) [\omega^2 - H(\eta) / \beta] d\eta; \quad \bar{g}_{i,2} = A \omega \int_0^1 \tilde{\psi}_i(\eta) [1 / \beta + H(\eta)] d\eta; \quad \bar{g}_{i,3} = -(1 / \beta) \int_0^1 H(\eta) \tilde{\psi}_i(\eta) [1 + \theta_c(1 + e\eta - e^\eta)] d\eta \quad (15d-f)$$

$$\bar{f}_i = - \int_0^1 \tilde{\psi}_i(\eta) [1 + A + \theta_c(1 + e\eta - e^\eta)] d\eta; \quad \delta_{ij} = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases} \quad (15g,h)$$

The coefficients may be analytically obtained through symbolic manipulation platforms such as the *Mathematica* system (Wolfram, 1999). Equations (14) form an infinite linear initial value problem, which has to be truncated in a sufficiently high order  $N$ , in order to compute the transformed potentials,  $\bar{\theta}_{h,i}(\xi)$ , to within an user prescribed accuracy target. For the solution of such a system, due to its expected stiff characteristics, specialized subroutines have to be employed, such as the subroutine DIVPAG from the IMSL Library (1991). This subroutine provides the important feature of automatic controlling the relative error over the solution of the ordinary differential equations system, allowing the user to establish error targets for the transformed potentials. Once this system is solved for the transformed potentials, the inverse formula, Eq. (13b), is recalled to provide the potential  $\theta_h(\eta, \xi)$ , which is added to the filtering potential  $\theta_f(\eta; \xi)$ , given by Eq. (9e), to furnish the complete temperature field.

In the realm of applications, one might be interested in approximate analytical solutions of the problem here proposed. In order to obtain a fully analytical solution for the system (14), we might recall the so-called lowest order solution of Eqs. (14) (Cotta, 1993), which accounts only for the diagonal terms in the coupling coefficients matrices above. In this way the system becomes decoupled and is rewritten as:

$$\frac{d^2 \bar{\theta}_{h,i}(\xi)}{d\xi^2} + A_{ii} \frac{d \bar{\theta}_{h,i}(\xi)}{d\xi} + B_{ii} \bar{\theta}_{h,i}(\xi) = \bar{g}_i(\xi), \quad i = 1, 2, 3, \dots \quad (16a)$$

$$\bar{\theta}_{h,i}(0) = \bar{f}_i; \quad \frac{d \bar{\theta}_{h,i}(0)}{d\xi} = 0 \quad (16b,c)$$

In order to express the analytical solution of system (16), one considers three different situations, namely:

- Case 1:  $A_{ii}^2 > 4B_{ii}$  :

$$\bar{\theta}_{hl,i}(\xi) = e^{-\frac{A_{ii}\xi}{2}} [C_1 \cosh(\arg \xi) + C_2 \sinh(\arg \xi)] + C_3 \cos(\omega\xi) + C_4 \sin(\omega\xi) + C_5 \quad (17a)$$

$$C_1 = \bar{f}_i - (C_3 + C_5) \quad (17b)$$

$$C_2 = \frac{\frac{A_{ii}}{2} [\bar{f}_i - (C_3 + C_5)] - \omega C_4}{\arg} \quad (17c)$$

- Case 2:  $A_{ii}^2 < 4B_{ii}$  :

$$\bar{\theta}_{hl,i}(\xi) = e^{-\frac{A_{ii}\xi}{2}} [C_1 \cos(\arg \xi) + C_2 \sin(\arg \xi)] + C_3 \cos(\omega\xi) + C_4 \sin(\omega\xi) + C_5 \quad (18a)$$

$$C_1 = \bar{f}_i - (C_3 + C_5) \quad (18b)$$

$$C_2 = \frac{\frac{A_{ii}}{2} [\bar{f}_i - (C_3 + C_5)] - \omega C_4}{\arg} \quad (18c)$$

- Case 3:  $A_{ii}^2 = 4B_{ii}$  :

$$\bar{\theta}_{hl,i}(\xi) = e^{-\frac{A_{ii}\xi}{2}} [C_1 + C_2\xi] + C_3 \cos(\omega\xi) + C_4 \sin(\omega\xi) + C_5 \quad (19a)$$

$$C_1 = \bar{f}_i - (C_3 + C_5) \quad (19b)$$

$$C_2 = \frac{\frac{A_{ii}}{2} [\bar{f}_i - (C_3 + C_5)] - \omega C_4}{2} \quad (19c)$$

where,

$$\arg = \frac{1}{2} \sqrt{|A_{ii}^2 - 4B_{ii}|} \quad (20a)$$

$$C_3 = \frac{(B_{ii} - \omega^2)\bar{g}_{i,1} - \omega A_{ii}\bar{g}_{i,2}}{(B_{ii} - \omega^2)^2 + \omega^2 A_{ii}^2} \quad (20b)$$

$$C_4 = \frac{\omega A_{ii}\bar{g}_{i,1} + (B_{ii} - \omega^2)\bar{g}_{i,2}}{(B_{ii} - \omega^2)^2 + \omega^2 A_{ii}^2} \quad (20c)$$

$$C_5 = \frac{\bar{g}_{i,3}}{B_{ii}} \quad (20d)$$

The solutions for each case given by Eqs. (17) to (20) are introduced into the inverse formula (13b), so that the solution for the potential  $\theta_i(\eta, \xi)$  is then completed. The potential  $\theta_f(\eta; \xi)$ , given by Eq. (9e), is then added to the three respective cases above to furnish the approximate analytical temperature field.

### 3. Results and discussion

Numerical results for the temperature field were obtained from a code developed in the FORTRAN 90/95 programming language. The complete solution was computed using up to five hundred terms ( $N \leq 500$ ) in the eigenfunction expansion, and all the results were obtained with  $A = 0.5$ ,  $\theta_e = 0.1$  and taking  $H(\eta) = e^\eta$ , as well as different values of the governing parameters  $\beta$  and  $\omega$ .

Tables (1) to (4) show the convergence behavior of the temperature distribution along the fin length with different dimensionless relaxation times, and fixed values of dimensionless frequency of base temperature oscillation and dimensionless time. The columns represent the lowest-order solution for the temperature field, with different truncation orders,  $N$ , which demonstrate an excellent convergence rate even for  $N = 50$  terms for dimensionless relaxation times  $\beta = 0$  and  $\beta = 0.1$ , as seen in Tables (1) and (2). For higher values of the dimensionless relaxation time, as shown in Tables (3) and (4), the solution yields slower convergence rates evidenced by numerical oscillations, and as a consequence, the complete convergence is reached only with higher truncation orders.

Table 1. Convergence behavior of the temperature field along the fin length for a fixed dimensionless time and  $\beta = 0$ .

$\eta/N$	50	100	200	300	400	500
$\beta = 0, \omega = 1, \xi = 0.5$						
0.45	0.90153	0.90153	0.90153	0.90153	0.90153	0.90153
0.50	0.85589	0.85589	0.85589	0.85589	0.85589	0.85589
0.55	0.81340	0.81340	0.81340	0.81340	0.81340	0.81340
0.60	0.77423	0.77423	0.77423	0.77423	0.77423	0.77423
0.65	0.73858	0.73858	0.73858	0.73858	0.73858	0.73858
0.70	0.70667	0.70667	0.70667	0.70667	0.70667	0.70667
0.75	0.67875	0.67875	0.67875	0.67875	0.67875	0.67875
0.80	0.65511	0.65511	0.65511	0.65511	0.65511	0.65511
0.85	0.63605	0.63605	0.63605	0.63605	0.63605	0.63605
0.90	0.62194	0.62194	0.62194	0.62194	0.62194	0.62194

Table 2. Convergence behavior of the temperature field along the fin length for a fixed dimensionless time and  $\beta = 0.1$ .

$\eta/N$	50	100	200	300	400	500
$\beta = 0.1, \omega = 1, \xi = 0.5$						
0.45	0.91154	0.91401	0.91032	0.91186	0.91128	0.91129
0.50	0.86540	0.86656	0.86720	0.86745	0.86759	0.86768
0.55	0.82715	0.82777	0.82677	0.82718	0.82703	0.82702
0.60	0.79093	0.78923	0.78956	0.78969	0.78976	0.78981
0.65	0.75591	0.75642	0.75578	0.75604	0.75594	0.75594
0.70	0.72491	0.72536	0.72560	0.72569	0.72574	0.72578
0.75	0.69941	0.69974	0.69923	0.69944	0.69936	0.69936
0.80	0.67777	0.67671	0.67692	0.67700	0.67704	0.67707
0.85	0.65906	0.65940	0.65894	0.65913	0.65906	0.65906
0.90	0.64509	0.64544	0.64562	0.64570	0.64574	0.64576

Table 3. Convergence behavior of the temperature field along the fin length for a fixed dimensionless time and  $\beta = 1$ .

$\eta/N$	50	100	200	300	400	500
$\beta = 1, \omega = 1, \xi = 0.5$						
0.45	0.85316	0.87114	0.84997	0.86179	0.85351	0.85990
0.50	0.42152	0.41924	0.41810	0.41772	0.41753	0.41741
0.55	0.03323	0.01332	0.03815	0.02430	0.03400	0.02652
0.60	0.00451	0.03127	0.02680	0.02531	0.02457	0.02412
0.65	0.01418	0.00811	0.01775	0.01239	0.01614	0.01325
0.70	0.01710	0.01192	0.00933	0.00847	0.00803	0.00777
0.75	0	0	0	0	0	0
0.80	0	0	0	0	0	0
0.85	0	0	0	0	0	0
0.90	0	0	0	0	0	0

Table 4. Convergence behavior of the temperature field along the fin length for a fixed dimensionless time and  $\beta = 5$ .

$\eta/N$	50	100	200	300	400	500
$\beta = 5.0, \omega = 1, \xi = 0.5$						
0.15	1.09470	1.11360	1.11190	1.10550	1.10930	1.11040
0.20	1.06010	0.97408	1.00490	1.00630	0.99611	0.98812
0.25	-0.03309	-0.00131	-0.00178	0.02307	0.00957	0.00476
0.30	0.03372	0.01623	0.00584	0.00589	0.00989	0.01286
0.35	0.01637	0.00600	0.00686	0.01290	0.00947	0.00835
0.40	-0.00292	0.01214	0.00681	0.00681	0.00880	0.01029
0.45	0.00536	0.00572	0.00633	0.01015	0.00797	0.00726
0.50	0.01590	0.00940	0.00556	0.00555	0.00698	0.00805
0.55	0.00766	0.00410	0.00458	0.00753	0.00585	0.00530
0.60	-0.00235	0.00654	0.00341	0.00340	0.00456	0.00543

Figure (2) analyzes the influence of the dimensionless relaxation time on the temperature field along the fin length. Also in this figure is offered a comparison with the results of Lin (1998). The governing parameters utilized were  $\omega = 1.0$ ,  $\xi = 0.5$  and  $\beta = 0, 0.1, 1$  and  $5$ . As can be seen from this figure, for lower values of dimensionless relaxation time, the thermal wave propagates very fast, as a result of the speed of propagation approaching infinity, on the other hand, for higher values of  $\beta$ , the thermal wave does not reach the fin tip, and extinguishes at approximately  $\eta = 0.55$  for  $\beta = 1.0$ , and at  $\eta = 0.24$  for  $\beta = 5.0$ . This figure also illustrates the good agreement between the present results with those of Lin (1998).

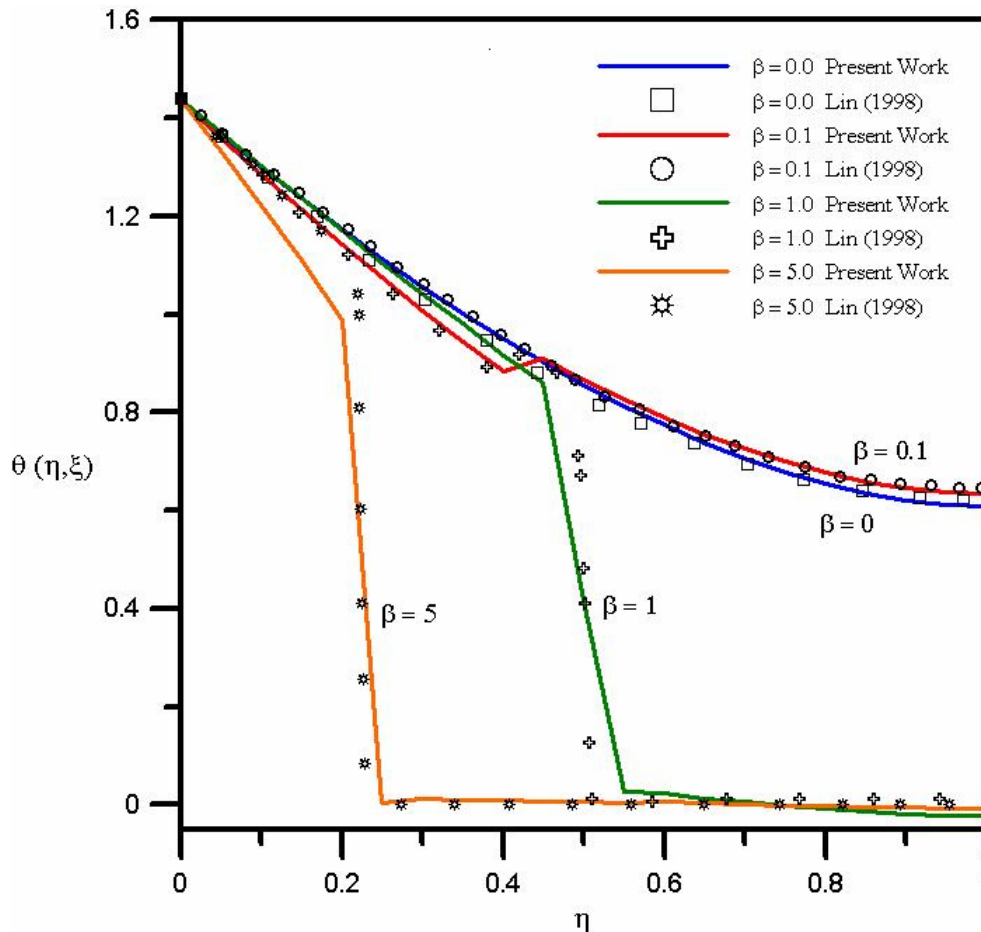


Figure 2. Temperature distribution along the fin length for different dimensionless relaxation times:  $\omega = 1.0$  and  $\xi = 0.5$ .

Figure (3) shows the temperature distribution along the fin length with different dimensionless frequencies of base temperature oscillation,  $\omega$ . Once again, it is offered a comparison of the present results with those of Lin (1998), which shows a good agreement. This figure makes it evident that even increasing the parameter  $\omega$ , the thermal wave remains extinguished at  $\eta = 0.55$  for  $\beta = 1.0$ , i.e., the dimensionless frequency of base temperature oscillation does not affect the extent of the non-Fourier effect along the fin length. The good agreement between the present results with those of Lin (1998) demonstrates that the lowest-order solution can give reasonable estimates of the hyperbolic conduction effects caused by the non-Fourier model, offering a straightforward analytical solution representation of this complex formulation.

Figure (4) investigates the effect of the dimensionless relaxation time on the temperature field at the fin tip while varying the dimensionless time. This figure demonstrates that for lower values of  $\beta$ , the fin tip temperature rapidly increases, as expected, because these values of  $\beta$  agree with the classical Fourier model, in which the thermal waves are propagated with a infinite speed. For higher values of dimensionless relaxation time, the fin tip is not immediately excited by the application of the thermal pulse. It is observed that the temperature gradually increases after a short time period, and the phase shift that appears among the curves is influenced by the dimensionless relaxation time, once the heat flow does not begin right afterwards, there is an interval of time between the application of the temperature gradient and the heat flow, which is exactly the period of storage of energy. Also, the comparison with the results of Lin (1998) again reconfirms the good agreement, although one may observe deviations, especially for higher values of  $\beta$ . However, for most practical purposes, the lowest-order solution can be employed as a good approximation of the hyperbolic heat conduction behavior.

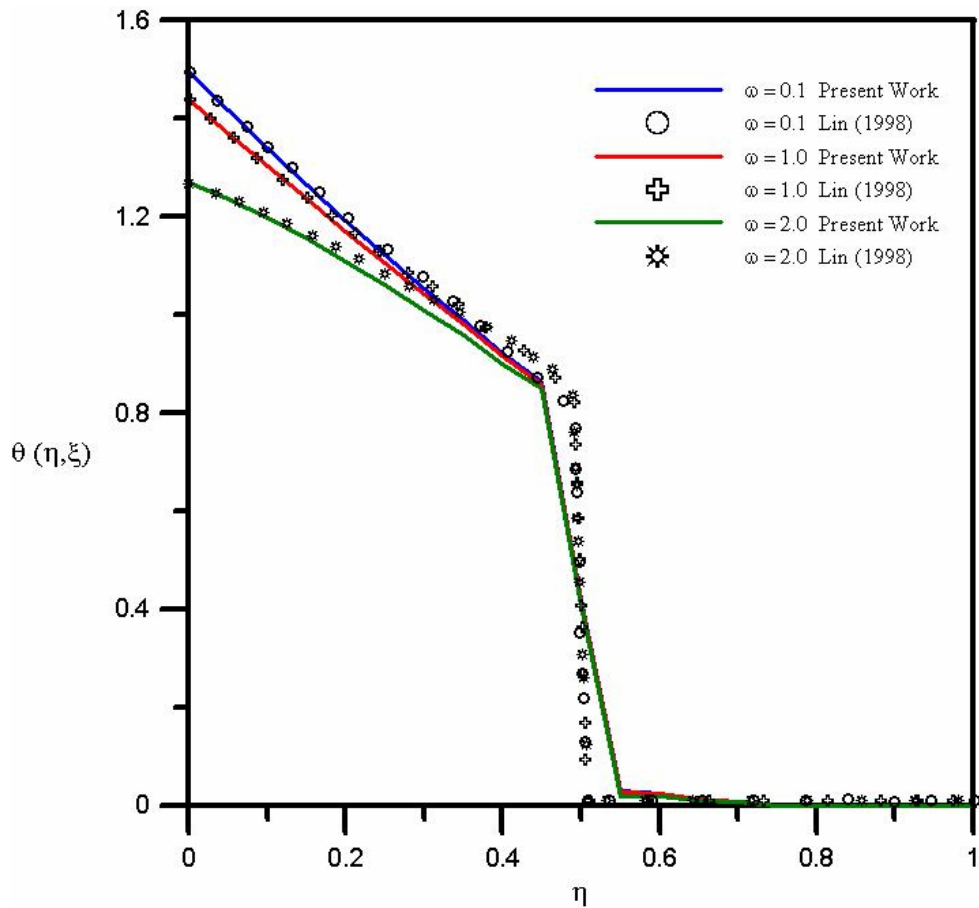


Figure 3. Temperature distribution along the fin length for different dimensionless frequencies of base temperature oscillation:  $\beta = 1.0$  and  $\xi = 0.5$ .

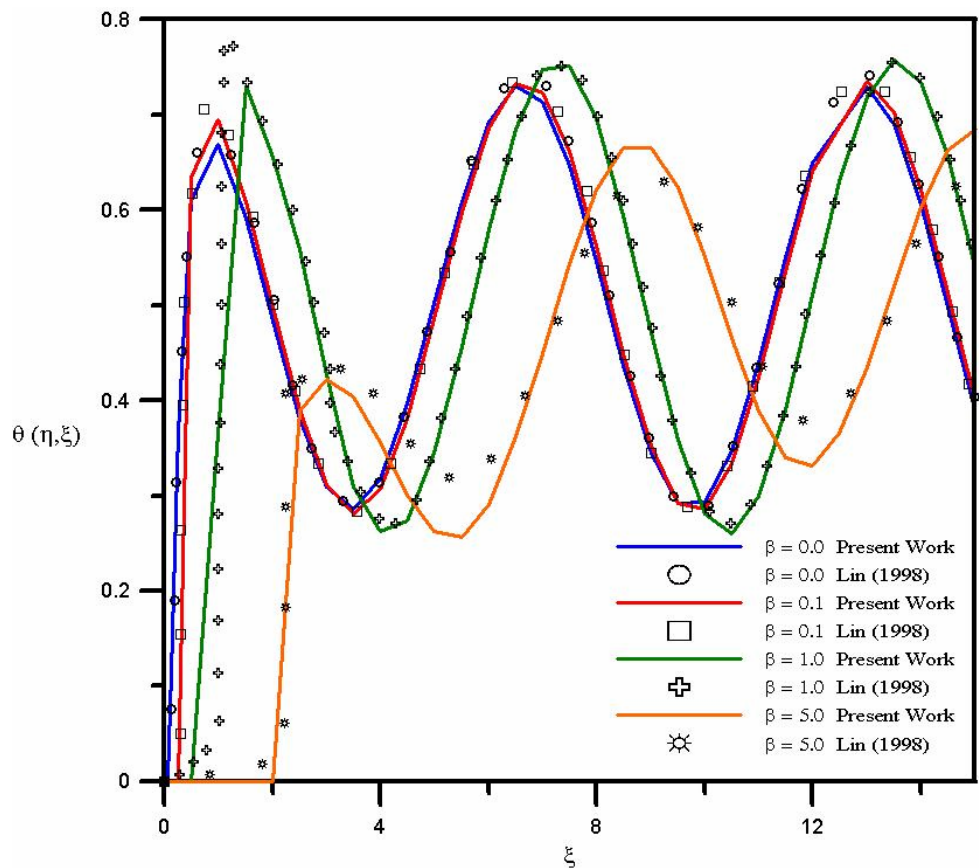


Figure 4. Temperature evolution at the fin tip for different dimensionless relaxation times ( $\omega = 1.0$ ).

#### 4. Conclusions

Hyperbolic heat conduction for a convective fin of constant cross-section area, submitted to periodic thermal conditions, has been analyzed using the Generalized Integral Transform Technique (GITT). A lowest-order solution was obtained in analytic form, offering approximate results for the fin temperature distribution for the investigated ranges of the governing parameters. It could be verified that the non-Fourier effect is significant for short times in the rectangular fin analysis, during the initial transient, when the thermal wave fronts are observed.

#### 5. Acknowledgments

The authors would like to acknowledge the financial support provided by CNPq and FAPERJ.

#### 6. References

- Cattaneo, C., 1958, "Sur une Forme de L'équation de la Chaleur Eliminant le Paradoxe d'une Propagation Instantanée", Comptes Rendus Hebdomadaires des Seances de L'Academie des Sciences, Vol. 247, pp. 431-433.
- Cotta, R.M., 1993, "Integral Transforms in Computational Heat and Fluid Flow", CRC Press, Boca Raton, USA.
- Cotta, R. M., 1994, "Benchmark Results in Computational Heat and Fluid Flow: - The Integral Transform Method", Int. J. Heat Mass Transfer (Invited Paper), Vol. 37 (Suppl. 1), pp. 381-393.
- Cotta, R.M. and Mikhailov, M.D., 1997, "Heat Conduction: Lumped Analysis, Integral Transforms, Symbolic Computation", Wiley-Interscience, USA.
- Cotta, R.M., and Mikhailov, M.D., 2006, "Hybrid Methods and Symbolic Computations", in: Handbook of Numerical Heat Transfer, 2<sup>nd</sup> edition, Chapter 16, Eds. W.J. Minkowycz, E.M. Sparrow, and J.Y. Murthy, John Wiley, New York, pp.493-522.
- Cruz, A.G.B., Macêdo, E.N. and Quaresma, J.N.N., 2001, One-dimensional Hyperbolic Heat Conduction Analysis through Finite Volume and Gear Methods, Proceedings of 16<sup>th</sup> Brazilian Congress of Mechanical Engineering, Uberlândia, Brazil, Paper code TRB0071.
- IMSL Library, 1991, Math/Lib, Houston, Texas.
- Lin, J. Y, 1998, "The non-Fourier Effect on the Fin Performance under Periodic Thermal Conditions", Applied Mathematical Modelling, Vol. 22, pp. 629-640.
- Macêdo, E.N., Monteiro, E.R. and Quaresma, J.N.N., 2005, "One-Dimensional Hyperbolic Heat Conduction Analysis through Integral Transform Method", Proceedings of the 4th International Conference on Computational Heat and Mass Transfer, Paris-Cachan, France.
- Quaresma, J.N.N., Macêdo, E.N. and Cruz, A.G.B., 2001, "Hybrid Solution in Hyperbolic Heat Conduction: Laplace Transform with Numerical Inversion", Proceedings of the 2<sup>nd</sup> International Conference on Computational Heat and Mass Transfer, Rio de Janeiro, Brazil.
- Tsai, C.S., Lin, Y.C. and Hung, C.I., 2005, "A Study on the non-Fourier Effects in Spherical Media due to Sudden Temperature Changes on the Surfaces", Heat and Mass Transfer, Vol.41, pp. 706-716.
- Vernotte, P., 1958, "Les Paradoxes de la Theorie Continue de L'Équation de la Chaleur", Comptes Rendus Hebdomadaires des Seances de L'Academie des Sciences, Vol. 246, pp. 3154-3155.
- Wolfram, S., 2003, "The *Mathematica* Book", 5th ed., Wolfram Media.

#### 7. Copyright Notice

The author is the only responsible for the printed material included in his paper.

## HYBRID SOLUTION FOR HYDRODYNAMICALLY DEVELOPING FLOW IN CIRCULAR TUBES

**Simone Patrícia Aranha da Paz**

**Emanuel Negrão Macêdo**

**João Nazareno Nonato Quaresma**

Chemical and Food Engineering Department, CT, Universidade Federal do Pará, UFPA  
Campus Universitário do Guamá, 66075-110, Belém, PA, Brazil; quaresma@ufpa.br

**Renato Machado Cotta**

Mechanical Engineering Department – EE/COPPE, Universidade Federal do Rio de Janeiro, UFRJ  
Cx. Postal 68503 – Cidade Universitária, 21945-970, Rio de Janeiro, RJ, Brazil; cotta@mecanica.coppe.ufrj.br

**Abstract.** *The Generalized Integral Transform Technique (GITT) is employed in the solution of the continuity and momentum equations in hydrodynamically developing laminar flow inside circular ducts. The streamfunction formulation is adopted to automatically satisfy the continuity equation and to eliminate the pressure field, thus providing improved computational performance. A novel eigenvalue problem in this eigenfunction expansion approach is considered, thus eliminating the difficulties associated with the singularity at the channel centerline. Results for the velocity field and for the product of the friction factor-Reynolds number are computed along the entrance region for different governing parameters, which are tabulated for reference purposes and graphically presented as functions of the dimensionless axial coordinate. Critical comparisons with previous results in the literature are also performed, in order to validate the numerical code here developed and to provide a set of benchmark results.*

**Keywords.** *Developing flow, circular tubes, integral transforms, friction factor, laminar flow.*

### 1. Introduction

Laminar flow within pipes is found in several engineering applications, and has therefore been studied in different geometric configurations and operational conditions. The most usual geometry utilized in pipelines is the circular tube, which has been extensively analyzed in laminar flow conditions by many researchers (Shah and London, 1978). In such studies, the aim has been the determination of the velocity field, and as a consequence parameters of practical interest such as pressure drop and friction factors, both in the entrance and fully developed regions. These computations are generally difficult to accomplish by analytical approaches in light of the nonlinearity of the momentum equations, either in the full Navier-Stokes equations formulation or as the simpler boundary layer equations. In this context, purely numerical schemes have been employed to solve the equations for the classical problem in developing laminar flow in circular tubes (Hornbeck, 1964; Liu, 1974).

In the search for a solution with automatically controlled accuracy, the Generalized Integral Transform Technique (GITT) (Cotta, 1993; Cotta, 1994; Cotta, 1998; Santos et al., 2001; Cotta and Mikhailov, 2006) with its automatic global error control capability is a reliable path for obtaining benchmark results, allowing for a more definitive critical evaluation of previously published numerical results of this classical problem. The GITT has already been utilized to find a hybrid analytical-numerical solution for laminar flow development inside parallel-plates channels (Carvalho et al., 1993; Machado and Cotta, 1995; Pérez Guerrero and Cotta, 1995; Figueira da Silva et al., 1996; Figueira da Silva et al., 1999), by using both the primitive variables and streamfunction-only formulations, in either the Navier-Stokes or boundary layer formulations. The application of the GITT under the streamfunction formulation, leads to the solution of an eigenvalue problem of fourth order, inherent to the analytical treatment of the biharmonic equation, which is readily solved in analytic form (Cotta, 1993, Perez Guerrero and Cotta, 1995). The same solution methodology has been employed in the solution of developing flow within annular channels (Pereira et al, 1998; Pereira et al., 2000), by handling the associated biharmonic-type eigenvalue problem in the cylindrical coordinates system.

Until now, however, the use of this hybrid approach in conjunction with the streamfunction formulation for the flow development in circular tubes was not yet possible, due to the need of finding an appropriate fourth order eigenvalue problem in the cylindrical coordinates system, which would alleviate the singularities of the Bessel functions at the duct centerline. This difficulty is here circumvented by adopting a recently introduced eigenvalue problem (Fedele et al., 2005) which accounts for the singularities at the central radial position. The present work is thus aimed at utilizing the ideas in the generalized integral transform technique to construct a hybrid analytical-numerical solution for the flow development in circular tubes, and to accurately compute the velocity field along the entrance and fully developed regions, as well as the product of friction factor-Reynolds number. Critical comparisons with other previously published works are also performed in order to assess the consistency of the available numerical results.

## 2. Analysis

Laminar flow of a Newtonian fluid inside a circular tube is considered as shown in Fig. (1). The continuity and incompressible steady two-dimensional Navier-Stokes equations in cylindrical coordinates are used to model the flow inside this circular geometry. The longitudinal velocity component  $v_z$  is assumed to be known at the channel entrance, and the inlet flow is assumed to be parallel ( $v_r = 0$ ). Fully developed flow conditions are attained at a sufficiently large channel length, recovering the parabolic flow structure. Therefore, the governing equations, in the region  $0 < r < 1$  and  $z > 0$ , are written in dimensionless form as:

$$\frac{1}{r} \frac{\partial(rv_r)}{\partial r} + \frac{\partial v_z}{\partial z} = 0 \quad (1)$$

$$v_r \frac{\partial v_r}{\partial r} + v_z \frac{\partial v_r}{\partial z} = -\frac{\partial p}{\partial r} + \frac{2}{\text{Re}} \left[ \frac{\partial}{\partial r} \left( \frac{1}{r} \frac{\partial}{\partial r} (r v_r) \right) + \frac{\partial^2 v_r}{\partial z^2} \right] \quad (2)$$

$$v_r \frac{\partial v_z}{\partial r} + v_z \frac{\partial v_z}{\partial z} = -\frac{\partial p}{\partial z} + \frac{2}{\text{Re}} \left[ \frac{1}{r} \frac{\partial}{\partial r} \left( r \frac{\partial v_z}{\partial r} \right) + \frac{\partial^2 v_z}{\partial z^2} \right] \quad (3)$$

Equations (1) to (3) are subjected to the following inlet and boundary conditions:

$$v_z(r, 0) = 1; \quad v_r(r, 0) = 0 \quad (4,5)$$

$$v_z(r, \infty) = 4q(1-r^2); \quad v_r(r, \infty) = 0 \quad (6,7)$$

$$v_r(0, z) = 0; \quad \frac{\partial v_z(0, z)}{\partial r} = 0 \quad (8,9)$$

$$v_z(1, z) = 0; \quad v_r(1, z) = 0 \quad (10,11)$$

The dimensionless groups employed in the equations above are defined as

$$r = r^*/r_w; \quad z = z^*/r_w; \quad v_z = v_z^*/u_0; \quad v_r = v_r^*/u_0; \quad p = p^*/\rho u_0^2; \quad \text{Re} = 2u_0 r_w / \nu \quad (12)$$

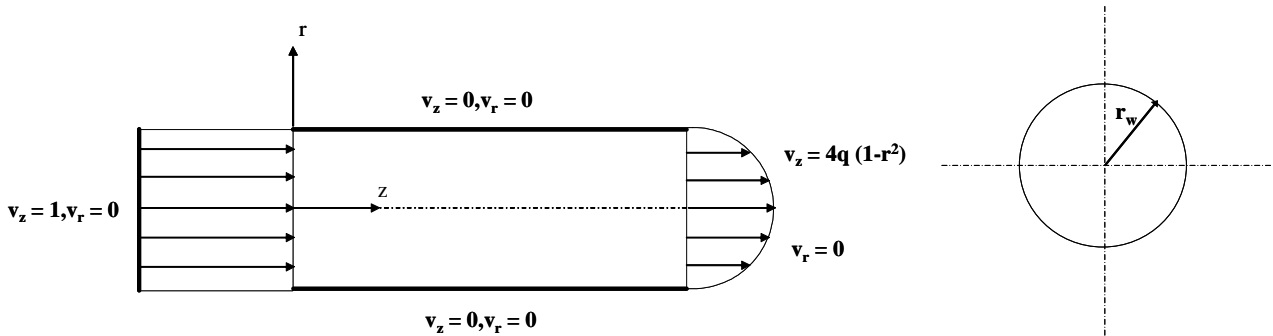


Figure 1. Geometry and coordinates system for developing flow in circular duct.

Now, expressing Eqs. (1) to (11) in the streamfunction-only formulation in order to automatically satisfy the continuity equation and to eliminate the pressure field, the flow modeling is represented by:

$$\frac{1}{r} \frac{\partial \psi}{\partial z} \left[ \frac{\partial}{\partial r} (E^2 \psi) - \frac{2}{r} (E^2 \psi) \right] - \frac{1}{r} \frac{\partial \psi}{\partial r} \frac{\partial}{\partial z} (E^2 \psi) = \frac{2}{\text{Re}} (E^4 \psi) \quad (13)$$

where the operators  $E^2$  and  $E^4$  are defined as

$$E^2 = \frac{\partial^2}{\partial r^2} - \frac{1}{r} \frac{\partial}{\partial r} + \frac{\partial^2}{\partial z^2}; \quad E^4 = E^2(E^2) = \frac{\partial^4}{\partial r^4} - \frac{2}{r} \frac{\partial^3}{\partial r^3} + \frac{3}{r^2} \frac{\partial^2}{\partial r^2} - \frac{3}{r^3} \frac{\partial}{\partial r} - \frac{2}{r} \frac{\partial^3}{\partial r \partial z^2} + 2 \frac{\partial^4}{\partial r^2 \partial z^2} + \frac{\partial^4}{\partial z^4} \quad (14,15)$$



The streamfunction is defined in terms of the dimensionless velocity components in the longitudinal (r) and transversal (z) directions, respectively, as

$$v_r(r, z) = \frac{1}{r} \frac{\partial \psi(r, z)}{\partial z}; \quad v_z(r, z) = -\frac{1}{r} \frac{\partial \psi(r, z)}{\partial r} \quad (16,17)$$

Equation (13) requires the specification of eight boundary conditions expressed in terms of the streamfunction. At the duct walls, no-slip and impermeability conditions are written as:

$$\frac{\psi(0, z)}{r} = C_1; \quad \frac{\partial}{\partial r} \left[ \frac{1}{r} \frac{\partial \psi(0, z)}{\partial r} \right] = 0 \quad (18,19)$$

$$\psi(1, z) = C_2; \quad \frac{\partial \psi(1, z)}{\partial r} = 0 \quad (20,21)$$

$$\frac{\psi(r, 0)}{r} = C_1 - \frac{r}{2}; \quad \frac{\partial \psi(r, 0)}{\partial z} = 0 \quad (22,23)$$

$$\frac{\psi(r, \infty)}{r} = C_1 - 4q \left( \frac{r}{2} - \frac{r^3}{4} \right); \quad \frac{\partial \psi(r, \infty)}{\partial z} = 0 \quad (24,25)$$

$C_1$  and  $C_2$  are constants that specify the streamfunction values at the channel centerline and wall. The constant  $q$  warrants the global mass conservation. Such constants are related by using the boundary conditions above, to yield

$$C_2 = C_1 - 1/2; \quad q = C_1 - C_2 \quad (26,27)$$

One may arbitrarily specify  $C_1 = 0$ , so that  $C_2 = -1/2$  and  $q = 1/2$ .

Equation (13) and boundary conditions (18) to (25) complete the problem formulation in terms of the streamfunction only. Following the ideas in the generalized integral transform technique (Cotta, 1993; Cotta, 1994; Cotta, 1998; Santos et al., 2001; Cotta and Mikhailov, 2006), for improved computational performance, it is convenient to define a filter that reproduces the fully developed flow solution in order to homogenize the boundary conditions in the r direction, which later will be the coordinate chosen for the specification of the eigenvalue problem. Therefore, the filter is written as

$$\psi(r, z) = \psi_\infty(r) + \phi(r, z); \quad \psi_\infty(r) = r^2 \left( \frac{r^2}{2} - 1 \right) \quad (28,29)$$

where the filter  $\psi_\infty(r) \equiv \psi(\infty, r)$  represents the streamfunction in the fully developed region and  $\phi(r, z)$  is now the filtered potential to be solved for. The resulting problem formulation is then given by:

$$\frac{1}{r} \frac{\partial \phi}{\partial z} \left[ \frac{\partial}{\partial r} (\mathbf{E}^2 \phi) - \frac{2}{r} (\mathbf{E}^2 \phi) \right] - \frac{1}{r} \frac{d\psi_\infty}{dr} \frac{\partial}{\partial z} (\mathbf{E}^2 \phi) - \frac{1}{r} \frac{\partial \phi}{\partial r} \frac{\partial}{\partial z} (\mathbf{E}^2 \phi) = \frac{2}{\text{Re}} (\mathbf{E}^4 \phi) \quad (30)$$

$$\frac{\phi(0, z)}{r} = 0; \quad \frac{\partial}{\partial r} \left[ \frac{1}{r} \frac{\partial \phi(0, z)}{\partial r} \right] = 0 \quad (31,32)$$

$$\phi(1, z) = 0; \quad \frac{\partial \phi(1, z)}{\partial r} = 0 \quad (33,34)$$

$$\frac{\phi(r, 0)}{r} = \frac{r}{2} (1 - r^2); \quad \frac{\partial \phi(r, 0)}{\partial z} = 0 \quad (35,36)$$

$$\frac{\phi(r, \infty)}{r} = 0; \quad \frac{\partial \phi(r, \infty)}{\partial z} = 0 \quad (37,38)$$

The present contribution is aimed at solving only the boundary layer equations. Therefore, within the range of validity of the boundary layer hypothesis, the momentum equation given by Eq. (30) and related boundary conditions become:

$$\frac{1}{r} \frac{\partial \phi}{\partial z} \left( \frac{\partial^3 \phi}{\partial r^3} - \frac{3}{r} \frac{\partial^2 \phi}{\partial r^2} + \frac{3}{r^2} \frac{\partial \phi}{\partial r} \right) - \frac{1}{r} \frac{\partial \phi}{\partial r} \left( \frac{\partial^3 \phi}{\partial r^2 \partial z} - \frac{1}{r} \frac{\partial^2 \phi}{\partial r \partial z} \right) - \frac{1}{r} \frac{d\psi_\infty}{dr} \left( \frac{\partial^3 \phi}{\partial r^2 \partial z} - \frac{1}{r} \frac{\partial^2 \phi}{\partial r \partial z} \right) = \frac{2}{\text{Re}} \left( \frac{\partial^4 \psi}{\partial r^4} - \frac{2}{r} \frac{\partial^3 \psi}{\partial r^3} + \frac{3}{r^2} \frac{\partial^2 \psi}{\partial r^2} - \frac{3}{r^3} \frac{\partial \psi}{\partial r} \right) \quad (39)$$

$$\frac{\phi(0, z)}{r} = 0; \quad \frac{\partial}{\partial r} \left[ \frac{1}{r} \frac{\partial \phi(0, z)}{\partial r} \right] = 0 \quad (40,41)$$

$$\phi(1, z) = 0; \quad \frac{\partial \phi(1, z)}{\partial r} = 0 \quad (42,43)$$

$$\frac{\phi(r, 0)}{r} = \frac{r}{2}(1-r^2) \quad (44)$$

## 2.1. Solution methodology

In applying the GITT approach in the solution of the PDE system given by Eqs. (39) to (44), due to the homogeneous characteristics of the boundary conditions in the r direction, it is more appropriate to choose this direction for the process of integral transformation. Therefore, the auxiliary eigenvalue problem is taken as (Fedele et al., 2005)

$$\mathcal{L}^2 X_i(r) = -\lambda_i^2 \mathcal{L} X_i(r) \quad (45)$$

$$\frac{X_i(0)}{r} = 0; \quad \frac{d}{dr} \left[ \frac{1}{r} \frac{dX_i(0)}{dr} \right] = 0 \quad (46,47)$$

$$X_i(1) = 0; \quad \frac{dX_i(1)}{dr} = 0 \quad (48,49)$$

where,

$$\mathcal{L} = r \frac{d}{dr} \left( \frac{1}{r} \frac{d}{dr} \right) = \frac{d^2}{dr^2} - \frac{1}{r} \frac{d}{dr}; \quad \mathcal{L}^2 = \mathcal{L}(\mathcal{L}) = \frac{d^4}{dr^4} - \frac{2}{r} \frac{d^3}{dr^3} + \frac{3}{r^2} \frac{d^2}{dr^2} - \frac{3}{r^3} \frac{d}{dr} \quad (50,51)$$

Problem (45) is analytically solved, to furnish:

$$X_i(r) = r^2 - \frac{r J_1(\lambda_i r)}{J_1(\lambda_i)} \quad (52)$$

where the eigenvalues  $\lambda_i$ 's are computed from the following transcendental equation:

$$J_2(\lambda_i) = 0, \quad i=1, 2, 3, \dots \quad (53)$$

The eigenfunctions satisfy the following orthogonality property:

$$\int_0^1 \frac{dX_i}{dr} \frac{dX_j}{dr} \frac{dr}{r} = - \int_0^1 X_i (\mathcal{L} X_j) \frac{dr}{r} = \begin{cases} 0, & i \neq j \\ N_i, & i = j \end{cases} \quad (54)$$

The normalization integral  $N_i$  is then computed from:

$$N_i = \int_0^1 \frac{1}{r} \left( \frac{dX_i}{dr} \right)^2 dr = \frac{\lambda_i^2}{2} \quad (55)$$

The eigenvalue problem defined by Eqs. (45) to (49) allow for the definition of the following integral transform pair:

$$\bar{\phi}_i(z) = -\frac{1}{N_i} \int_0^1 \phi(r, z) (\mathcal{L} X_i) \frac{dr}{r}, \quad \text{transform} \quad (56)$$

$$\phi(r, z) = \sum_{i=1}^{\infty} X_i(r) \bar{\phi}_i(z), \quad \text{inverse} \quad (57)$$

The next step is thus to accomplish the integral transformation of the original partial differential system given by Eqs. (39) to (44). For this purpose, Eqs. (39) and the inlet condition (44) are multiplied by  $[X_i(r)/r]$ , and after that,

integrated over the domain [0,1] in r, and the inverse formula given by Eq. (57) is employed, resulting in the following coupled ordinary differential system for the calculation of the transformed potentials  $\bar{\phi}_i(z)$  :

$$\sum_{j=1}^{\infty} D_{ij} \frac{d\bar{\phi}_j(z)}{dz} = \frac{2}{\text{Re}} \lambda_i^2 \bar{\phi}_i(z) \quad , \quad i=1,2,\dots \quad (58)$$

$$\bar{\phi}_i(0) = \frac{1}{N_i} \int_0^1 (1-2r^2) \frac{dX_i}{dr} dr \quad (59)$$

where,

$$D_{ij} = \frac{1}{N_i} \left[ B_{ij} + \sum_{k=1}^{\infty} A_{ijk} \bar{\phi}_k(z) \right] \quad (60)$$

$$A_{ijk} = \int_0^1 \frac{X_i}{r^2} \left( X_j X_k''' - \frac{3X_j X_k''}{r} + \frac{3X_j X_k'}{r^2} - X_j'' X_k' + \frac{X_j' X_k'}{r} \right) dr \quad (61)$$

$$B_{ij} = \int_0^1 \frac{X_i}{r^2} \left( \frac{X_j'}{r} - X_j'' \right) \frac{d\psi_{\infty}}{dr} dr \quad (62)$$

In order to numerically handle the ODE system given by Eqs. (58) to (62), through the subroutine DIVPAG from the IMSL Library (1991), it is necessary to truncate the infinite series in a sufficiently high number of terms so as to guarantee the requested relative error in obtaining the original potentials. This subroutine solves initial value problems with stiff characteristics, and provides the important feature of automatically controlling the relative error in the solution of the ordinary differential equations system, allowing the user to establish error targets for the transformed potentials.

Once the transformed potentials are available, the velocity field is obtained from the definition of the streamfunction given by Eqs. (16) and (17), after introducing the inverse formula (57), to yield:

$$v_r(r,z) = \sum_{i=1}^{\infty} \frac{X_i(r)}{r} \frac{d\bar{\phi}_i(z)}{dz} ; \quad v_z(r,z) = 2(1-r^2) - \sum_{i=1}^{\infty} \frac{X_i'(r)}{r} \bar{\phi}_i(z) \quad (63,64)$$

In the realm of applications, one is concerned with quantities of practical interest such as the product of the Fanning friction factor-Reynolds number, fRe. The friction factor is defined as:

$$f = \tau_w / (\rho u_0^2 / 2) \quad (65)$$

with the introduction of the inverse formula for the velocity field given by Eq. (64), it then results

$$f\text{Re} = 16 + 4 \sum_{i=1}^{\infty} \frac{d}{dr} \left[ \frac{1}{r} \frac{dX_i(1)}{dr} \right] \bar{\phi}_i(z) \quad (66)$$

### 3. Results and discussion

Numerical results for the velocity profiles and the product of the Fanning friction factor-Reynolds number were produced along the entrance region of a circular tube. The computational code was developed in FORTRAN 90/95 programming language and implemented on a PENTIUM-IV 1.3 GHz computer. First, the numerical code was validated against those results presented by Hornbeck (1964), Liu (1974), Shah and London (1978), and Nascimento et al. (2006). The routine DIVPAG from the IMSL Library (1991) was used to numerically handle the truncated version of the system of ordinary differential equations (Eqs. (58) to (62)), with a relative error target of  $10^{-8}$  prescribed by the user, for the transformed potentials. These results were produced with different truncation orders ( $N = 10, 20, 40, 60, 80$  and  $100$ ) and for  $\text{Re} = 2000$ , but it should be noted that the dimensionless axial coordinate  $X^+ = z/(2\text{Re})$  makes the results independent of the Reynolds number for the boundary layer formulation.

Figure (2) and Table (1) show the convergence behavior of the longitudinal velocity component at the centerline of the circular tube, as well as its comparison with the results presented in Hornbeck (1964), Liu (1974), Shah and London (1978) and Nascimento et al. (2006), demonstrating a good agreement, which provides a direct validation of the numerical code here developed. From inspection of Figure (2), a monotonic convergence behavior for the longitudinal velocity component at the centerline of the circular tube is observed.

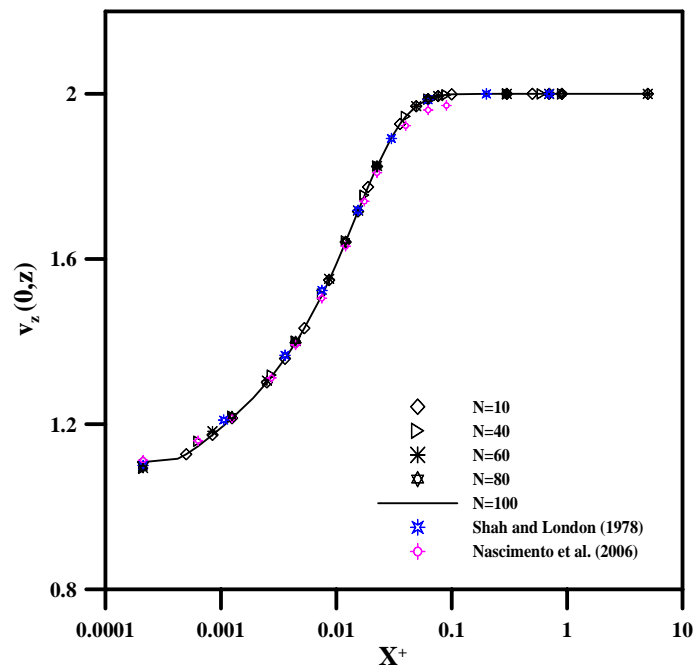


Figure 2. Development of the longitudinal velocity component  $v_z$  at the centerline of the duct.

Results for the product of the Fanning friction factor-Reynolds number are shown in Fig. (3) as a function of the dimensionless axial positions,  $X^+$ . It can be observed that the product  $fRe$  diminishes until the fully developed region is reached, in which this parameter assumes a constant value. Also, it should be recalled that the product  $fRe$  presents higher values in the entrance region of the circular tube due to higher velocity gradients experimented by the fluid in this region, and for this reason convergence of the eigenfunction expansions is slower within the entrance. Also, a comparison with the results of Nascimento et al. (2006) is performed, showing an excellent agreement. This previous contribution with the GITT implemented a hybrid solution for the same boundary layer formulation, but under primitive variables mathematical description, which leads to different auxiliary eigenvalue problems, and thus different convergence behavior patterns.

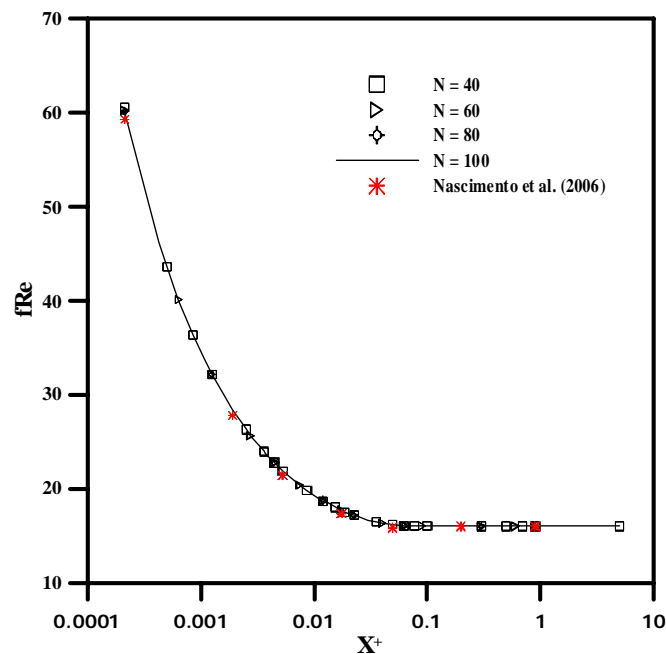


Figure 3. Product  $fRe$  along the entrance region of a circular tube.



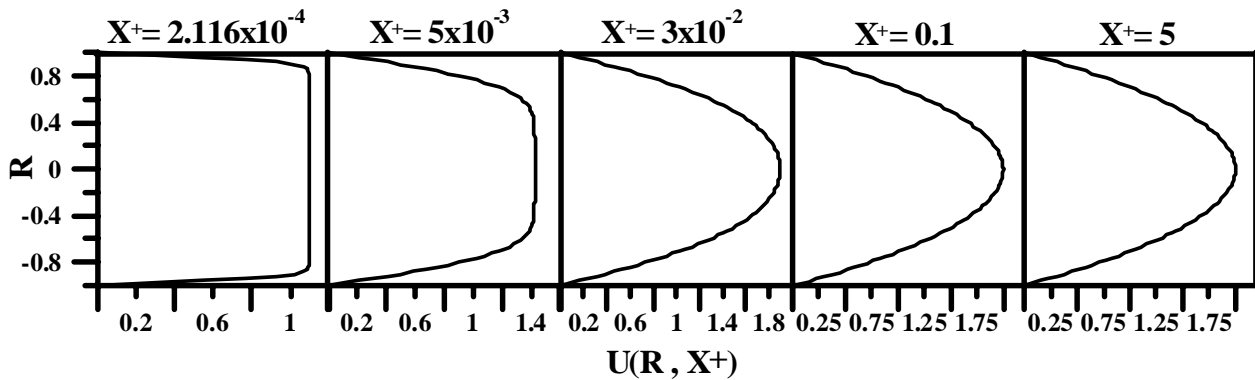


Figure 4. Development of the longitudinal velocity component profile along the tube length.

#### 4. Conclusions

Numerical results for the velocity field and product Fanning friction factor-Reynolds number were produced by using the GITT approach in the solution of the continuity and momentum equations for the flow of Newtonian fluids within circular tubes. The adoption of a recently described eigenvalue problem in the cylindrical coordinates system made possible the application of a streamfunction formulation in such a geometry. The excellent agreement of the present results with previously reported ones demonstrates the consistency of this approach for benchmarking such class of problems. The approach presented, in addition, opens up new perspectives in the hybrid numerical-analytical solution of nonlinear heat and fluid flow problems in non-cartesian coordinates systems.

#### 5. Acknowledgements

The authors would like to acknowledge the financial support provided by CNPq.

#### 6. References

- Carvalho, T.M.B., Cotta, R.M. and Mikhailov, M.D., 1993, "Flow Development in Entrance Region of Ducts", *Communications in Numerical Methods in Engineering*, Vol. 9, pp. 503-509.
- Cotta, R.M., 1993, "Integral Transforms in Computational Heat and Fluid Flow", CRC Press, Boca Raton, USA.
- Cotta, R.M., 1994, "Benchmark Results in Computational Heat and Fluid Flow: - The Integral Transform Method", *Int. J. Heat Mass Transfer (Invited Paper)*, Vol. 37 (Suppl. 1), pp. 381-393.
- Cotta, R.M., 1998, "The Integral Transform Method in Thermal and Fluid Sciences and Engineering", Begell House, New York.
- Cotta, R.M., and Mikhailov, M.D., 2006, "Hybrid Methods and Symbolic Computations", in: *Handbook of Numerical Heat Transfer*, 2<sup>nd</sup> edition, Chapter 16, Eds. W.J. Minkowycz, E.M. Sparrow, and J.Y. Murthy, John Wiley, New York, pp.493-522.
- Fedele, F., Hitt, D.L. and Prabhu, R.D., 2005 "Revisiting the Stability of Pulsatile Pipe Flow", *European Journal of Mechanics B/Fluids*, Vol. 24, pp. 237-254.
- Figueira da Silva, E., Perez Guerrero, J.S., and Cotta, R.M., 1996, "A Comparison of Integral Transform Strategies for Solving the Boundary Layer Equations", *Proc. of the 6th National Thermal Sciences Meeting, ENCIT-96*, Florianópolis, SC, Brasil.
- Figueira da Silva, E., Perez-Guerrero, J.S., and Cotta, R.M., 1999, "Integral Transform Solution of Boundary Layer Equations in Streamfunction-only Formulation", *Int. J. Non-Linear Mechanics*, V.34, pp.51-61.
- Hornbeck, R.W., 1964, "Laminar Flow in the Entrance Region of a Pipe", *Applied Scientific Research, Section A*, Vol. 13, pp. 224-232.
- IMSL Library, 1991, MATH/LIB, Houston, TX.
- Liu, J., 1974, "Flow of a Bingham Fluid in the Entrance Region of an Annular Tube", M. S. Thesis, University of Wisconsin – Milwaukee, USA.
- Machado, H.A. and Cotta, R.M., 1995, "Integral Transform Method for Boundary Layer Equations in Simultaneous Heat and Fluid Flow Problems", *Int. J. Num. Meth. Heat & Fluid Flow*, V. 5, pp. 225-237.
- Nascimento, S.C.C., Macêdo, E.N. and Quaresma, J.N.N., 2006, "Generalized Integral Transform Solution for Hydrodynamically Developing Non-Newtonian Flows in Circular Tubes", *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Vol. 28, pp. 125-130.

- Pereira, L.M., Perez Guerrero, J.S., and Cotta, R.M., 1998, “Integral Transformation of the Navier-Stokes Equations in Cylindrical Geometry”, *Computational Mechanics*, V.21, no.1, pp.60-70.
- Pereira, L.M., Cotta, R.M., and Perez Guerrero, J.S., 2000, “Analysis of Laminar Forced Convection in Annular Ducts Using Integral Transforms”, *Proc. of the 15<sup>th</sup> Brazilian Congress of Mechanical Engineering, COBEM 99, Águas de Lindóia, São Paulo, Brasil, December 1999 (CD-ROM)*; also, *Hybrid Meth. Eng.*, V.2, no.2, pp.221-232.
- Perez Guerrero, J.S. and Cotta, R.M., 1995, “Integral Transform Solution of Developing Laminar Duct Flow in Navier-Stokes Formulation”, *International Journal for Numerical Methods in Fluids*, Vol. 20, pp. 1203-1213.
- Santos, C.A.C., Quaresma, J.N.N., and Lima, J.A., eds., 2001, “Benchmark Results for Convective Heat Transfer in Ducts: - The Integral Transform Approach”, *ABCM Mechanical Sciences Series, Editora E-Papers, Rio de Janeiro*.
- Shah, R. K. and London, A. L., 1978, “Laminar Flow Forced Convection in Ducts”, In *Advances in Heat Transfer (Supplement 1)*, New York, Academic Press.

## **7. Copyright Notice**

The author is the only responsible for the printed material included in his paper.

## FINITE ELEMENT MESH GENERATION FOR NUMERICAL SIMULATION OF HYDROELECTRIC POWER PLANT RESERVOIR FILLING

### **Abrão Macandi Dongala**

Universidade do Estado do Rio de Janeiro, UERJ, Faculdade de Engenharia.  
Rua São Francisco Xavier, 524 – 20550-013 – Rio de Janeiro, RJ, Brasil.  
[abramado1@hotmail.com](mailto:abramado1@hotmail.com)

### **Leon Matos Ribeiro De Lima**

Universidade do Estado do Rio de Janeiro, UERJ, Faculdade de Engenharia.  
Rua São Francisco Xavier, 524 – 20550-013 – Rio de Janeiro, RJ, Brasil.  
[leonmatos@hotmail.com](mailto:leonmatos@hotmail.com)

### **Norberto Mangiavacchi**

Departamento de Engenharia Mecânica, Universidade do Estado do Rio de Janeiro, UERJ.  
Rua São Francisco Xavier, 524 – 20550-013 – Rio de Janeiro, RJ, Brasil.  
[norberto.mangiavacchi@gmail.com](mailto:norberto.mangiavacchi@gmail.com)

### **Cássio Botelho Pereira Soares**

Departamento de Engenharia Ambiental, Furnas Centrais Elétricas, S.A.  
Rua Real Grandeza, 219 – Rio de Janeiro, RJ, Brasil.  
[cassiobp@furnas.com.br](mailto:cassiobp@furnas.com.br)

**Abstract.** *This paper presents the application of the Finite Element Method on the numerical simulations of hydroelectric power plant reservoir filling, focusing on two-dimensional mesh generation. The problem of the heterogeneity of the data sets in hydrodynamic simulations, which differ in precision, type, structure, among other factors, is addressed. The methodology employed in the 2D mesh generation is shown, as well as its relation to the construction of the 3D terrain models. Four different available representations of the terrain are presented, with methods of conversion from one representation to another. The mesh triangulation is made from a set of Cartesian points, using Delaunay triangulation. The present methodology is part of a larger project, based on the object oriented programming paradigm, which has been an effective improvement in this kind of software development.*

**Keywords:** *mesh generation, finite element, object-oriented programming, hydrodynamic simulations.*

### **1. Introduction:**

The filling process of a hydroelectric power plant reservoir can cause several problems, such as under water biomass decomposition, which may compromise the quality of all the water affected by the reservoir. In order to avoid such consequences, a precise analysis is necessary, and studies based on numerical methods are a powerful tool that allows a reliable prediction analysis. This is an important issue given that water supply and water quality is an increasing problem that sooner or later we will have to face.

Many projects of energy generation for hydroelectric power plants have been elaborated and concluded. As the number of such projects increases a larger body of scientific studies on the ambient and socio-economic impacts is necessary. Especially in Brazil, where the hydrographic conditions are an advantage for the exploitation of hydroelectric reservoirs, the area covered by them is very extensive, and so the problems they bring. We can minimize these problems by obtaining information that provides a more consistent forecast capability of the consequences of a hydrographic basin flooding. A useful tool to gather forecast information about the consequent ambient impacts is the numerical simulation.

The main contribution of this work is a 2D mesh generation procedure for the application on numerical simulation of the hydrodynamic phenomena present in a hydroelectric reservoir. The presented system of mesh generation is part of a larger project of studies and simulations on the ambient impacts related to the flooding of a certain region for hydroelectric power plant construction. There are other similar works on hydrodynamic simulations, which make use of two-dimensional meshes, such as FESWMS (Finite Element Surface Water Modeling System). The difference to the method we are presenting is that, although the focus of this paper is grid assembling for applications on 2DH modeling for shallow water problems, the generation of 2D mesh is also employed on three-dimensional mesh generation, for 3D fluid flow problems, which is one of the simulation models of this project.

The simulations results are obtained basically from the solution of the momentum conservation equations for  $x$  and  $y$  directions and the continuity equation, utilizing the Finite Element Method (FEM) for the mathematical modeling.



One of the input data is the terrain information, and this project has among its objectives to give a major flexibility on the terrain representation options, offering conversion techniques. Objects and methods related to ground representation belong to the Terrain module, which prepares the data for the mesh generation, in the Finite Elements Mesh (FEMesh) module.

This structure is part of the software engineering environment of this work, and it is based on the object oriented concept. MATLAB was chosen for the development work because it is a high-level language directed to mathematical applications that provides all the features of a traditional programming language, including OOP.

## 2. Simulation environment

The simulation environment is composed of several modules that encapsulate specific tasks in the simulation process. A brief explanation over each module is shown below.

### 2.1. Terrain

Here the terrain data sets are handled. This is where the user defines the domain of simulation.

### 2.2. FEMesh2d

This module is responsible for 2D and 3D mesh assembling.

### 2.3. ShapePhyto

The vegetation map of the terrain in study is manipulated in this module, as well as the phyto-physiognomy data.

### 2.4. Decay

This module is responsible for the biomass decay calculations. It quantifies the organic mass transferred to the water, which impacts on the chemical elements percentages present in the reservoir water.

### 2.5. Simulator

In this module all the information handled in the other modules is collected for the simulation itself. This module also has a strong coupling with the Decay module: on one hand, the Decay provides biomass quantities present in the water for the transport equations in the Simulator module, and, on the other hand, the decay coefficients are function of transport parameters, like temperature, pressure, oxygen concentration etc.

The focus of this paper is the Terrain and FEMesh modules.

The modules can be accessed using a high level script language, based on MATLAB, which gives great power and flexibility to the user. Alternatively, the user can employ a simple Graphic User Interface (GUI) to perform common tasks. Figure 1 shows a simplified scheme of the whole system of simulation, where the data flux between the modules is represented by the arrows.

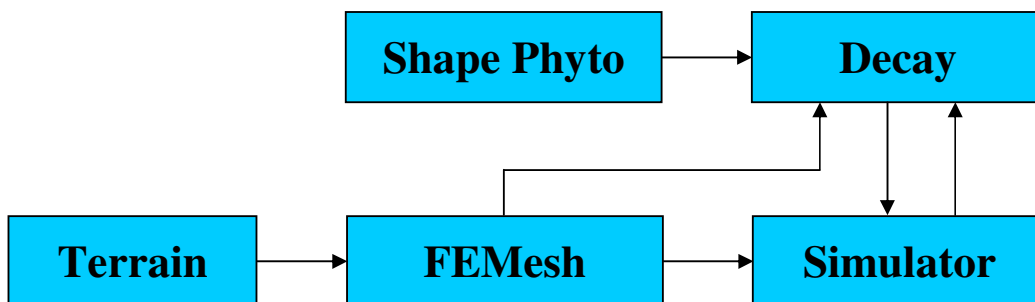


Figure 1. Data flux between the system's modules.

The FEMesh generator module is closely related to the terrain modeling module. In fact, the triangular mesh is built using topological data provided by the Terrain module. The terrain can be represented by a set of points (named CloudTerrain), a cartesian mesh of points (RasterTerrain), a set of level-curves (ShapeTerrain), or a triangular mesh (MeshTerrain). The mesh triangulation is made from the CloudTerrain representation, using Delaunay triangulation. After this stage, the mesh can be improved employing a refining algorithm, and information regarding boundary

conditions is included in the data structure, to prepare for the application of the mesh in the Finite Elements Method, in the Numerical Simulations Module.

### 3. Governing equations

In this work the two-dimensional mesh generation is analyzed for the case of shallow water hydrodynamic simulations. In this kind of problem it is convenient to consider hydrostatic pressure, integrated on the vertical direction ( $z$  axis, in a cartesian coordinates system), and average velocities only in the horizontal plane ( $x$  and  $y$  directions), which is the 2DH modeling method.

The governing equations for the hydrodynamic circulation were described by Rosman (2001), in which the 2DH technique is employed to solve the kind of problem this work is focused on. More details about 2DH hydrodynamic modeling can be found in this reference. Here, the main results will be briefly presented.

For each time step the 2DH simulator determines the vertically averaged velocity, on the  $x$  and  $y$  directions,  $U(x,y,t)$  and  $V(x,y,t)$  respectively, and the elevations on the free surface,  $z=\zeta(x,y,t)$ . The three necessary equations to determine three unknowns of hydrodynamic circulation in 2DH flow, integrated on the vertical direction,  $\zeta$ ,  $U$  and  $V$ , are represented below:

2DH equation of momentum conservation for a flow integrated on the vertical, on the  $x$  direction:

$$\frac{\partial U}{\partial t} + U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial y} = -g \frac{\partial \zeta}{\partial x} + \frac{1}{\rho_0 H} \left( \frac{\partial(H \tau_{xx}^-)}{\partial x} + \frac{\partial(H \tau_{xy}^-)}{\partial y} \right) + \frac{1}{\rho_0 H} (\tau_x^s - \tau_x^B) + 2\phi \sin \theta V \quad (1)$$

Where  $H$  is the distance from the bottom to the surface at each point, and  $\rho_0$  is the water density. 2DH equation of momentum conservation for a flow integrated on the vertical, on the  $y$  direction:

$$\frac{\partial V}{\partial t} + U \frac{\partial V}{\partial x} + V \frac{\partial V}{\partial y} = -g \frac{\partial \zeta}{\partial x} + \frac{1}{\rho_0 H} \left( \frac{\partial(H \tau_{xy}^-)}{\partial x} + \frac{\partial(H \tau_{yy}^-)}{\partial y} \right) + \frac{1}{\rho_0 H} (\tau_y^s - \tau_y^B) - 2\phi \sin \theta U \quad (2)$$

Continuity equation integrated along the vertical direction:

$$\frac{\partial \zeta}{\partial t} + \frac{\partial UH}{\partial x} + \frac{\partial VH}{\partial y} = 0 \quad (3)$$

The terms of equation (1) can be explained as follows:

$\frac{\partial U}{\partial t} \rightarrow$  represents the local acceleration of the flow, at a given position, or, in other words, the temporal variation rate of the 2DH momentum flux per mass unit. In permanent flows, this term is zero.

$U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial y} \rightarrow$  represents the advective acceleration of the flow, on a given instant. These terms represents the spatial variation rate of the 2DH momentum flux on the  $x$  direction per mass unit. In uniform flows, these terms are zero.

$-g \frac{\partial \zeta}{\partial x} \rightarrow$  represents the resultant hydrostatic pressure on the  $x$  direction (pressure gradient). Due to the water surface declivity on the  $x$  direction, as indicated by the negative sign, the flow tends to move from places where the water level is higher to places where the water level is lower.

$\frac{1}{\rho_0 H} \left( \frac{\partial(H \tau_{xx}^-)}{\partial x} + \frac{\partial(H \tau_{xy}^-)}{\partial y} \right) \rightarrow$  represents the resultant of the 2DH turbulent dynamic tensions in the flow. For example, these terms are caused by horizontal vortices in recirculation zones.

$2\phi \text{sen}\theta V \rightarrow$  represents the coriolis force. This term is present because the referential is moving with the Earth surface.

$\frac{1}{\rho_0 H} (\tau_x^s) \rightarrow$  represents the wind tension on the free surface per mass unit. If the wind is on the same direction of the flow, this term will speed up the flow; otherwise, if the wind is on the opposite direction, it will slow down the flow.

$\frac{1}{\rho_0 H} (-\tau_x^B) \rightarrow$  represents the friction tension on the bottom of the flow per mass unit. As indicated by the negative sign, this term always tends to deaccelerate the flow. It is always opposite to the flow.

Evidently, the meaning of the similar terms in the equation (2), of the momentum conservation on the y direction is analogous.

For the 2DH continuity equation (3), we have:

$\frac{\partial(\zeta + h)}{\partial t} = -\frac{\partial HU}{\partial x} - \frac{\partial HV}{\partial y} \rightarrow$  represents the variation of the water column height ( $\zeta+h$ ) in time as result of the effective fluxes through the water column on the x and y directions, respectively  $\frac{\partial HU}{\partial x}$  and  $\frac{\partial HV}{\partial y}$ .

The water source is given only by affluents, where the boundary conditions are prescribed velocity. The points of outflow are treated with prescribed height  $\zeta$  boundary conditions, and on the reservoir margins the velocity is equal to zero. The water source due to precipitations is accounted for in the affluents inflow velocities.

The solution technique chosen is the Finite Element Method with Galerkin variational formulation. The method is applied on a mesh of Delaunay triangles, with the domain represented by mini-elements – pressure on the vertices and velocity on the vertices and centroid. For the convective term, the semi-lagrangian formulation was used, and the projection method was chosen to decouple the velocity and pressure calculations.

The discretization produces two algebraic linear equation systems in each time iteration. To solve these systems, the preconditioned conjugated gradient method is employed, using an incomplete Cholesky preconditioner.

#### 4. Terrain module

This is the module that deals with the terrain topological data, which can be manipulated by four alternative basic representations (Raster Terrain, Cloud Terrain, Shape Terrain and Mesh Terrain representations). The Terrain module has a class which is responsible for each representation type, described as follows:

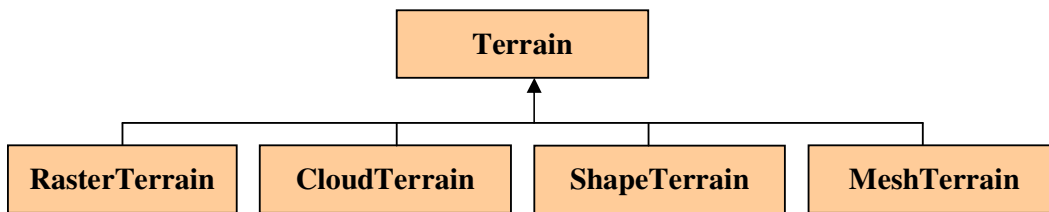


Figure 2. Terrain module classes.

##### 4.1. RasterTerrain

The terrain is represented by a set of cartesian coordinates  $(x,y,z)$ , where each one is a surface point of the terrain. The representation is based on structured quadrilateral grids. It can be constructed, for instance, using a disordered set of

points, defining the intervals between each coordinate, and interpolating the z values. This class handles Digital Terrain Model (DTM) and Digital Elevation Model (DEM) raster files, and converts MeshTerrain models to RasterTerrain.

#### 4.2. ShapeTerrain

This class represents terrain models by a set of level curves. It handles Shapefile struct files. Figures 7 and 8 show a top view visualization of a Shape Terrain. Visualizations can be also performed employing different perspective views.

#### 4.3. CloudTerrain

The terrain is represented by a “cloud” of disordered cartesian points, with position given by x, y and z coordinates. This class converts RasterTerrain and ShapeTerrain into CloudTerrain.

#### 4.4. MeshTerrain

The terrain is represented by a two-dimensional triangular mesh, obtained by a Delaunay triangulation algorithm (Shojaee, 2006; Hao,1997). Figure 9 shows a Mesh Terrain representation. The triangular grid can be generated on the terrain from a set of disordered cartesian points. It converts CloudTerrain into MeshTerrain.

Figure 3 shows the relation between the constructor classes of the terrain models and the FEMesh2D class, which belongs to the FEMesh module.

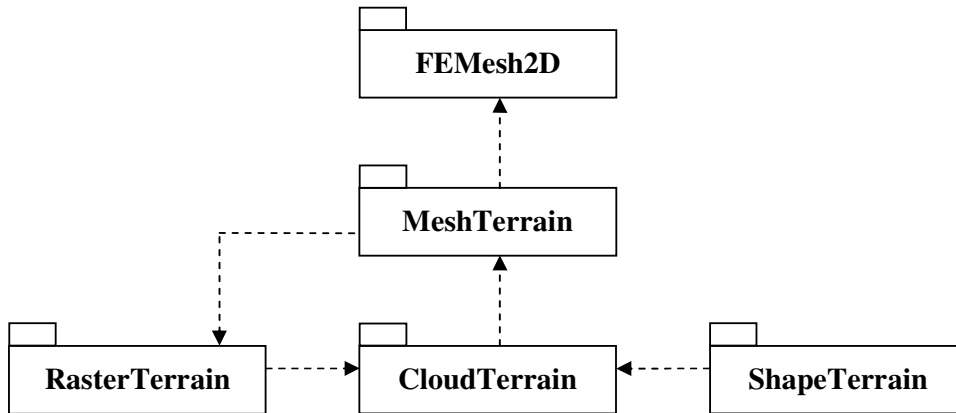


Figure 3. Package diagram showing the relation between Terrain classes and FEMesh2D class.

The availability of four possible terrain representations is a way of increasing the user’s flexibility in the data input for the simulation. However, the basic terrain type used by the simulation is the MeshTerrain, because it is this the terrain model employed by the FEMesh module to generate the mesh that will be used by the Simulator module. In fact, the class that builds 2D meshes is the MeshTerrain.

### 5. FEMesh module

This is the module that creates the data structures required for the Finite Element Mesh representation. This module includes three basic classes, as shown in figure 4, described as follows:

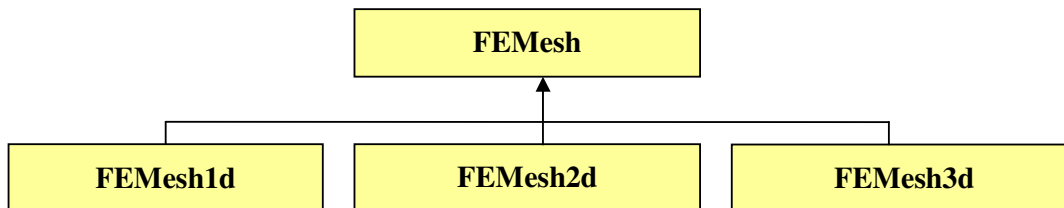


Figure 4. FEMesh module classes.

### 5.1. FEMesh1d

Represents 1d FEM Meshes. It can be employed for 1d simulations, or to collect information on 1d geometric features, such as reservoir boundaries in 2d simulations.

### 5.2. FEMesh2d

Represents 2d FEM Meshes. It can be employed for 2d simulations, or to collect information on 2d geometric features, such as reservoir boundaries in 3d simulations.

### 5.3. FEMesh3d

Represents 3d FEM Meshes. It can be employed for 3d simulations.

## 6. Mesh generation process

The mesh utilized for the terrain representation is a mesh of triangular elements. A justification for this choice is the fact that triangular grid models allow a more reliable representation of the superficial distribution in regions with regions of large altitude gradients. According to Trisiripisal (2003), since each triangular patch contains redundancy in information, it can be used to indicate the characteristics of a region. This advantage is very useful in computer vision and pattern recognition applications to describe the information present in the image.

Another advantage of using triangular meshes is a large reduction in data for processing, and this is very useful in many applications. For example, instead of transforming the entire data, only the representative vertex data in the triangular mesh is required in the calculation. This can significantly speed-up the computation. Delaunay triangulation is a good technique, from the point of view of its approximation properties in combination with finite elements, because it maximizes the minimum angle compared to any other triangulation of the same set of points.

### 6.1. Delaunay triangulation

In a few words, given a set of data points, the Delaunay triangulation is a set of lines connecting each point to its natural neighbors. The Delaunay triangulation is related to the Voronoi diagram – the circle circumscribed about a Delaunay triangle has its center at the vertex of a Voronoi polygon. A Voronoi polygon about point  $P$  is the region that is closer to this point, or as close to, than to any other point. The Voronoi or Dirichlet tessellation is made up from these Voronoi polygons and the Delaunay triangulation is the dual of the Dirichlet tessellation. The next figure shows a set of Delaunay triangles and Voronoi polygons, both made up from a set of disordered points.

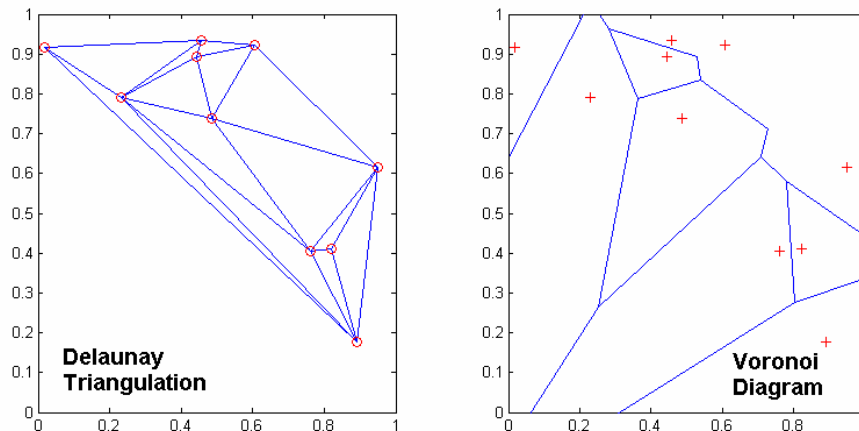


Figure 5. Mesh of Delaunay triangles (left box) and Voronoi polygons diagram (right box).

### 6.2. Mesh generation routine

The first step on the mesh assembling is to convert the terrain format into a CloudTerrain representation. We need this conversion because the Delaunay triangulation is made on a set of disordered cartesian points. After the triangulation, the mesh is assembled, and we can proceed to the next step, in the FEMesh module. Now there are some

adjustments to improve the grid. The first step is the elimination of the elements above the maximum height, which is an essential routine. The other one is a refining option, consisting in the insertion of a point in the middle of edges larger than an arbitrary size limit. All These tasks are implemented by the **FEMesh2D** class.

There's also another alternative way for generating the mesh of Delaunay triangles. Instead of sending a cloud of cartesian points for the Delaunay triangulation, this routine starts from a given domain boundary definition, and selects the interior points, necessary to build the triangular elements. The advantage is that these points disposition seeks for the most well shaped triangles, which assures a more stable implementation of the FEM. The drawback is a major computational cost.

The following image shows a two-dimensional triangular mesh representing a terrain.

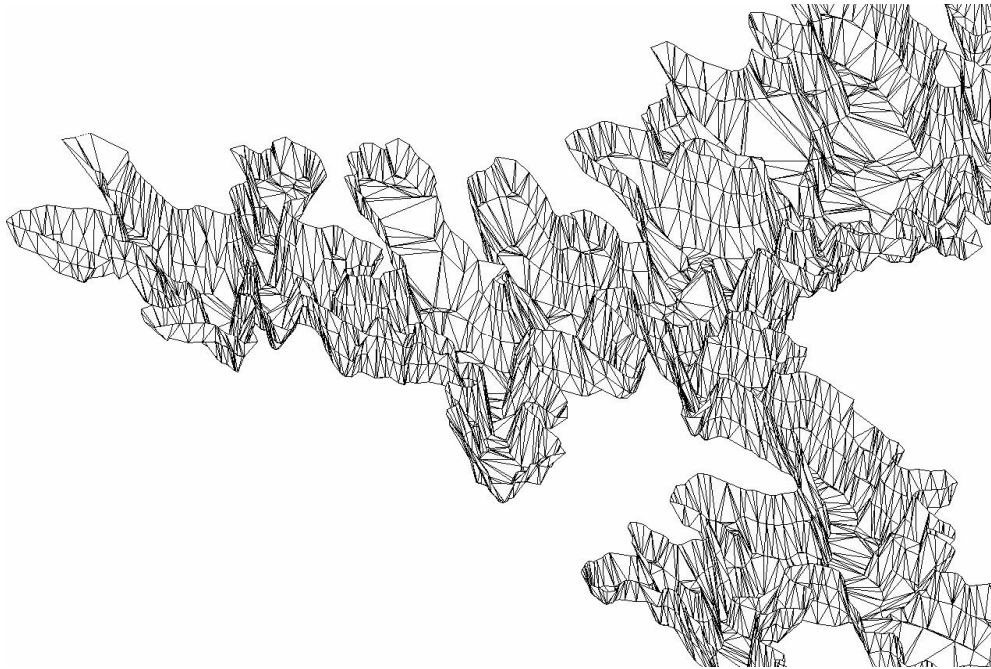


Figure 5. Example of a Terrain represented by a Delaunay triangular mesh.

When this stage is over, the mesh is ready to use in the simulation by the Transport module.

Now an example will be shown in order to illustrate the mesh generation process. Figure 6 shows a shell script sample carrying out the steps required for the generation of a FEM mesh.

```
% FEMesh2d Sample script

% Read shapefile using ShapeTerrain
filename=[filedir,'theme1'];
t1=ShapeTerrain(filename);

% Select a region
t2=trim(t1,7.365e5,7.455e5,75.23e5,75.31e5,0,1120);

% Convert to CloudTerrain
t3=CloudTerrain(t2);

% Convert to MeshTerrain
t4=MeshTerrain(t3);

% Show resulting MeshTerrain
surf(t4);

% Generate FEMesh2d
m=FEMesh2d(t4);

% Show resulting FEMesh2d
figure,surf(m);
```

Figure 6. Example of a shell script to generate a FEMesh2d object.

The script starts reading data from a Shapefile struct file of a region in the city of Nova Friburgo, in the state of Rio de Janeiro, named Caledonia. Since we start with a Shapefile struct file, we employ the ShapeTerrain constructor. Following convenient criteria, an appropriate place for a hydroelectric reservoir must be chosen. The ShapeTerrain class provides convenient clipping resources, employing the *trim* function. We will assume, for the sake of this example, that the region shown in figures 7 and 8 is a good choice.

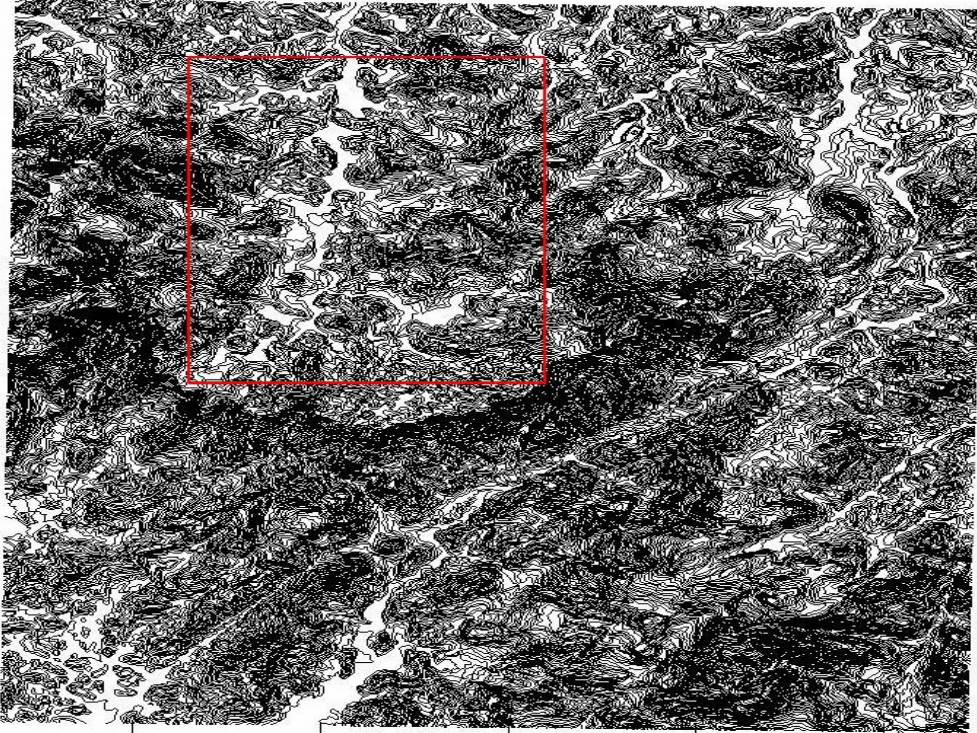


Figure 7. ShapeTerrain representation before trim.

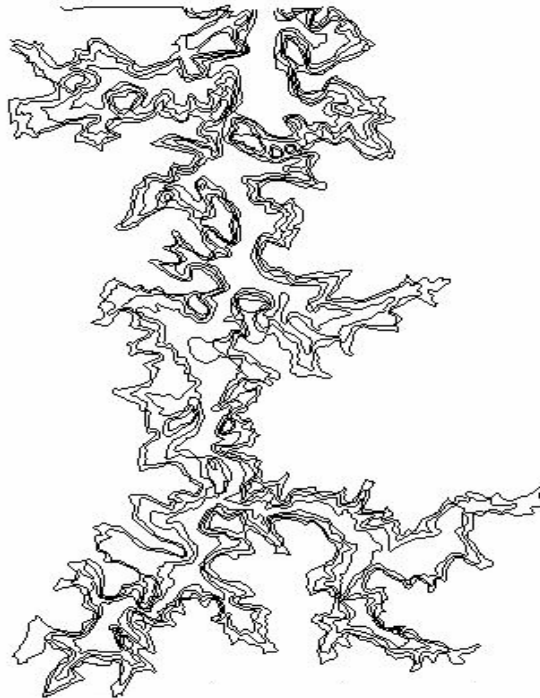


Figure 8. Selected region

The next step is to convert the terrain level curves into a CloudTerrain representation.

The MeshTerrain class can now be employed to perform the Delaunay triangulation to build the mesh (Figure 9), and send it to the FEMesh module, for the final adjustments (Figures 10 and 11).

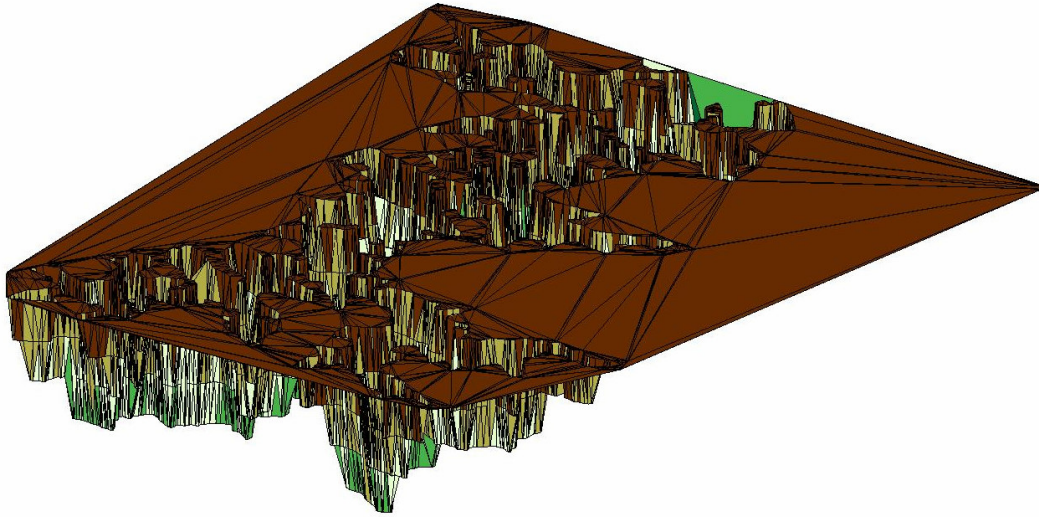


Figure 9. MeshTerrain representation. Here a primitive mesh is assembled. But note that there are many triangles on the maximum height of the selected region. These elements do not represent the terrain surface, therefore, they will be eliminated in the next stage, processed by the FEMesh module.

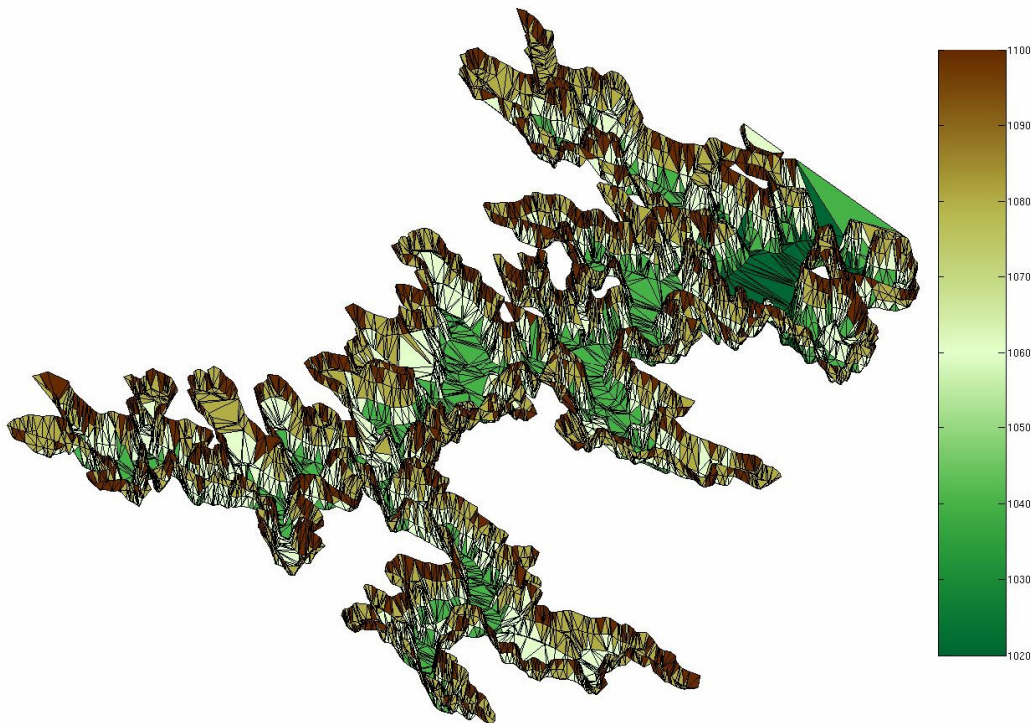


Figure 10. FEMesh2D mesh. This is the mesh after deleting the elements on the maximum height. The scale on the right side of the figure indicates the altitude in meters.



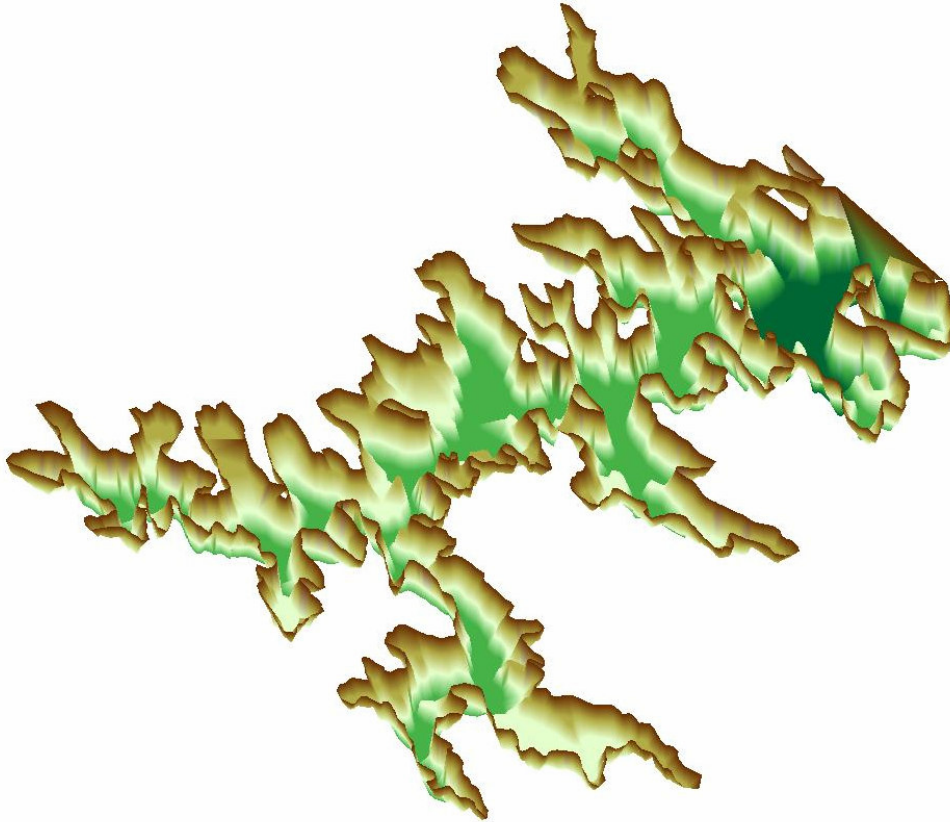


Figure 11. Same mesh of the previous figure, but with a color interpolation. The colors have the only purpose of coloring the figure and give some perspective notion.

## 7. Conclusion

This work is part of a major project, under development, which proposes the construction of a methodology to study and forecast the environmental impacts caused by a hydroelectric power plant reservoir flooding. All its elaboration has been based on software engineering concepts, aggregated to object oriented programming. The development of the Terrain and FEMesh modules, as well as the other ones, is inserted in this methodology, aiming the optimization of the structure and construction speed. This strategy allows a more flexible communication between the modules, in such a way that, for example, if a necessary correction must be done in the internal structure of the Terrain module, without any modification in its output data, the FEMesh module doesn't have to be altered.

The employed software engineering methodology allowed the implementation of simple yet powerful tools for terrain data manipulation in several representations, providing adequate visualization and simulation to the user. This last includes the mesh generation, which can start from any of these possible terrain representations.

One objective of this research is the construction of a friendly tool for simulations of the consequences of a hydroelectric reservoir filling process. The availability of different terrain representation types and the visualization resources are signs of this intent.

## 8. Acknowledgements

The authors thank FURNAS Centrais Elétricas S.A. for financial support, PROCIÊNCIA/UERJ and CNPq for the aids granted to N. Mangiavacchi.

## 9. References

- Trisiripisal, P., 2003, "Image Approximation Using Triangulation", Thesis Submitted to the Faculty of Virginia Polytechnic and State University, Blacksburg, Virginia, pp. 2-4.
- Rosman, P.C.C., 2001, "Um Sistema Computacional de Hidrodinâmica Ambiental", in Silva, R. C. V.,

- “Métodos Numéricos em Recursos Hídricos 5.1”, ed. Porto Alegre: ABRH, v. 5, pp. 1-161.
- Burgos, B. X. V., 2000, “Approximation and Geometric Processing of Digital Images With Adaptive Triangular Meshes”, Universidad Politécnica de Cataluña, pp. 72-93.
- Fowler, M., Scott, K., 2000, “UML Essencial, Um Breve Guia Para a Linguagem-Padrão de Modelagem de Objetos”, ed. Bookman, pp. 103-111.
- Paetau, P., 2005, “On the Benefits and Problems of the Object-Oriented Paradigm Including a Finnish Study”, Swedish School of Economics and Business Administration, Helsingfors, pp. 53-94.
- Shojaee, D., Helali, H., Alesheikh, AA., 2006, "Triangulation for Surface Modelling", Islamic Azad University, Maybod Branch, Yazd, Iran, pp 1-5.
- Hao, C., Jonathan, B., 1997, "Delaunay Triangulation for Curved Surfaces", in Meshing Research Corner, pp 1-13.

## Modelagem numérica de escoamentos acoplados ao transporte de uma espécie química por elementos finitos

### Gustavo Rabello dos Anjos

UFRJ - Programa/Depto. de Eng. Metalúrgica e de Materiais PEMM-DMM/COPPE/UFRJ  
Caixa Postal 68505 Rio de Janeiro, RJ - CEP 21941972 - Brasil  
gustavo\_rabello@metalmat.ufrj.br

### Norberto Mangiavacchi

UERJ - Depto. de Engenharia Mecânica/GESAR  
Rua Fonseca Teles, 121 - Rio de Janeiro, RJ - CEP 20940903 - Brasil  
norberto@uerj.br

### José Pontes

UFRJ - Programa/Depto. de Eng. Metalúrgica e de Materiais PEMM-DMM/COPPE/UFRJ  
Caixa Postal 68505 Rio de Janeiro, RJ - CEP 21941972 - Brasil  
jopontes@metalmat.ufrj.br

### Cássio Botelho Pereira Soares

Furnas Centrais Elétricas S.A. - Depto de Engenharia Ambiental  
Rua Real Grandeza, 219 - Rio de Janeiro, RJ - CEP22283400 - Brasil  
cassiobp@furnas.com.br

**Resumo.** Um modelo numérico é proposto para a solução das equações de Navier-Stokes acopladas à equação de transporte de uma espécie química. O Método dos Elementos Finitos é utilizado para a discretização do problema e o paradigma da orientação a objetos, para elaboração do código numérico. A discretização espacial dos termos difusivos e da pressão é feita pelo método de Galerkin e a da derivada substancial, através de uma abordagem semi-Lagrangeana utilizando um esquema implícito por diferenças regressivas de primeira ordem. O sistema linear é resolvido pelo método da projeção discreto, baseado em decomposição LU.

**Palavras chave:** equações de Navier-Stokes, método de elementos finitos, método semi-Lagrangeano, método da projeção, programação orientada a objetos

## 1. Introdução

Inicialmente, um estudo sobre campo hidrodinâmico próximo ao eixo de um eletrodo rotatório presente em uma célula eletroquímica foi desenvolvido utilizando análise de estabilidade. Pode-se citar importantes estudos nesta área de pesquisa, tais como: Smith, 1946, Gregory *et al.*, 1955, Chin e Litt, 1972, Pontes *et al.*, 2004 e Pontes e Mangiavacchi, 2005. O presente trabalho tem como objetivo propor uma análise detalhada, sob outro ponto de vista, do campo hidrodinâmico descrito. Um modelo numérico é proposto para a solução das equações de Navier-Stokes (equações de quantidade de movimento e equação de conservação de massa) acopladas à equação de transporte. O Método dos Elementos Finitos é utilizado para a discretização do problema proposto, sendo caracterizado por 4 técnicas: *método de Galerkin*, *método semi-Lagrangeano*, *discretização do tempo por diferenças regressivas de primeira ordem* e *método da projeção para solução do sistema linear*. O método de Galerkin foi utilizado para discretização espacial dos termos difusivos, da pressão e de forçagem. A utilização do método Semi-Lagrangeano consiste na discretização da derivada substancial ( $D\mathbf{u}/Dt$ ). É nessa derivada que se encontra o termo convectivo, responsável pela não linearidade do problema. Para a discretização do tempo foi utilizada uma técnica de diferenças regressivas de primeira ordem. Depois de passar por todas essas etapas recai-se em um problema de resolução de um sistema linear do tipo  $Ax = b$ . Utilizar métodos diretos para resolução desse sistema pode parecer a opção mais fácil, porém, o uso de alguns algoritmos agilizam o processamento da solução. O método da projeção discreto é utilizado para dividir a matriz original em dois fatores do tipo LU em blocos obtendo assim

um sistema com custo computacional menor devido ao desacoplamento entre velocidade e pressão. O código numérico é implementado usando os paradigmas da orientação a objetos.

## 2. Equações de Governo

As equações de governo do campo hidrodinâmico e de concentração em sua forma adimensional utilizadas na formulação do método de elementos finitos podem ser escritas como (Batchelor, 1967; Pontes e Mangiavacchi, 2005).

$$\frac{D\mathbf{v}}{Dt} = -\frac{1}{\rho}\nabla p + \frac{1}{Re}\nabla \cdot [\nu(\nabla\mathbf{v} + \nabla\mathbf{v}^T)] \quad (1)$$

$$\nabla \cdot \mathbf{v} = 0 \quad (2)$$

$$\frac{Dc}{Dt} = \frac{1}{ReSc}\nabla \cdot (D\nabla c) \quad (3)$$

As equações (1) e (2) são conhecidas como equações de *Navier-Stokes* e a equação (3) como equação de transporte de massa. Essas equações são responsáveis pela descrição do movimento de um fluido e de uma massa respectivamente onde  $Re = \rho UL/\mu$  representa o número de Reynolds e  $Sc = \nu/D$  é o número de Schmidt,  $\mathbf{v}$  é o vetor de velocidades,  $p$  a pressão e a concentração em massa da espécie química representada pela letra  $c$ . A saber,  $\rho$ ,  $U$ ,  $L$ ,  $\mu$  e  $D$  são grandezas dimensionais representativas do problema associadas a massa específica, velocidade, comprimento, viscosidade e difusividade, respectivamente. As equações (1) e (3) são acopladas por intermédio das propriedades do fluido, que são dependentes de  $C$  e são avaliadas pelas seguintes expressões:  $\nu = \nu_0 \exp(mc)$  e  $D = 1/\nu$ .

## 3. Método dos Elementos Finitos

### 3.1. Formulação Variacional

Considere as equações de Navier-Stokes e transporte de massa para escoamentos incompressíveis dadas em sua forma adimensional representadas pelas equações (1), (2) e (3), válidas em um domínio  $\Omega \subset \mathbb{R}^m$  sujeitas às condições de contorno

$$\mathbf{v} = \mathbf{v}_\Gamma \quad \text{em } \Gamma_1 \quad (4)$$

$$\mathbf{v}_t = 0 \quad \text{e } \sigma^{nn} = 0 \quad \text{em } \Gamma_2 \quad (5)$$

$$c = c_\Gamma \quad \text{em } \Gamma_3 \quad (6)$$

$$\nabla c \cdot \mathbf{n} = 0 \quad \text{em } \Gamma_4 \quad (7)$$

Considere o subespaço:

$$\mathbb{V} = H^1(\Omega)^m = \{\mathbf{v} = (v_1, \dots, v_m) : v_i \in H^1(\Omega), \forall i = 1, \dots, m\} \quad (8)$$

onde  $H^1(\Omega)$  é o espaço de *Sobolev* dado por:

$$H^1(\Omega) = \left\{ v \in L^2(\Omega) : \frac{\partial v}{\partial x_i} \in L^2(\Omega), i = 1, \dots, m \right\} \quad (9)$$

sendo  $L^2(\Omega)$  um espaço de dimensão infinita definido por

$$L^2(\Omega) = \left\{ v : \Omega \rightarrow \mathbb{R}, \int_{\Omega} v^2 d\Omega < \infty \right\} \quad (10)$$

Definindo-se

$$\mathbb{V}_{\mathbf{v}_\Gamma} = \{\mathbf{v} \in \mathbb{V} : \mathbf{v} = \mathbf{v}_\Gamma \text{ em } \Gamma_1\}$$

$$\mathbb{P}_{p_\Gamma} = \{q \in L^2(\Omega) : q = p_\Gamma \text{ em } \Gamma_2\}$$

$$\mathbb{C}_{c_\Gamma} = \{r \in L^2(\Omega) : r = c_\Gamma \text{ em } \Gamma_3\}$$

(11)

a formulação variacional do problema consiste em encontrar soluções  $\mathbf{v}(x, t) \in \mathbb{V}_{\mathbf{v}_\Gamma}$ ,  $p(x, t) \in \mathbb{P}_0$  e  $c(x, t) \in \mathbb{C}_{c_\Gamma}$  tais que:

$$\begin{aligned}
 \int_{\Omega} \frac{D\mathbf{v}}{\partial t} \cdot \mathbf{w} d\Omega - \frac{1}{\rho} \int_{\Omega} p \nabla \cdot \mathbf{w} d\Omega + \frac{1}{Re} \int_{\Omega} \nu [\nabla \mathbf{v} + \nabla \mathbf{v}^T] : \mathbf{w} d\Omega &= 0 \\
 \int_{\Omega} [\nabla \cdot \mathbf{v}] q d\Omega &= 0 \\
 \int_{\Omega} \frac{Dc}{\partial t} r d\Omega + \frac{1}{Re Sc} \int_{\Omega} (D\nabla c) \nabla r^T d\Omega &= 0
 \end{aligned} \tag{12}$$

### 3.2. Método de Galerkin semi-discreto

Após a formulação variacional das equações de governo, parte-se para aproximação pelo método de Galerkin. Considere as equações de governo em sua forma adimensional e variacional (12). Considere  $NV$  o número de pontos de velocidade,  $NP$  o número de pontos de pressão e  $NE$  o número de elementos na malha de elementos finitos que discretiza o domínio  $\Omega$ . O método de *Galerkin* consiste em fazer as seguintes substituições em (12):

$$u(\mathbf{x}, t) \approx \sum_{n=1}^{NV} N_n(\mathbf{x}) u_n(t) \tag{13}$$

$$v(\mathbf{x}, t) \approx \sum_{n=1}^{NV} N_n(\mathbf{x}) v_n(t) \tag{14}$$

$$p(\mathbf{x}, t) \approx \sum_{n=1}^{NP} P_n(\mathbf{x}) p_r(t), \tag{15}$$

que são aproximações semi-contínuas, isto é, contínuas no tempo ( $t$ ) e discretas no espaço ( $x$ ). Aqui,  $N_n(x)$  representam as funções de interpolação utilizadas para a velocidade e  $P_n(x)$  as funções de interpolação para a pressão. Considere ainda que as propriedades  $\rho$ ,  $\nu$  e  $D$  são constantes em cada elemento.

A equação de conservação de quantidade de movimento é normalmente avaliada em todos os nós livres de velocidade, e portanto, as funções peso  $w_x$  e  $w_y$  são substituídas por funções de interpolação  $N_m = N_m(x)$ ,  $m = 1, \dots, NV$ . Aplicando este procedimento nas equações (12), chega-se a:

$$\begin{aligned}
 \sum_e \int_{\Omega^e} \sum_n \frac{Du_n}{Dt} N_m N_n d\Omega - \sum_e \int_{\Omega^e} \sum_r \frac{\partial N_m}{\partial x} P_r p_r d\Omega \\
 + \frac{1}{Re} \sum_e \int_{\Omega^e} \sum_n \nu^e \left( \frac{\partial N_m}{\partial x} u_n \frac{\partial N_n}{\partial x} + \frac{\partial N_m}{\partial y} \frac{\partial N_n}{\partial y} u_n + \frac{\partial N_m}{\partial x} \frac{\partial N_n}{\partial x} u_n + \frac{\partial N_m}{\partial y} \frac{\partial N_n}{\partial x} v_n \right) d\Omega = 0
 \end{aligned} \tag{16}$$

$$\begin{aligned}
 \sum_e \int_{\Omega^e} \sum_n \frac{Dv_n}{Dt} N_m N_n d\Omega - \sum_e \int_{\Omega^e} \sum_r \frac{\partial N_m}{\partial x} P_r p_r d\Omega \\
 + \frac{1}{Re} \sum_e \int_{\Omega^e} \sum_n \nu^e \left( \frac{\partial N_m}{\partial x} v_n \frac{\partial N_n}{\partial x} + \frac{\partial N_m}{\partial y} \frac{\partial N_n}{\partial y} v_n + \frac{\partial N_m}{\partial x} \frac{\partial N_n}{\partial y} v_n + \frac{\partial N_m}{\partial y} \frac{\partial N_n}{\partial y} v_n \right) d\Omega = 0
 \end{aligned} \tag{17}$$

$$\sum_e \int_{\Omega^e} \sum_n \frac{Dc_n}{Dt} N_m N_n d\Omega + \frac{1}{Re Sc} \sum_e \int_{\Omega^e} \sum_n D^e \left( \frac{\partial N_m}{\partial y} \frac{\partial N_n}{\partial y} c_n + \frac{\partial N_m}{\partial x} \frac{\partial N_n}{\partial x} c_n \right) d\Omega = 0 \tag{18}$$

A equação da continuidade (2), é avaliada nos nós livres de pressão e, portanto, a função peso  $q$  é aproximada pelas funções de interpolação associadas à pressão  $P_r(x)$ , resultando

$$\sum_e \int_{\Omega^e} \sum_n \left( \frac{\partial N_n}{\partial x} u_n + \frac{\partial N_n}{\partial y} v_n \right) P_r d\Omega = 0. \tag{19}$$

para  $r = 1, \dots, NP$ . Restringindo as funções de interpolação a cada elemento  $e$ , conclui-se

$$\sum_e \int_{\Omega^e} \sum_{j,k \in e} \left( \frac{\partial N_j^e}{\partial x} u_j + \frac{\partial N_j^e}{\partial y} v_j \right) P_k^e d\Omega = 0. \tag{20}$$

As equações (16), (17), (20) e (18) podem ser representadas na forma de um sistema de equações diferenciais ordinárias

$$M_x \dot{u} + \frac{1}{Re} \{ (2K_{xx} + K_{yy})u + K_{xy}v \} - G_x p = 0 \tag{21}$$

$$M_x \dot{v} + \frac{1}{Re} \{ K_{yx}u + (K_{xx} + 2K_{yy})v \} - G_y p = 0 \tag{22}$$

$$D_x u + D_y v = 0 \tag{23}$$

$$M_c \dot{c} + \frac{1}{Re Sc} (K_{xx} + K_{yy})c = 0 \tag{24}$$

onde  $\dot{u}$ ,  $\dot{v}$  e  $\dot{c}$  representam a derivada substancial, sendo definidas por  $\dot{u} = [\frac{Du_1}{Dt}, \dots, \frac{Du_{NV}}{Dt}]^T$ ,  $\dot{v} = [\frac{Dv_1}{Dt}, \dots, \frac{Dv_{NV}}{Dt}]^T$ ,  $\dot{c} = [\frac{Dc_1}{Dt}, \dots, \frac{Dc_{NV}}{Dt}]^T$ ,  $u = [u_1, \dots, u_{NV}]^T$ ,  $v = [v_1, \dots, v_{NV}]^T$ ,  $c = [c_1, \dots, c_{NV}]^T$ ,  $p = [p_1, \dots, p_{NP}]^T$ ,  $g_x = [g_1^x, \dots, g_{NV}^x]^T$ ,  $g_y = [g_1^y, \dots, g_{NV}^y]^T$ , , são os vetores dos valores nodais para as variáveis de velocidade, pressão e concentração.

As dimensões das matrizes que aparecem no sistema (24) são  $NV \times NP$  para  $G_x$  e  $G_y$ ,  $NP \times NV$  para  $D_x$  e  $D_y$  e  $NV \times NV$  para todas as outras.

### 3.3. Método semi-Lagrangeano

O método *semi-Lagrangeano* foi primeiramente utilizado em sistemas convecção-difusão com o objetivo de se obter duas características: *passo de tempo grande* e *estabilidade*. Além disso, discretizações em ordens elevadas levam a erros de dispersão mínimos. A aproximação semi-Lagrangeana tem sido utilizada em meteorologia para predições numéricas das condições climáticas, onde o uso de grandes passos de tempo é essencial para eficiência. Nas equações de Navier-Stokes, entretanto, seu uso não é tão freqüente, porém, trabalhos recentes vêm demonstrando sua elevada eficiência, principalmente quando o escoamento é caracterizado por alto número de Reynolds.

Esse método foi introduzido no início dos anos 80 por Robert, 1981 e Pironneau, 1982 e a idéia básica procedia na discretização da solução da derivada Lagrangeana no tempo ao invés da derivada Euleriana. Como exemplo, pode-se citar um esquema semi-Lagrangeano de uma equação do tipo convecção-difusão qualquer. Toma-se a equação de transporte de massa (3) em sua forma compacta, com derivada substancial. Seguindo o método semi-Lagrangeano, pode-se discretizar a equação (3) no tempo no ponto  $x_i$  utilizando um esquema de primeira ordem implícito

$$\frac{Dc}{Dt} = \frac{c_i^{n+1} - c_d^n}{\Delta t} \tag{25}$$

onde  $c_d^n = c^n(x_d, t^n)$  e  $x_d$  é chamado de ponto de saída.

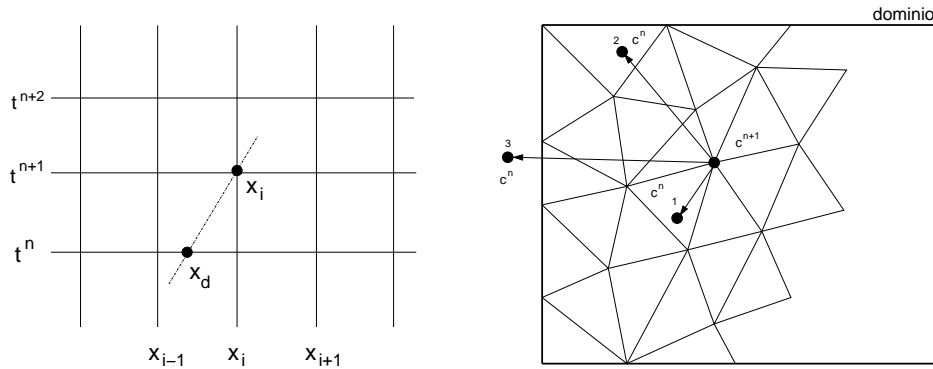


Figura 1: método semi-lagrangeano

Na forma forte a derivada substancial é calculada ao longo do trajeto característico, determinando-se o ponto  $x_d$  e resolvendo a equação  $Dx/Dt = a$  para trás no tempo  $t^{n+1} \geq t \geq t^n$  usando a condição inicial  $x(t^{n+1}) = x_i$ . Este esquema é mostrado na figura (1a). Um método de integração deve ser utilizado para se encontrar a posição do ponto do passo anterior na malha. Utilizando-se um esquema de discretização de primeira ordem, a trajetória é aproximada por uma reta. Dependendo da trajetória, três situações podem ocorrer, a primeira e a segunda são mostradas na figura (1b) pelos pontos 1 e 2, e a terceira, pelo ponto 3. Na trajetória 1, o ponto do passo anterior  $c^n$  se encontra próximo ao ponto do passo atual  $c^{n+1}$  e dentro do domínio do problema. Depois de identificado o elemento que contém o ponto 1, uma interpolação entre os nós do elemento é necessária para se conhecer seu valor. No caso do ponto 2, o ponto do passo anterior  $c^n$  se encontra distante do ponto do passo atual  $c^{n+1}$  e dentro do domínio do problema. A diferença entre o ponto

1 e o ponto 2 está no comprimento das trajetórias. No ponto 2, a trajetória também será aproximada por reta o que pode gerar um grande erro de aproximação, pois com pouca informação (tempos inicial e final) não se sabe a real trajetória do ponto. No cado ponto 3, o valor de origem é calculado utilizando as condições de contorno. Após a discretização pelo método semi-Lagrangeano, as equações de governo tomam a forma de:

$$M_x \left( \frac{u_i^{n+1} - u_d^n}{\Delta t} \right) + \frac{1}{Re} \{ (2K_{xx} + K_{yy})u^{n+1} + K_{xy}v^{n+1} \} - G_x p^{n+1} = 0 \quad (26)$$

$$M_y \left( \frac{v_i^{n+1} - v_d^n}{\Delta t} \right) + \frac{1}{Re} \{ K_{yx}u^{n+1} + (K_{xx} + 2K_{yy})v^{n+1} \} - G_y p^{n+1} = 0 \quad (27)$$

$$D_x u^{n+1} + D_y v^{n+1} = 0 \quad (28)$$

$$M_c \left( \frac{c_i^{n+1} - c_d^n}{\Delta t} \right) + \frac{1}{Re Sc} (K_{c_{xx}} + K_{c_{yy}})c^{n+1} = 0 \quad (29)$$

### 3.4. Elementos de Malha

A escolha do tipo de elementos em equações onde o acoplamento de variáveis existe, como no caso das equações de Navier-Stokes, é restrito. Na literatura, essa restrição recebe o nome de condição de *Babuska-Brezzi*. Essa é uma condição de estabilidade referente ao modo de discretização do problema, portanto, a escolha de elementos adequados é fundamental para manter essa condição satisfeita. Muitos autores mencionam e detalham essa condição específica. Para se saber mais, recomendam-se: Cuvelier *et al.*, 1986, Zienkiewicz e Taylor, 2000 e Oden e Carey, 1984. Há na literatura casos onde se encontra a solução do problema sem mesmo a condição de *Babuska-Brezzi* ser satisfeita, mas para tais elementos o método de *Galerkin* não pode ser utilizado. Métodos de penalidades também podem ser usados para contornar essa restrição, porém não é assunto deste trabalho o uso desta prática.

**Elemento triangular linear:** Esse é o elemento mais utilizado no método dos elementos finitos, por sua forma geométrica mais simples, podendo ser usado para aproximar superfícies irregulares. Por ser linear, suas funções de interpolação são representadas por planos (ver Fig. 2), ou seja, um polinômio de grau 1. A pressão na equação de Navier-Stokes (equação 1) e a concetração na equação de transporte de massa (equação 3) foram discretizadas utilizando esse elemento.

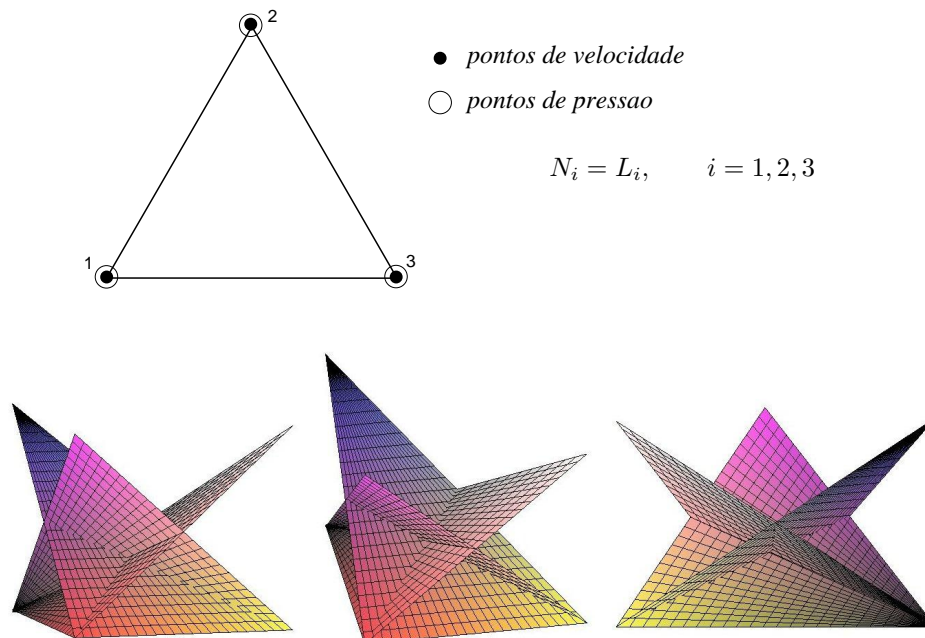


Figura 2: representação das funções de interpolação para o triângulo linear

**Elemento triangular cúbico:** Este elemento apresenta um grau de liberdade a mais para velocidade localizado no centróide do triângulo. A função de interpolação recebe o nome especial de *função bolha*, pois aparece uma bolha localizada no interior do elemento, como pode ser visto na figura (3b). A combinação de funções de interpolação linear para pressão e de funções de interpolação cúbicas para a velocidade forma o que é conhecido na literatura por elemento *MINI*. Sendo um elemento cúbico, seu polinômio de interpolação é de grau 3. As velocidades das equações de *Navier-Stokes* foram calculadas utilizando esse elemento.

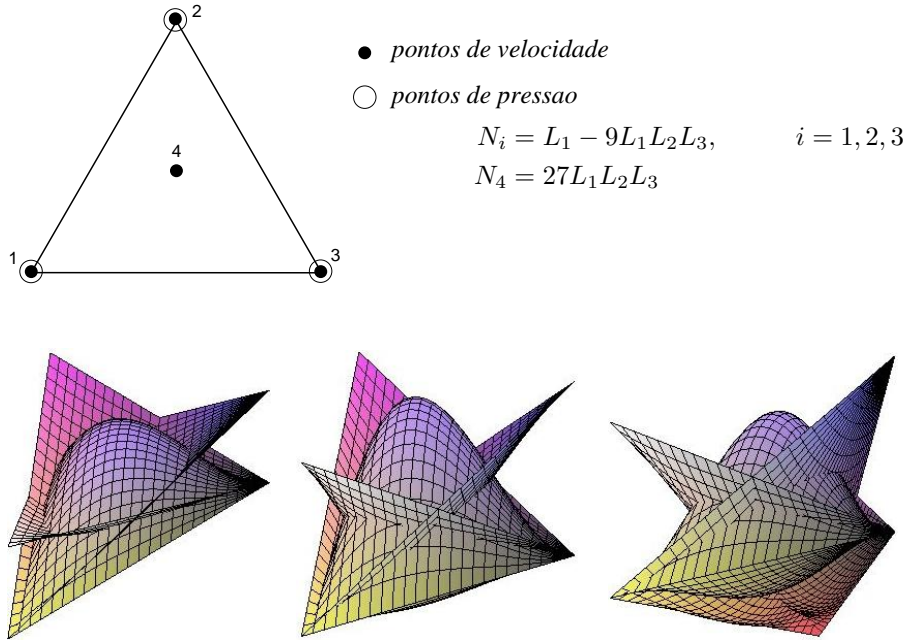


Figura 3: representação das funções de interpolação para o triângulo MINI

### 3.5. Método da Projeção Discreto

O método da projeção discreto baseado em decomposição *LU* é obtido através de fatoração em blocos do sistema linear resultante. Isto sugere que a separação (ou *split*) entre velocidade e pressão é feita depois da discretização no espaço e no tempo das equações de governo. Considere as equações discretizadas no tempo e no espaço como se segue:

$$\begin{aligned}
 M\left(\frac{u^{n+1} - u_d^n}{\Delta t}\right) + \frac{1}{Re}Ku^{n+1} - Gp^{n+1} &= 0 \\
 Du^{n+1} &= 0 \\
 M\left(\frac{c_i^{n+1} - c_d^n}{\Delta t}\right) + \frac{1}{Re Sc}Kc^{n+1} &= 0
 \end{aligned} \tag{30}$$

a última equação de (30) pode ser resolvida separadamente, no entanto, as equações restantes formam um sistema de equações que pode ser representado por:

$$\begin{bmatrix} B & -\Delta tG \\ D & 0 \end{bmatrix} \cdot \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} r^n \\ 0 \end{bmatrix} + \begin{bmatrix} bc_1 \\ bc_2 \end{bmatrix} \tag{31}$$

onde agora o sistema é escrito apenas para as incógnitas do problema, ou seja,  $u^{n+1} = [u_1^{n+1}, \dots, u_{Nu}^{n+1}, v_1^{n+1}, \dots, v_{Nv}^{n+1}]^T$ ,  $p^{n+1} = [p_1^{n+1}, \dots, p_{Np}^{n+1}]^T$ , sendo *Nu*, *Nv* e *Np* o número de incógnitas (nós livres) para velocidade na direção *x*, velocidade na direção *y* e pressão respectivamente. A notação para as matrizes e vetores foi mantida a mesma por simplicidade. A matriz *B* é dada por

$$B = M + \frac{\Delta t}{Re}K \tag{32}$$

e o lado direito representa as grandezas conhecidas no tempo *n*,

$$r^n = -\Delta t(Au_d^n) + Mu^n, \tag{33}$$

mais as condições de contorno que nada mais são do que as contribuições dos valores conhecidos de velocidade e pressão no lado direito do sistema.

O método da projeção baseado em fatoração *LU* visa decompor a matriz do sistema (31) através de uma fatoração por blocos. Em Lee *et al.*, 2001 são apresentadas várias formas de se fatorar esta matriz, cada forma dando origem a uma família de métodos diferentes. Utilizando uma fatoração canônica *LU* por blocos, tem-se o seguinte sistema:



$$\begin{bmatrix} B & 0 \\ D & \Delta t D B^{-1} G \end{bmatrix} \cdot \begin{bmatrix} I & -\Delta t B^{-1} G \\ 0 & I \end{bmatrix} \cdot \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} r^n \\ 0 \end{bmatrix} + \begin{bmatrix} bc_1 \\ bc_2 \end{bmatrix} \quad (34)$$

O sistema apresentado em (34), se resolvido, dá origem ao método de *Uzawa* (Chang *et al.*, 2002). Porém sua solução é cara computacionalmente devido à inversão da matriz  $B$  a cada iteração. Duas técnicas foram utilizadas neste trabalho, todas utilizando um processo de aproximação chamado *lumping*. Na primeira técnica, a matriz  $B^{-1}$  foi aproximada por uma matriz de massa diagonal  $B_L^{-1}$ , neste caso alguns oscilações foram encontradas nos campos de velocidade para número de  $Re$  baixo. Portanto, uma outra solução resolveu esse problema, em vez de aproximar a matriz de massa, a diagonalização foi feita em  $B$ . Para este caso a solução não apresenta problemas para nenhum valor de  $Re$ . A matriz diagonalizada envolve uma aproximação conhecida na literatura por *lumping*. Esta técnica consiste em somar todos os elementos de linha e localizá-los na diagonal principal da matriz. O mesmo procedimento pode ser feito em elementos de coluna já que a matriz  $B$  é simétrica. Note que a utilização desta técnica se faz necessária pois a inversão de matrizes não-diagonais gera custos computacionais elevados. No métodos dos elementos finitos, a técnica de *Lumping* tem sua principal utilização na matriz de massa, pois ao final do procedimento, a massa total do elemento se mantém conservada.

### 3.6. Orientação a Objetos

O paradigma da orientação a objetos é empregado para a elaboração do código numérico. As vantagens deste tipo de programação comparadas à programação estrutural (convencional) são: facilidade de manutenção, modularidade, implementação de classes e objetos, abstração de dados, etc. A figura (4) representa o diagrama de classes em UML (Unified Modeling Language) do código desenvolvido neste trabalho.

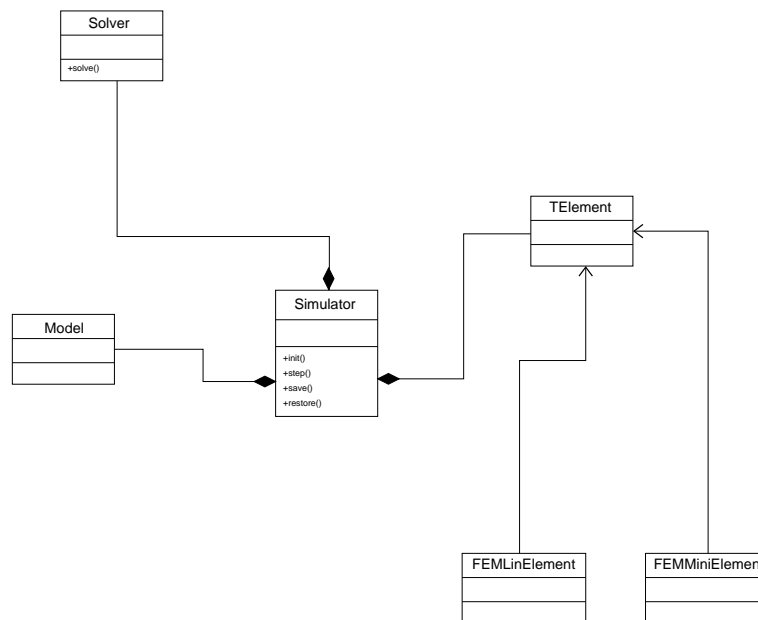


Figura 4: diagrama de classes

Na classe *Model* encontram-se as condições de contorno, a discretização do domínio e a geometria do problema. Em *FEMMiniElement* e *FEMLinElement* encontram-se as informações sobre os elementos utilizados e suas respectivas matrizes, tais como: matriz de rigidez, matriz de massa e matriz de operadores para o cálculo da vorticidade. A classe *TElement* está preparada para capturar as informações das classe *FEMMiniElement* e *FEMLinElement* ou de uma outra classe com tipo de elemento diferente. A classe *Simulator* é responsável pela montagem das matrizes (*Assemble*), aplicação das condições de contorno e preparação do sistema linear a ser resolvido. Logo, este sistema é encaminhado para a classe *Solver* onde a solução do problema é processada.

#### 4. Resultados

O método foi validado para o caso de escoamento plano de Poiseuille estacionário, produzindo valores nodais virtualmente exatos. O problema do degrau, ou em inglês, *backward-step* foi utilizado para ilustrar as características do método desenvolvido. As condições de contorno e geometrias são apresentadas na figura (5).

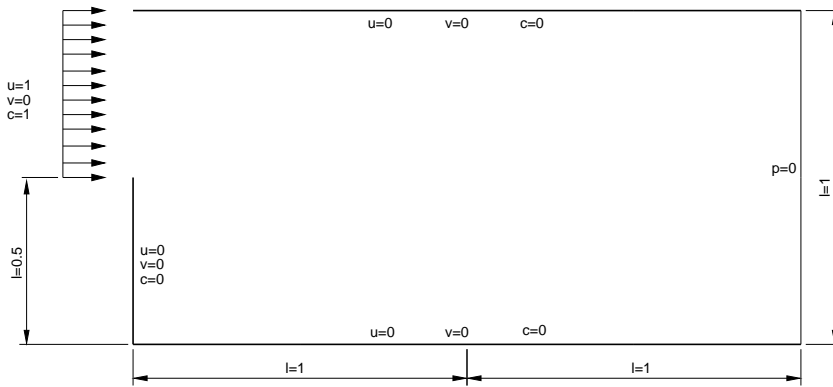


Figura 5: condições de contorno para o problema proposto

Os resultados mostrados nas figuras (6), (7) e (8) a seguir foram retirados de três situações caracterizadas em seqüência. A primeira imagem de cada linha representa uma condição logo após a condição inicial (2 iterações). A segunda figura de cada linha representa um estágio intermediário, de aproximadamente 8 iterações. A terceira figura representa um estado próximo ao permanente. Foram testadas combinações dos números de  $Re$  e  $Sc$ . A seguir são mostrados os casos  $Re = 10$  e  $Sc = 10$  (figura 6),  $Re = 100$  e  $Sc = 100$  (figura 7) e  $Re = 10000$  e  $Sc = 100$  (figura 8), usando uma malha triangular com 800 vértices.

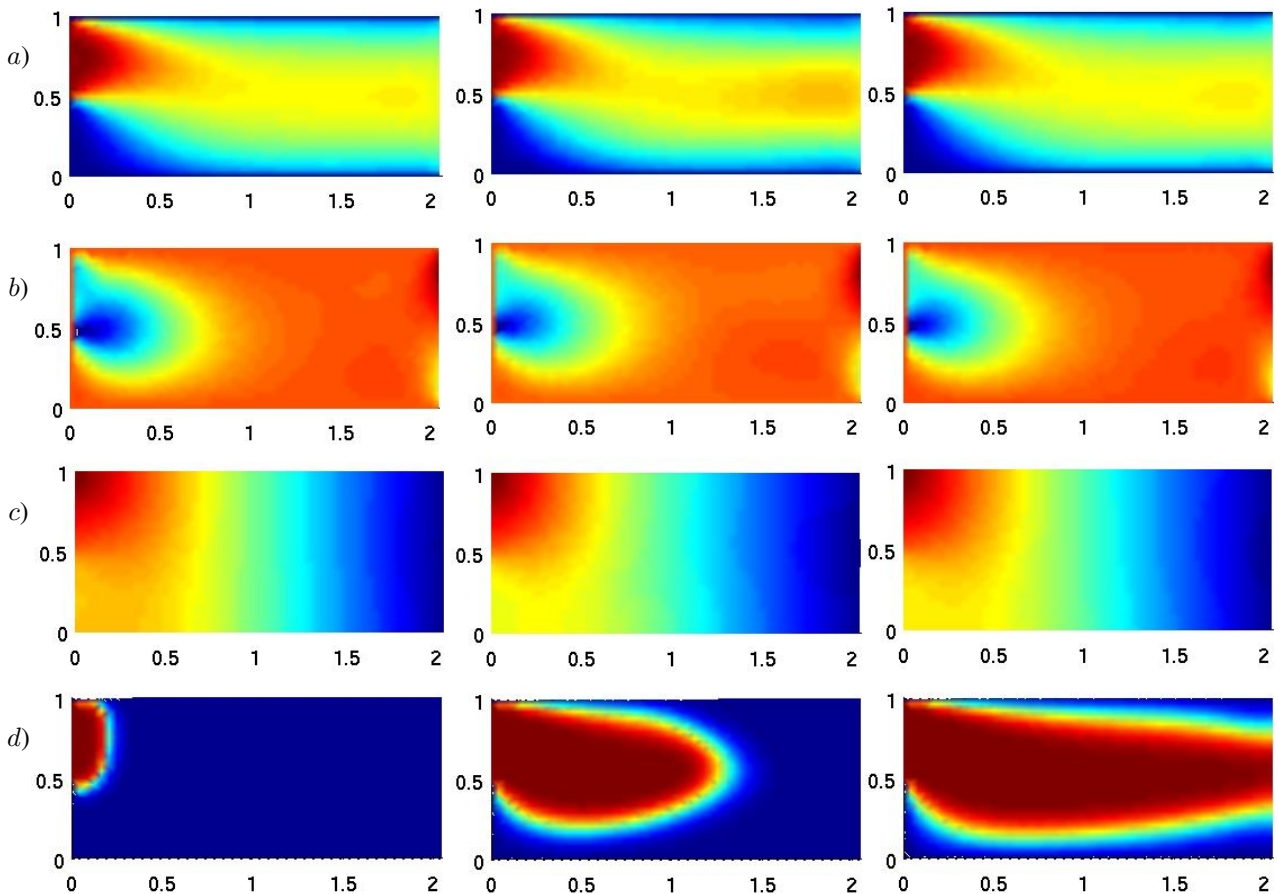


Figura 6: Simulação de  $Re = 10$  e  $Sc = 10$ . Evolução das componentes  $u$  (a) e  $v$  (b) da velocidade, da pressão (c) e da concentração da espécie química (d).

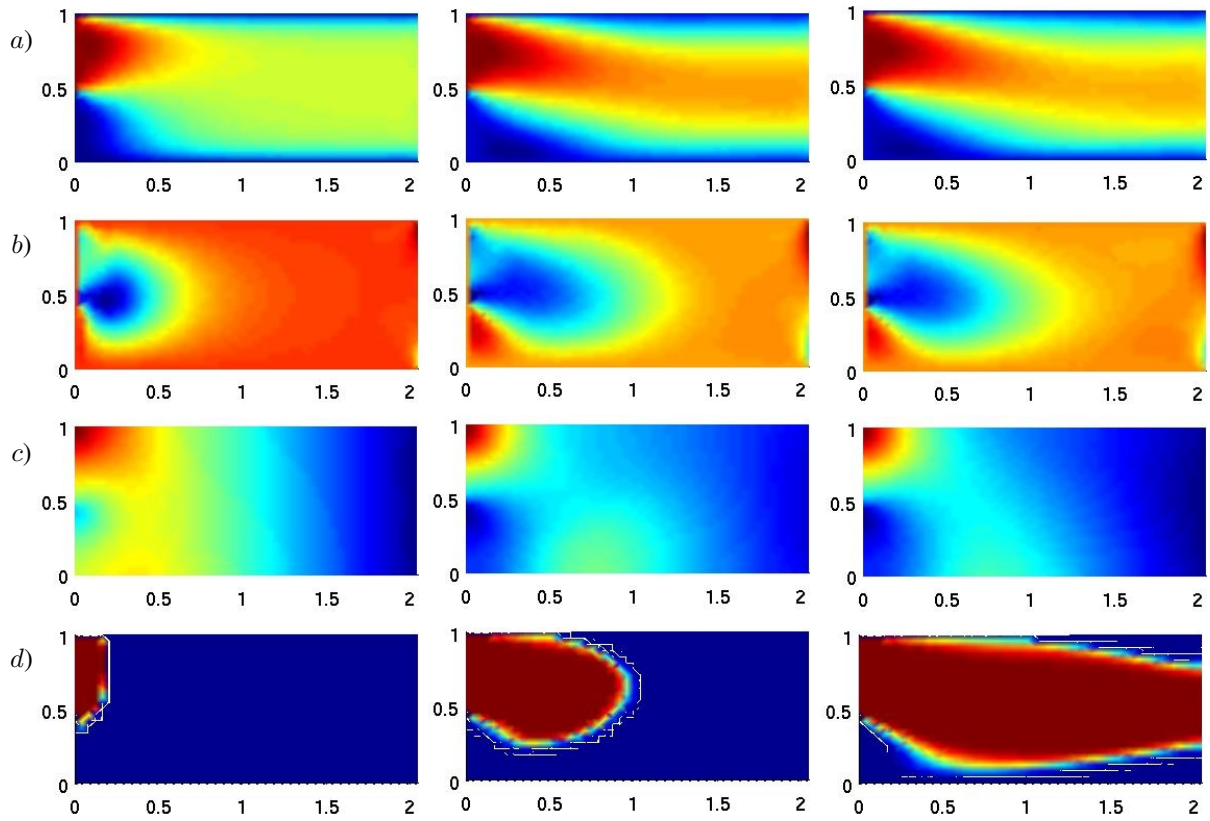


Figura 7: Simulação de  $Re = 100$  e  $Sc = 100$ . Evolução das componentes  $u$  (a) e  $v$  (b) da velocidade, da pressão (c) e da concentração da espécie química (d).

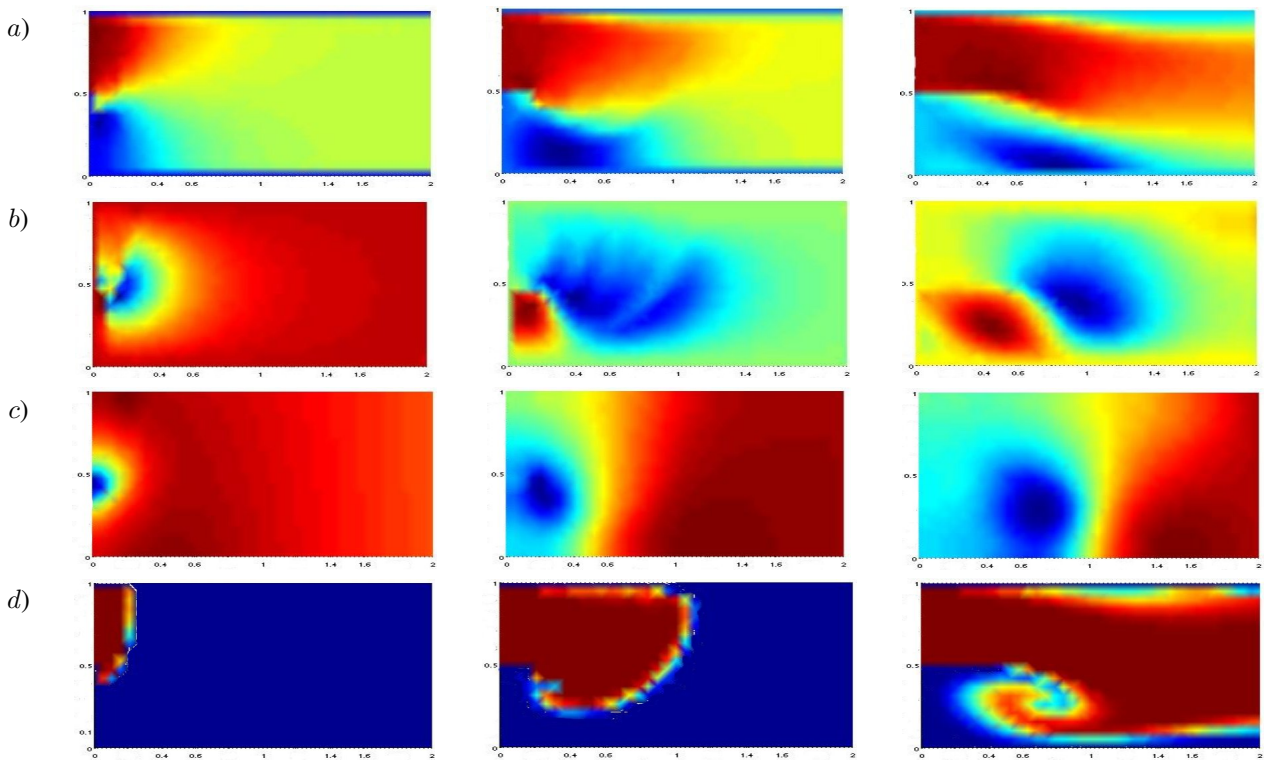


Figura 8: Simulação de  $Re = 10000$  e  $Sc = 100$ . Evolução das componentes  $u$  (a) e  $v$  (b) da velocidade, da pressão (c) e da concentração da espécie química (d).

Os resultados mostram que o método é estável, mesmo para passos de integração no tempo longos ( $CFL = \Delta t U / \Delta x \approx 5$ ), e não apresenta oscilações mesmo para valores de  $Re$  e  $Sc$  elevados.

O efeito da variação do  $Re$  pode ser apreciado na redução da espessura da camada limite hidrodinâmica e no crescimento da região de recirculação a jusante do degrau com o aumento do  $Re$ .

O efeito da variação do  $Sc$  se manifesta também pela redução da espessura da camada limite de concentração e o afinamento do jato na região a jusante do degrau com o aumento do produto  $Re Sc$ .

Em particular, pode-se observar que para  $Re Sc = 1000000$  a frente de propagação do campo escalar  $c$  é muito bem definida, mostrando que o método numérico apresenta baixa difusão artificial e estabilidade.

## 5. Conclusão

O problema tratado nesse trabalho apresenta a originalidade de abordar o estudo do campo hidrodinâmico afetado pelo acoplamento, através dos coeficientes de viscosidade e de difusão dependentes da concentração da espécie, com a dinâmica de transporte de uma espécie química.

O método dos elementos finitos proposto nesse trabalho, baseado no método de Galerkin e na formulação semi-Lagrangeana, se mostrou estável para todos os valores de  $CFL$ ,  $Re$  e  $Sc$  analisados. Os resultados não apresentaram oscilações espúrias e nem difusão numérica excessiva nas simulações para altos valores de  $Re$  e  $Sc$ . Essas características estão associadas à escolha da formulação semi-Lagrangeana em combinação com o elemento *MINI* para a discretização das equações de *Navier-Stokes* e o elemento linear para a discretização da equação de transporte da espécie química.

O método de projeção discreto, baseado em fatoração LU, combinado com a técnica de aproximação *Lumped* resultou em um esquema numérico computacionalmente eficiente, permitindo a utilização de métodos de solução de sistemas lineares com matrizes de coeficientes simétricas e positivas definidas como o método dos gradientes conjugados com pré-condicionador de *Cholesky* incompleto. A utilização do paradigma de orientação a objetos permitiu a implementação de um sistema de simulação de qualidade do ponto de vista de engenharia de *software*, de fácil manutenção e desenvolvimento futuro.

## 6. Agradecimentos

Agradecemos a Furnas Centrais Elétricas S.A., ao CNPQ e a FAPERJ pelo suporte financeiro ao desenvolvimento deste trabalho.

## 7. Referências

- Batchelor, G. K., 1967, “An Introduction do Fluid Dynamics”, Cambridge University Press.
- Chang, W., Giraldo, F., and Perot, B., 2002, Analysis of an Exact Fractional Step Method, “Journal of Computational Physics”.
- Chin, D. T. and Litt, M., 1972, An Electrochemical Study of Flow Instability on a Rotating Disk, “J. Fluid Mech.”, Vol. 54, pp. 613–625.
- Cuvelier, C., Segal, A., and van Steenhoven, A. A., 1986, “Finite Element Method and Navier-Stokes Equations”, Dordrecht, Holland.
- Gregory, N., Stuart, J. T., and Walker, W. S., 1955, On the stability of three-dimensional boundary layers with application to the flow due to a rotating disk, “Phil. Trans. Roy. Soc. London”, Vol. A-248, pp. 155–199.
- Lee, M. J., Oh, B., and Kim, Y. B., 2001, Canonical fractionalstep methods and consistent boundary conditions for the incompressible Navier-Stokes equations, “Journal of Computational Physics”.
- Oden, J. T. and Carey, G., 1984, “Finite Elements: Mathematical Aspects”, Prentice-Hall, vol. iv edition.
- Pironneau, O., 1982, On the transpor-diffusion algorithm and its applications to the Navier-Stokes equation, “Numer. Math”, pp. 38–309.
- Pontes, J. and Mangiavacchi, N., 2005, Fenômentos de Transferência, Apostila.
- Pontes, J., Mangiavacchi, N., Conceição, A. R., Barcia, O. E., Mattos, O. E., and Tribollet, B., 2004, Rotating Disk Flow Stability in Electrochemical Cells: Effect of Viscosity Stratification, “Phys. Fluids”, Vol. 16, No. 3, pp. 707–716.
- Robert, A., 1981, A stable numerical integration scheme for the primitive meteorologica equations, “Atmos. Ocean”, pp. 19–35.
- Schlichting, H., 1979, “Boundary Layer Theory”, McGraw-Hill Science.
- Smith, N., 1946, Exploratory investigation of laminar boundary layer oscillations on a rotating disk, Technical report, NACA, Technical Report TN-1227.
- Zienkiewicz, O. C. and Taylor, R. L., 2000, “The Finite Element Method Volume 1: The Basis”, Wiley John and Sons, 5th edition.

## Numerical Solution of Navier-Stokes Equations Coupled to Chemical Species by the Finite Element Method

### Gustavo Rabello dos Anjos

UFRJ - Programa/Depto. de Eng. Metalúrgica e de Materiais PEMM-DMM/COPPE/UFRJ  
Caixa Postal 68505 Rio de Janeiro, RJ - CEP 21941972 - Brasil  
gustavo\_rabello@metalmat.ufrj.br

### Norberto Mangiavacchi

UERJ - Depto. de Engenharia Mecânica/GESAR  
Rua Fonseca Teles, 121 - Rio de Janeiro, RJ - CEP 20940903 - Brasil  
norberto@uerj.br

### José Pontes

UFRJ - Programa/Depto. de Eng. Metalúrgica e de Materiais PEMM-DMM/COPPE/UFRJ  
Caixa Postal 68505 Rio de Janeiro, RJ - CEP 21941972 - Brasil  
jopontes@metalmat.ufrj.br

### Cássio Botelho Pereira Soares

Furnas Centrais Elétricas S.A. - Departamento de Engenharia Ambiental  
Rua Real Grandeza, 219 - Rio de Janeiro, RJ - CEP 22283400 - Brasil  
cassiobp@furnas.com.br

**Abstract.** *A numerical model is proposed for solving the Navier-Stokes equations coupled to the transport equation of a chemical species. The Finite Element Method is employed in the problem discretization and the object-oriented paradigm, in the development of the numerical code. Spatial discretization of the diffusion and pressure terms is made through the Galerkin method and the substantial derivative, through a semi-Lagrangian technique, using a first-order backward Euler implicit scheme. The linear system is solved through the discrete projection method, based on LU decomposition..*

**Keywords** *Navier-Stokes equations, finite element method, semi-Lagrangian method, projection method, object-oriented programming*

## LARGE EDDY SIMULATION OF A FLOW OVER A BACKWARD-FACING STEP USING A CVFEM-FORMULATION

Rosiane Cristina de Lima<sup>1</sup>

[rosiane@ita.br](mailto:rosiane@ita.br)

<sup>1</sup> Instituto Tecnológico de Aeronáutica, Departamento de Engenharia Aeronáutica e Mecânica, Área de Aerodinâmica, Propulsão e Energia – Praça Marechal Eduardo Gomes, 50, Vila das Acácias, CEP 12228-900, São José dos Campos, SP

João Batista Campos Silva<sup>2</sup>

[jbcampos@dem.feis.unesp.br](mailto:jbcampos@dem.feis.unesp.br)

<sup>2</sup> UNESP-Faculdade de Engenharia de Ilha Solteira, Departamento de Engenharia Mecânica, Área de Ciências Térmicas, Av. Brasil, 56 – Ilha Solteira -SP

*Abstract: The turbulent flow over a backward-facing step has been investigated by many researchers due to its several applications in engineering problems, as: diffusers, airfoils with separation, buildings, combustors and turbines blades. This problem is largely used to validate numerical codes due to simplicity geometric. It's important to supply data that can be used to ascertain turbulence models, wall laws and their implementation in computational codes. The main goal of this work is to verify the performance of the large-eddy methodology, which was implemented in a nine-node CVFEM-code. In this case none wall law was utilized. The domain is discretized using nine-node finite elements and the equations are integrated into control volumes around the nodes of the finite elements. The Navier-Stokes equations are filtered for large scale simulations and the sub-grid scale stress, which appear due to the filtering process, are modeled by the eddy viscosity model of Smagorinsky. The principal parameter evaluated in this problem is the reattachment length value and the appearing of a secondary recirculation at the corner eddy. The results are compared to experimental data available in the literature presenting good agreement.*

**Keywords:** backward-facing step, numerical simulation, CVFEM, large eddy simulation, Smagorinsky sub-grid model

### 1. Introduction

The main goal of this work is to simulate numerically turbulent flow over a backward-facing step by a control volume finite element method (CVFEM) with large-eddy simulation methodology (LES). The computational domain is discretized with nine-node quadrilateral finite elements.

The first CVFEM was presented by Baliga and Patankar (1980) for triangular elements and afterwards Raw and Schneider (1986) used the method, therefore using quadrilateral elements. Since then, several other authors have improved and applied the CVFEM in a vast number of problems in engineering.

Saabas (1991) developed a CVFEM to solve steady state flow problems in multi-dimensional domains using three-node triangular elements and four-node tetrahedron elements. According to the author CVFEM offers the combination of geometric flexibility of the finite element method (FEM) and the ease of physical interpretation associated with the commonly used Finite Volume Method (FVM).

Recently, Lima, Campos-Silva and Mansur (2004) enhanced the CVFEM code built by Campos-Silva (1998) including a Large Eddy Simulation (LES) methodology with the Smagorinsky viscosity model to simulate turbulent fluid flows. In that work the authors presented a lid-driven cavity using nine-node quadrilateral elements to discretize the domain presenting excellent agreement to literature results.

The first applications on LES in engineering were present by Deardorff (1970), in the investigation of a turbulent flow inside a channel with high Reynolds number. Initially, LES was used to perform basic flows in simple geometries; however, due to the rapid development of computational power, it has been applied to practical engineering flows in relatively complex geometries (Hamba, 2003; Löhner, 2001).

In this study will be analyzed the turbulent flow over a backward-facing step. Such problem has been investigated by many researchers due to its several applications in engineering problems, as: diffusers, airfoils with separation, buildings, combustors and turbines blades. According to Eaton and Johnston (1981), who performed an experimental study about this case, the backward-facing step offers one of the least complex separating and reattaching flows. The separation line of the backward-facing step flow is straight and fixed at the step edge; therefore the process of separation-reattachment can be examined without any complexities resulting from motion of the separation point. There is only one separated flow zone with two opposing eddies.

The main parameters evaluated in the step flow are the reattachment length value ( $L_R/h$ ) and the appearing of a secondary recirculation at corner eddy. The results are compared to numerical and experimental results from literature.

In next sections of this work, it will be reported the mathematical model, a summary about the used numerical method, the problem description and the obtained results.

## 2. Mathematical Model

### 2.1 Governing Equations

The following equations set describe the mathematical modeling of a turbulent fluid flow, with constant properties. According to Lima (2005) the dimensionless equations for large-eddy simulation are:

$$\frac{\partial U_i}{\partial t} + \frac{\partial(U_j U_i)}{\partial X_j} = -\frac{\partial P_t}{\partial X_i} + \frac{\partial}{\partial X_j} \left( \left( \frac{1}{\text{Re}} + \nu_t \right) \frac{\partial U_i}{\partial X_j} \right) + \frac{\partial}{\partial X_j} \left( \nu_t \frac{\partial U_j}{\partial X_i} \right) + F_i \quad (1)$$

$$\frac{\partial U_i}{\partial X_i} = 0 \quad (2)$$

The pressure term in Eq. (1) includes the kinetic turbulent energy,  $P_t = \bar{P} + \frac{2}{3} \frac{k}{u_0^2}$ .

The dimensionless variables, in Eqs. (1) and (2), are defined as follows:

$$X_i = \frac{x_i}{L}; U_i = \frac{u_i}{u_0}; \bar{P} = \frac{p - p_0}{\rho u_0^2}; t = \frac{t^*}{L/u_0}; \nu_t = \frac{\nu_t^*}{u_0 L} = \left( C_s \frac{\Delta}{L} \right)^2 \left( 2\bar{S}_{kl}\bar{S}_{kl} \right)^{1/2}; \text{Re} = \frac{\rho u_0 L}{\mu} \quad (3)$$

where,

- $C_s$ : Smagorinsky constant;
- $L$ : reference length;
- $p, p_0$ : static pressure and reference pressure, respectively;
- $t$ : non-dimensional time;
- $u_0$ : reference velocity;
- $U_i$ : velocities;
- $X_i$ : coordinates;
- $\mu$ : dynamic fluid viscosity;
- $\nu_t$ : kinematics turbulent fluid viscosity;
- $\rho$ : fluid density;
- $\Delta$ : filter width;
- $*$ : dimensional variables.

In the large-eddy simulation methodology the physical quantities such as velocity and pressure are decomposed in large grid-scale (LGS) and sub-grid scale (SGS) components through a filtering process. The filter function used in this case was set be equal to the length of a control volume face where are considered the convective and diffusive fluxes. The filter function most common is the filter function by volume. The filter width,  $\Delta$ , is generally defined as:  $\Delta = (\Delta_x \Delta_y \Delta_z)^{1/3}$  in 3D and  $\Delta = (\Delta_x \Delta_y)^{1/2}$  in 2D problems with  $\Delta_{x_i}$  being the grid size in the  $x_i$ -axis. In the following section it will be presented the numerical technique used in this work.

## 3. Numerical Method

The formulation of control-volume finite element method (CVFEM) involves five basic steps, Saabas and Baliga (1994a, 1994b): (1) discretization of the domain into finite elements; (2) another discretization of the domain into control volumes, which surround the nodes of the finite element mesh; (3) definition of element-based interpolation functions for variables and physical properties of the fluid; (4) derivation of algebraic equations by using

the sub-domain weighted residual method; and (5) assembling of the element equations creating the global matrix and the choice of a procedure to compute the system of algebraic equations.

**Figure 1** illustrates a nine-node finite element, control volumes and integration points (Ip) at faces of control volumes inside an element. The continue lines, the bold points, dashed lines and arrows represent the finite element contour, the nodal points, the control volumes contours and fluxes direction respectively. It can be noticed that each nodal point is inside a control volume like in finite volume method (FVM) and the integrations are done in counter-clockwise rotation.

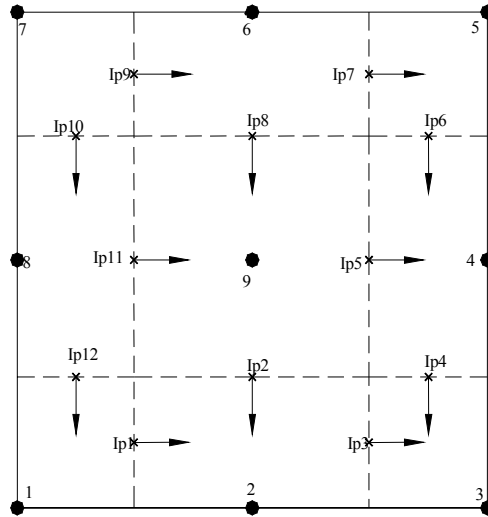
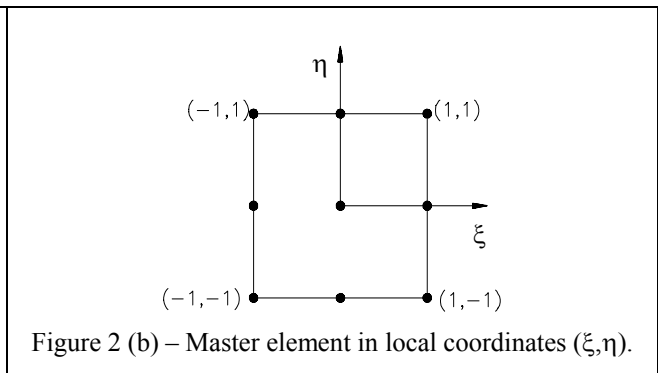
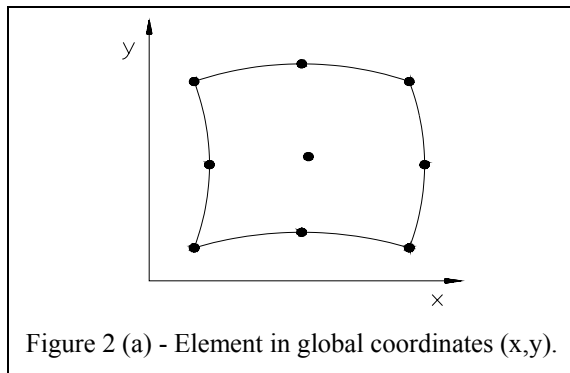


Figure 1 – Finite element divided in control volumes and integration points.

In order to simplify the integration process, each element in global coordinates (x,y), (Figure 2a), is mapped in a master element in local coordinates (ξ,η), (Figure 2b). So, all the nodal points of the integration domain are definite in the range [-1, 1], which is primordial to apply the integration Gaussian quadrature rule. However, it also must be done a further mapping to each control volume area and contour inside elements.



Thus, the integrals in areas defined by  $\int_{\xi_1}^{\xi_2} \int_{\eta_1}^{\eta_2} f(\xi, \eta) d\xi d\eta$  can be calculated by  $\int_{-1}^1 \int_{-1}^1 g(r, s) dr ds$ , which facilitates the integration process. Integrals in contours are of the type  $\int_{\xi_1}^{\xi_2} f(\xi) d\xi$  or  $\int_{\eta_1}^{\eta_2} f(\eta) d\eta$ , may be calculated as the definite integral in the range [-1, 1],  $\int_{-1}^1 g(s) ds$ .

In this way, the coefficient matrices are computed element by element in local coordinates and a global system of equations is assembled like in the classical finite element method. In this work have been used nine Gauss points for integration in areas to exactly integrate the mass matrix and three Gauss points for integration along contours. After the assembling, the algebraic global system of equations in the matrix notation is as follow:

$$K(U)U = F \tag{4}$$



where it can be seen that the global matrix  $K$  depends on the global vector of unknown parameters ( $U$ ), this occurs due to the non-linearity of the inertia terms. Iterative successive substitutions were employed to solve Eq. (4) until the convergence is attained.

The global system of equations is solved by the frontal method developed by Taylor and Hughes (1981). In this method the global system never is totally assembled in the computer memory. So, personal computers may be used to solve large problems (with more nodal points or degree of freedom).

Following, is presented the integration process in each control volume. Integrating Eqs. (1) and (2), ones obtain for each element:

$$\int_{\Omega} \frac{\partial U_i}{\partial t} d\Omega + \oint_{\Gamma} \left( U_j U_i - \left( \frac{1}{Re} + v_t \right) \frac{\partial U_i}{\partial X_j} \right) n_j d\Gamma + \int_{\Omega} \frac{\partial P_t}{\partial X_i} d\Omega = \oint_{\Gamma} v_t \frac{\partial U_j}{\partial X_i} n_j d\Gamma + \int_{\Omega} F_i d\Omega \quad (5)$$

$$\int_{\Omega} \frac{\partial U_i}{\partial X_i} d\Omega = 0 \quad (6)$$

where  $\Omega$  and  $\Gamma$  are symbols for area and contour of a control volume around a node inside elements, respectively. The arrows in Figure 1 indicate the outward normal vector to the contour of a control volume,  $n_j$ , in which there are convective and diffusive fluxes. This normal vector for integration in the counter-clockwise direction has been defined as:  $\vec{n} dS = n_1 dS \vec{i} + n_2 dS \vec{j} = dy \vec{i} - dx \vec{j}$ .

In order to transform the integrals of Eqs. (5) and (6) into algebraic equations, the variables in those integrals must be interpolated by appropriate functions. In this study the variables and coordinates are interpolated by Lagrangian interpolation functions. The variables inside each element are interpolated as:

$$U^e(X(\xi, \eta), Y(\xi, \eta), t) = \sum_{\alpha=1}^{nnep} N_{\alpha}(\xi, \eta) U_{\alpha}(t) \quad (7)$$

$$V^e(X(\xi, \eta), Y(\xi, \eta), t) = \sum_{\alpha=1}^{nnep} N_{\alpha}(\xi, \eta) V_{\alpha}(t) \quad (8)$$

$$P^e(X(\xi, \eta), Y(\xi, \eta), t) = \sum_{\lambda=1}^{nnel} N_{\lambda}(\xi, \eta) P_{\lambda}(t) \quad (9)$$

$$X^e(\xi, \eta) = \sum_{\alpha=1}^{nnep} N_{\alpha}(\xi, \eta) X_{\alpha} \quad (10)$$

$$Y^e(\xi, \eta) = \sum_{\alpha=1}^{nnep} N_{\alpha}(\xi, \eta) Y_{\alpha} \quad (11)$$

where, according to Dhatt and Touzot (1984),

- $\alpha$  : interpolation nodes for the velocities;
- $\lambda$  : interpolation nodes for the pressure;
- $N_{\alpha}$  and  $N_{\lambda}$  : interpolation functions;
- nnep and nnel: node numbers for quadratic (parabolic) and linear elements, respectively;
- $P_{\lambda}$  : nodal pressure;
- $U_{\alpha}$  and  $V_{\alpha}$  : nodal values of velocity components;
- $X_{\alpha}$  and  $Y_{\alpha}$  : coordinates at the node  $\alpha$  of the elements.

After the replacement of Eqs. (7)-(9) into Eqs. (5) and (6) ones obtain the set of algebraic equations:

$$M_{\alpha\beta} \dot{U}_{i\beta} + C_{\alpha\beta} (U_{i\beta}) U_{i\beta} - S_{i\alpha\beta} U_{i\beta} + H_{i\alpha\beta} P_{\beta} = F_{i\alpha} \quad (12)$$

$$D_{i\alpha\beta}U_{i\beta} = 0 \tag{13}$$

where  $M_{\alpha\beta}$ ,  $C_{\alpha\beta}$ ,  $S_{i\alpha\beta}$ ,  $H_{i\alpha\beta}$ ,  $D_{i\alpha\beta}$  and  $F_{i\beta}$  are mass coefficients, convection, diffusive, pressure term, continuity matrices and the source term vector, respectively, for each element. The detailed definition for these matrices can be founded in Campos-Silva (1998).

After the discretization of the transient term  $\dot{U}_{i\beta}$  of Eqs. (12) and (13) are obtained the following algebraic equations:

$$\frac{M_{\alpha\beta}}{\Delta t}U_{i\beta}^{n+1} + \theta(C_{\alpha\beta}^{n+1} - S_{i\alpha\beta})U_{i\beta}^{n+1} + \theta H_{i\alpha\beta}P_{\beta}^{n+1} = \frac{M_{\alpha\beta}}{\Delta t}U_{i\beta}^n - (1-\theta)(C_{\alpha\beta}^n - S_{i\alpha\beta})U_{i\beta}^n - (1-\theta)H_{i\alpha\beta}P_{\beta}^n \tag{14}$$

$$D_{i\alpha\beta}U_{i\beta}^{n+1} = 0 \tag{15}$$

where the time discretization parameter ( $\theta$ ) is in the range [0, 1]. In the present work was employed the fully implicit scheme,  $\theta = 1$ . In this case is not necessary the usage of any initial condition for the pressure field.

#### 4. Results

Following are presented the simulations done for a classical benchmark problem: a flow over the backward-facing step. Despite the simplicity geometrical it presents a substantial complexity in its flows due to the appearing of a recirculation zone, which posses two opposite vortex.

The main goal of this study is to verify the capability of a turbulence model implemented in an in-house code to predict the dependent features of the backward-facing step problem (Figure 3) as: reattachment length and corner eddy recirculation. These results are function of independent parameters as: initial boundary layer state, the initial boundary layer thickness, free stream turbulence, pressure gradient and aspect ratio.

The geometrical configuration and boundary conditions, for this challenge case, are shown in Figure 3. The inflow is placed  $-1.33h$  upstream of the step (with  $h$ =step height). The outflow is located about  $28h$  downstream of the step. Two cases were performed in this study with different velocity profiles at the inflow ( $U_0$ ). In the first and second case were imposed parabolic and uniform velocity profile conditions at the inflow, respectively. A set of null pressure was forced at the exit boundary and a set of null velocities was imposed at the remaining walls. In both cases the square of the Smagorinsky constant was set to  $c_s^2 = 0.0324$ .

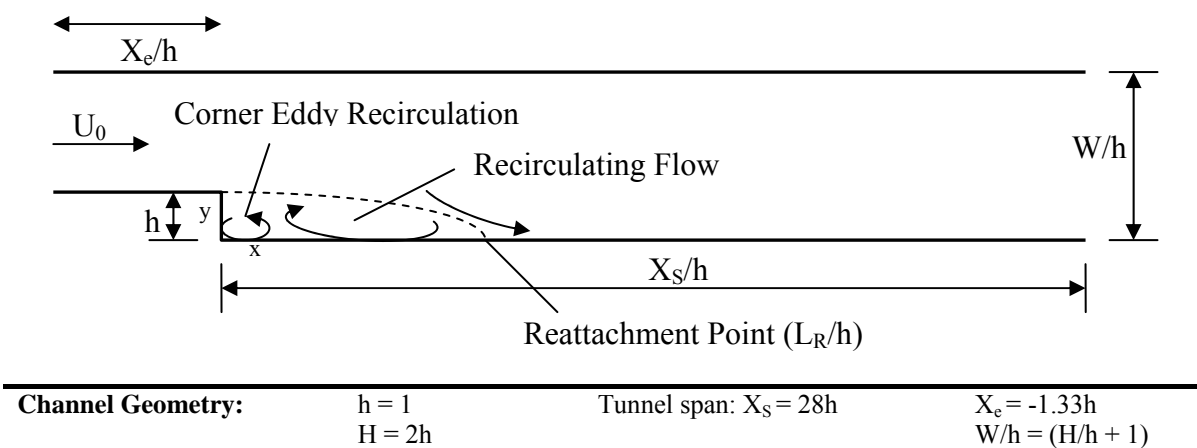


Figure 3– Computational domain and boundary conditions of the flow in a sudden expansion of a channel.

Figure 4 presents a slice of the computational domain discretized by a non-uniform quadratic nine-node finite elements; this grid was used for both the cases. This mesh amount 2,550 elements (or 10,413 grid points) and 23,483 degrees of freedom, which correspond to 13 by 51 and 121 by 81 nodal points at the entrance and in the tunnel span of the channel, respectively.



Figure 4 – Slice of the computational domain of the backward facing step discretized in 13 by 51 (entrance) and 121 by 81 (tunnel span) grid points.

In Table 1 are showed the boundary conditions for the backward facing step with parabolic velocity profiles in the entrance. The coordinate R has been started in the middle point of the inlet. The calculations have been performed for Reynolds numbers of 73 and 229.

Table 1 – Boundary conditions for the backward facing step with parabolic velocity profile in the entrance.

<b>Inlet Conditions:</b>	$U_o = 1.33(1 - R^2)$ $-1 \leq R \leq 1$	$Re_h=73$ $Re_h=229$
<b>Outlet Conditions:</b>	$p=0$	
<b>Channel Walls:</b>	$u=v=0$	

Figures 5-6 show velocity profiles for Reynolds number equal to 73 along of the channel. The results were compared to Winterscheidt and Surana (1994) and Campos-Silva (1998). In this second work wasn't used none turbulence model, however it can be noted that for this Reynolds number there isn't a substantial influence of the turbulence model, so the results present good agreement.

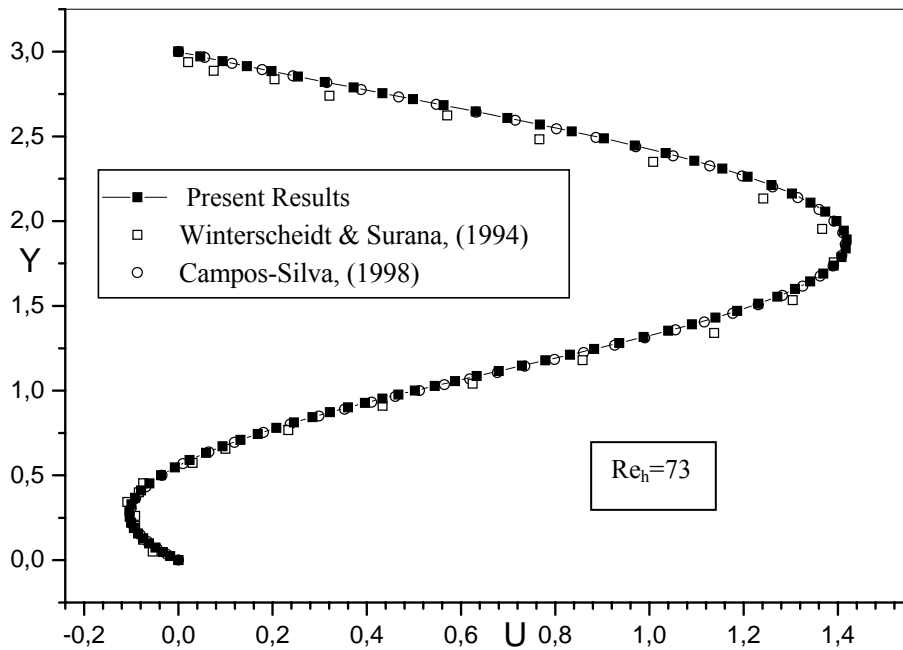


Figure 5 – U-velocity profiles at the downstream of the backward facing step (X/h=2).

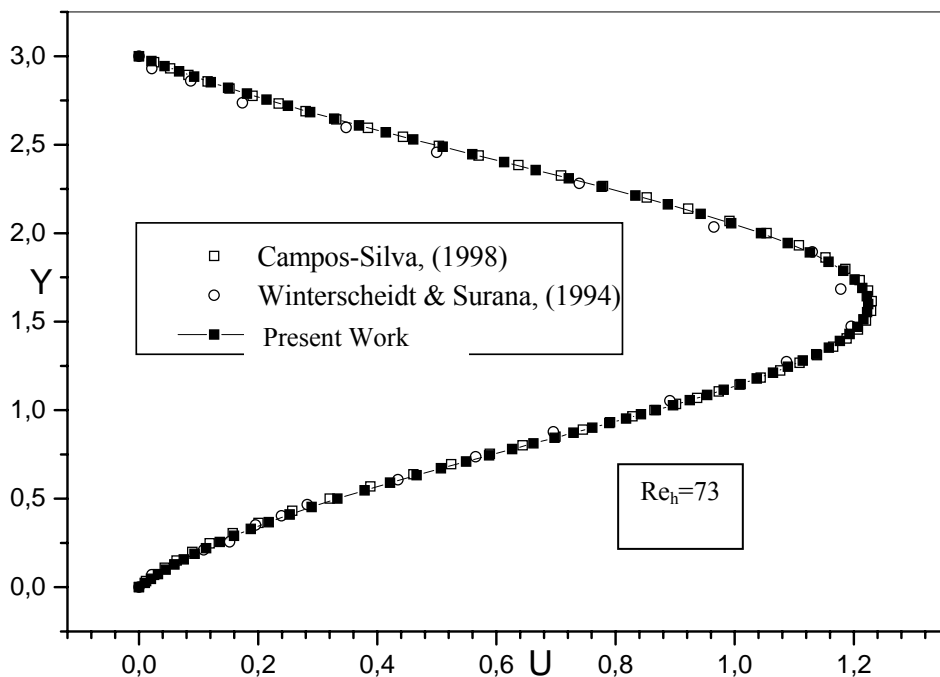


Figure 6 - U-velocity profiles at the downstream backward facing step ( $X/h=6$ ).

Velocity profiles for Reynolds number equal to 229 in two points in downstream of the step ( $X/h=2$  and  $X/h=6$ ) were compared to Winterscheidt and Surana (1994); Figure 7 enlightens this comparison, which present good agreement.

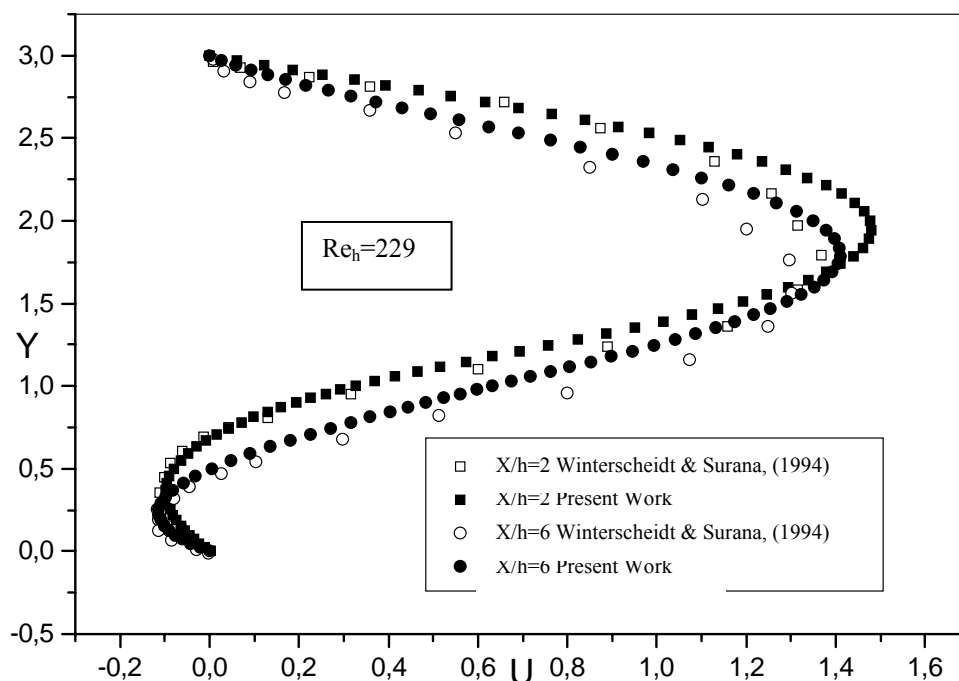


Figure 7 - U-velocity profiles at the downstream of the step.

**Table 2** shows the reattachment point for the Reynolds number 73 and 229. The results were compared to numerical (B and C columns) and experimental (A column) values. The experimental value was extract to

Winterscheidt & Surana (1994). The results of this work (D column) presented a small deviation (E column), below 14%.

Table 2 – Reattachment point ( $L_R/h$ ) for the Reynolds numbers 73 and 229.

	A	B	C	D	E
$Re_h$	Experimental	Campos-Silva (1998)	Winterscheidt & Surana (1994)	Present Work	$ (C-D)*100/C $
73	4.0	5.0	5.3	5.9	11.3%
229		9.7	9.7	11.0	13.4%

Following are presented results which posses uniform velocity profile condition in the inflow of the step flow. Table 3 presents the boundary conditions and Reynolds number imposed in this case.

Table 3 - Boundary conditions for the backward facing step with uniform velocity profile in the entrance.

<b>Inlet Conditions:</b>	$U_0 = 1$	$10,000 \leq Re_h \leq 71,000$
<b>Outlet Conditions:</b>	$p=0$	
<b>Channel Walls:</b>	$u=v=0$	

The results presented are of calculations performed for Reynolds numbers of 10,000, 14,000, 45,000 and 71,000. Initially, it was simulated the case for  $Re_h = 2,000$  with intention to use their velocity profiles as initial condition for  $Re_h=10,000$ , which would be used as initial condition for  $Re_h=14,000$  and so on, this procedure assisted in the convergence process.

The numerical values ( $L_R/h=7.3$ ) obtained in the present study for all Reynolds numbers simulated shows an excellent agreement to experimental results ( $L_R/h=7 \pm 1$ ) by Kim et al (1978). Thought the results can be noted that the flow apparently becomes independent of the Reynolds number when the boundary layer is fully turbulent, this also was observed by Eaton and Johnston (1981).

The streamlines are exhibited in the Figures 8-11 for Reynolds numbers of 10,000, 14,000, 45,000 and 71,000, respectively. The streamlines become possible the visualization of the separation line of the flow over the backward-facing step, which is straight and fixed at the downstream of the step. There is only one separated flow zone with two opposing eddies.

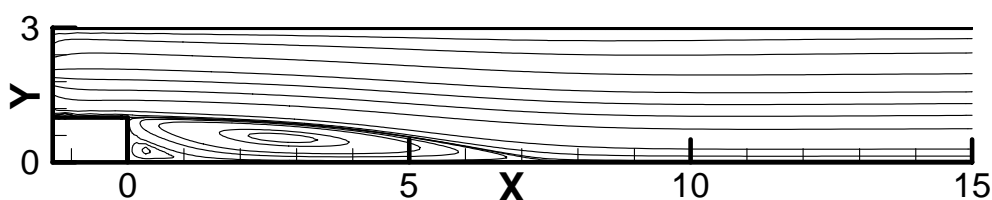


Figure 8 - Streamlines in a sudden expansion of a channel for  $Re_h=10,000$ .

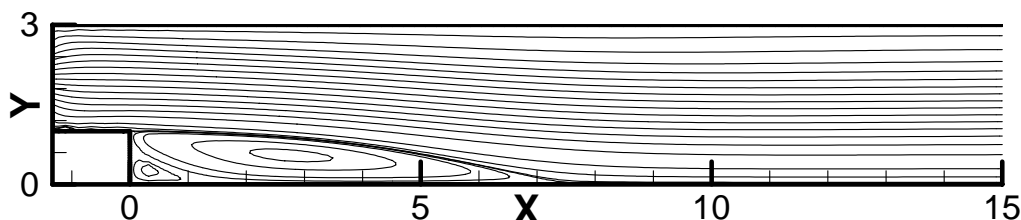


Figure 9 - Streamlines in a sudden expansion of a channel for  $Re_h=14,000$ .

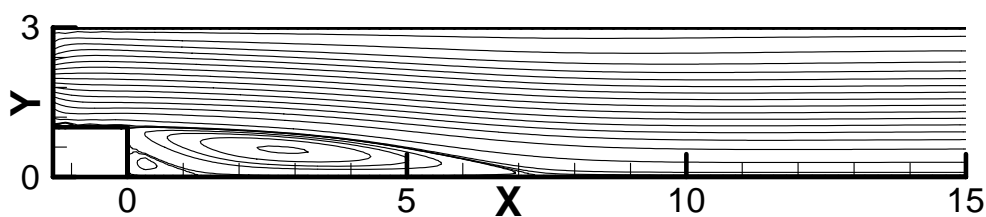


Figure 10 - Streamlines in a sudden expansion of a channel for  $Re_h=45,000$ .

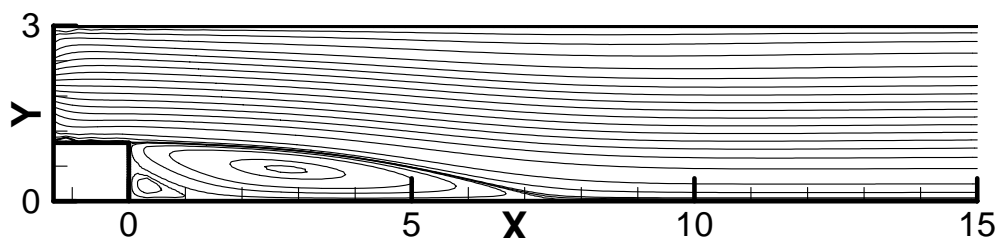


Figure 11 - Streamlines in a sudden expansion of a channel for  $Re_h=71,000$ .

## 5. Conclusions

In the present work were presented large-eddy simulations by a control volume–finite element or sub-domain finite element method using nine-node finite elements with quadratic and linear interpolation functions for velocity and pressure fields, respectively. The benchmark problem backward-facing step flows have been solved on Pentium 4 personal computer, 2 GHz and 2 Gb of RAM. Despite, no upwind technique has been applied and the fully implicit first order method has been used for time discretization the results present excellent agreement to results available from the literature.

In the flow over a backward facing step two principal parameters are evaluated: reattachment length values (a comparison can be view in Table 2) and the appearing of the corner eddy recirculation, which can be noted in Figures 8 -11. Both the results present good agreement to the literature.

## 6. Acknowledgement

The authors would like to acknowledge the FAPESP (Process 03/11737-2) for the computational resources and the CNPq for the scholarship for the graduate student R. C de Lima.

## 7. References

- Baliga, B. R. and Patankar, S. V., 1980, “A new finite-element formulation for convection-diffusion problems”, *Numerical Heat Transfer*, vol. 3., pp. 393-409.
- Campos Silva, J. B., 1998, “Numerical simulation of fluid flow by the finite element method based on control volumes”, Doctorate Thesis (in Portuguese), State University of Campinas, Campinas, São Paulo, Brazil.
- Deardorff, J. W., 1970, “A numerical study of tri-dimensional turbulent channel flow at large Reynolds number”, *Journal Fluids Mechanics*, 41, 342-379 pp.
- Dhatt, G. and Touzot, G., 1984, “The Finite Element Method Displayed”, John Wiley and Sons Ltda, New York, 526 p.
- Eaton, J.K. and Johnston, J.P., 1981, “A review of research on subsonic turbulent flow reattachment”, *AIAA Journal*, Vol. 19, No. 9, 1093-1100 pp.
- Hamba, F., 2003, “A Hybrid RANS/LES Simulation of Turbulent Channel Flow”. *Theoretical and Computational Fluid Dynamics*, vol. 16, pp. 387–403.
- Kim, J., Kline, J., Johnston, J.P., 1978, “Investigation of Separation and Reattachment of a Turbulent Shear Layer: Flow Over a Backward-Facing Step”, Report MD-37, Thermosciences Division, Department of Mechanical Engineering, Stanford University, Stanford.
- Lima, R. C., Mansur, S. S. and Campos Silva, J. B., 2004, “Large Eddy Simulation of Turbulent Incompressible Fluid Flows by a Nine-Nodes Control Volume-Finite Element Method”, *Proc. X Brazilian Congress of Thermal Sciences and Engineering - ENCIT'04*, Rio de Janeiro/RJ, 29/11/04 a 03/12/2004, paper code 0636.

- Lima, R. C., 2005, "Large-eddy simulation incompressible fluid flows with heat and mass transfer by a sub-domain finite element method" (In Portuguese)., Master Dissertation, São Paulo State University, (UNESP-Ilha Solteira), Ilha Solteira, S.P., Brazil, 161 p.
- Löhner, R., 2001, "Applied computational fluid dynamics techniques – An introduction based on finite element methods". John Wiley and Sons Ltd, New York, 376 p.
- Raw, M. J. and Schneider, G. E., 1986, "A skewed, positive influence coefficient up-winding procedure for control-volume-based finite-element convection-diffusion computation", Numerical Heat Transfer, vol. 9, pp. 1-26.
- Saabas, H. J., 1991, "A control volume finite element method for three-dimensional, incompressible, viscous fluid flow", Ph.D. Thesis, Dept. of Mech. Eng., McGill University, Montreal, Que, Canada.
- Saabas, H. J. and Baliga, B. R., 1994a, "Co-located equal-order control-volume finite-element method for multidimensional, incompressible, fluid flow - Part I: Formulation", Numerical Heat Transfer, Part B, vol. 26, pp.381-407.
- Saabas, H. J. and Baliga, B. R., 1994b, "Co-located equal-order control-volume finite-element method for multidimensional, incompressible, fluid flow - Part II: Verification", Numerical Heat Transfer, Part B, vol. 26, pp.409-424.
- Taylor, C. and Hughes, T. G., 1981, "Finite Element Programming of the Navier-Stokes Equations", Pineridge Press Ltd, Swansea, U.K.
- Winterscheidt, D. and Surana, K.S., 1994, "p-Version Least Squares Finite Element Formulation for Two-Dimensional, Incompressible Fluid Flow", Int. J. Numer. Methods Fluids, vol. 18, p.43-69.

## **8. Copyright Notice**

The author is the only responsible for the printed material included in his paper.

## COMPARATION BETWEEN SOLVERS USED IN ADDITIVE CORRECTION MULTIGRID METHOD WITH UNSTRUCTURED GRIDS

**Susie C. Keller**

susie@sinmec.ufsc.br

**Jonas Cordazzo**

jonas@sinmec.ufsc.br

**Clóvis R. Maliska**

maliska@sinmec.ufsc.br

**Lynceo F. Braghirolli**

lynceo@sinmec.ufsc.br

SINMEC – Laboratory of Numerical Simulation in Fluid Dynamics and Computational Heat Transfer, Dept. of Mechanical Engineering, Federal University of Santa Catarina  
Florianopolis, SC, Brazil

**Abstract.** *This paper shows a study about Additive Correction Multigrid method. The general motivation of multigrid methods is to solve efficiently a large algebraic equations set resulting from the discrete approximation of differential equations using several meshes in order to improve the convergence rate of iterative methods reducing all frequency modes of the error. The Additive Correction Multigrid (ACM) has several advantages over the traditional ones: the discretization is made only on the fine grid, reducing the complexity and the computational cost, as well as eliminating the possibility of inconsistencies among grids; it is a conservative method and it allows the use of an adaptive agglomeration cell scheme. However, the performance of any multigrid method is directly associated with the type of solver used in the several mesh levels. In this paper is studied the use of Gauss-Seidel or ILU to solve the linear system of different levels of ACM. The coarsest level equations, in turn, are solved with a direct-LU solver because the linear system is small becoming possible to obtain the exact solution. Another comparison between the adaptive agglomeration cell scheme and the geometric agglomeration cell scheme is made too. The Element-based Finite Volume Method (EbFVM) is the numerical method chosen to discretize the heat conduction equation in quadrilateral unstructured grids.*

**Keywords.** *Additive Correction Multigrid (ACM), ILU solver, Adaptive agglomeration scheme, Element-based Finite Volume Method (EbFVM), Unstructured grids.*

### 1. Introduction

In the mathematical and engineering literature several types of multigrid methods have been presented. The philosophy and theory behind such one was presented initially by Brandt (1977). The basic idea of any multigrid strategy is to use a sequence of grids with classical iterative solvers to accelerate the convergence. This is achieved because is possible to smooth the high-frequency modes of the error for each grid. Brandt decomposed the error of the approximate solution to a set of discrete equations in Fourier modes and analyzed how the solvers smooth the various modes. On the finest (original) grid, the high-frequency modes of the error are effectively reduced with classical iterative solvers, but the low-frequency modes are difficult to remove, since they would require coarser grids to attain this goal. In the coarse grids, the low-frequency modes of the error become high-frequency modes being easily removed by the linear solver. Practical applications and demonstrations of multigrid methods can be found in Briggs (2000) and Trottenberg et al. (2001).

The multigrid methods can be classified in two types: geometric and algebraic ones. The geometric multigrid methods present reasonable performance when the coefficients of the algebraic equations are nearly isotropic and homogeneous, but these methods perform poorly when the coefficients are strongly anisotropic. In the literature we can find several examples of geometric multigrid applications (Brandt, 1977; Vanka, 1986; Volker, 1996; Wesseling and Oosterlee, 2001; Rabi and Lemos, 2003; Mesquita and Lemos, 2005).

The algebraic multigrid methods, on the other hand, appeared to solve the problem related with anisotropic coefficients. Hutchison and Raithby (1986), Elias (1993), Venkatakrishnan and Mavriplis (1996), Francescatto and Dervieux (1998), Stüben (1999, 2001) among others have shown that the choice of block shape can have a drastic effect



on the convergence. Thus, the way as the cells are agglomerated in the fine grid to form the next coarse grid demands special consideration, and has been received lately in this kind of methods.

This article deals with one type of the algebraic multigrid scheme, namely the Additive Correction Multigrid method (Hutchinson and Raithby, 1986), simply called here as ACM. In this method, the coarse grid-equations are generated by the summation of the finer grid equations. Once the solutions on the coarser grid are obtained, they are simply added to the solution on the finer grid. Thus, interpolation or extrapolation operators, commonly used by multigrid classical methods, are not required.

The effects of the use of adaptive agglomeration in problems with anisotropic coefficients and a comparison of the performance of multigrid cycles V, W and F performance applied to unstructured meshes are presented in Keller et al (2004). These authors pointed out the flexibility of the method.

One relevant aspect to be analyzed in this paper is the base solver used to solve the several grid levels. An adequate solver will improve the convergence rate. The most common solvers used in multigrid methods are Gauss-Seidel and Incomplete Lower Upper factorization - ILU. In this work they are tested in order to point out their characteristics and to show the situations where each one is more suitable. For solving the coarsest mesh equations, the LU-direct solver is used because the linear system is small.

As ACM provides the use of adaptive agglomeration scheme besides the geometric agglomeration scheme, one study of these different schemes is also performed in this work. The former scheme consists in adding fine-grid cells with small transport timescales in order to reduce the variation in timescales for propagation of information on a coarse-grid cell, while the last scheme consists in joining the cells in order to form regular blocks.

In all cases analyzed here, the ACM method will be applied to solve the heat conduction problems whose differential equations are discretized by the Element based Finite Volume Method (EbFVM). This numerical method, which is suitable to deal with unstructured grids, will be shortly described here too.

## 2. The additive correction multigrid method

The Additive Correction Multigrid method (ACM) distinguishes from conventional multigrid methods for generating the coarse grid equations without the use of fixed stencils. In ACM, the coarse grid equations are formed by adding the fine-grid equations, resulting in the determination of constant corrections in coarse grid cells, which are summed to the fine grid solution in order to accelerate the convergence.

To exemplify the ACM, an unstructured mesh is depicted in Fig. (1). Even though the blocks may be formed in any convenient way, in this example the volumes are joined in blocks with five cells on average. We can see that the new coarse-grid blocks are formed by joining fine-grid control volumes in different directions, resulting in an agglomeration with highly unstructured forms.

Initially, we can considerate the system of equations to be solved written in the common form as

$$A_p^i \phi_i = \sum_{nb} A_{nb}^i \phi_{nb} + b_i, \tag{1}$$

where  $A_p^i$  is the central coefficient of the control volume  $i$  considered in the discrete equation for  $\phi$ ,  $A_{nb}^i$  are the coefficients connecting the control volume  $i$  to the neighbors control volumes,  $b_i$  is the source coefficient and  $\phi$  is the solution. The coefficients of Eq. (1) can be obtained by applying Finite Volume Method (FVM) or Element based Finite Volume Method (EbFVM) to either structured or unstructured grids (Maliska, 2004).

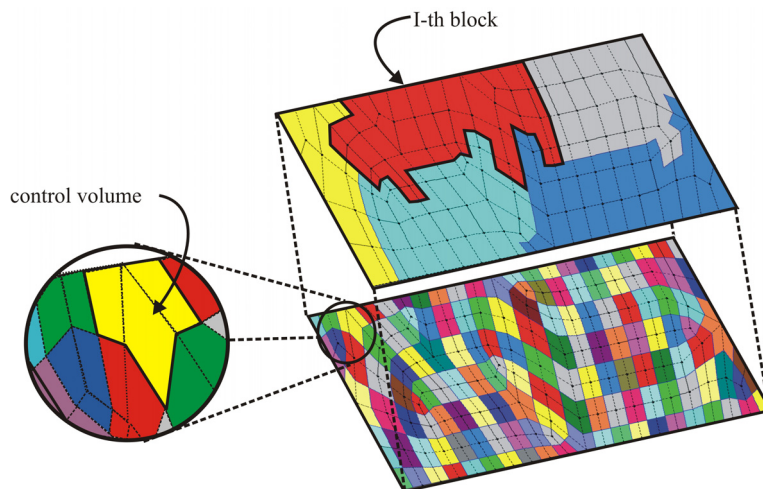


Figure 1. Example of a block agglomeration scheme in unstructured grids.

The core of ACM is the definition of a correction equation that has the role of adding coarse-grid corrections ( $\phi^*$ ) to the best estimate of  $\phi$  on the fine grid, as

$$\tilde{\phi}_i = \phi_i + \phi_{I,i}^*, \quad (2)$$

where  $\tilde{\phi}_i$  is the improved solution in each cell and  $\phi_{I,i}^*$  is the correction related to all volumes that lie within the  $I$  block.

Requiring that the residual be zero for the corrected solution, one obtains the linear system for  $\phi_{I,i}^*$  (Hutchinson and Raithby, 1986)

$$A_p^* \phi_p^* = \sum_{nb} A_{NB}^* \phi_{NB}^* + b_p^*, \quad (3)$$

where

$$b_p^* = \sum_{i,I} r_i, \quad (4)$$

$$A_p^* = \sum_{i,I} A_p^i - \sum A_{nb}^i, \quad (5)$$

$$A_{NB}^* = \sum_{i,I} A_{nb}^i. \quad (6)$$

In these equations  $\sum A_{nb}^i$  denotes the connection among the control volumes inside of one same block and  $\sum_{i,I} A_{nb}^i$  represents the connection among volumes of neighbor blocks.

Equation (3) must be solved in order to obtain the correction  $\phi_{I,i}^*$ . This correction is added to each  $\phi$  value of control volumes that lie within the  $I$ -block, Eq. (2). Thus, the improved estimate  $\tilde{\phi}_i$  is obtained, as already stated. Additional details of the ACM method can be found elsewhere (Hutchinson and Raithby, 1986; Maliska, 2004).

As can be seen, using the ACM method, there are no further decisions to be made related to the boundary conditions, transfer of residuals or interpolation of variables.

In the next section the different agglomeration schemes used in this paper are presented.

### 3. Agglomeration scheme

The manner of joining cells in multigrid methods to obtain the coarse grids can affect their efficiency. Moreover, considering the use of unstructured grids, it is expected that this is not a simple task. Following two forms of performing the agglomeration is presented.

#### 3.1 Adaptive agglomeration scheme

The nature of the coefficients in the fine mesh equations can strongly affect the convergence behavior of an iterative solver. One can show that there is a natural timescale for a perturbation to travel from a node to the neighbor node (Elias, 1993). This timescale is inversely proportional to the coefficient that connects two nodes. The iterative solver has a time step that approaches a converged solution by propagating disturbances through the solution domain. This time step is defined as the smallest timescale to propagate the information between nodes. The performance of iterative solver degrades when there is a high difference between the solver time step and the timescales to propagate the information between nodes. In order to accelerate the convergence of the discrete linear equations set solution, we can generate a coarse grid equations set by adding cells to reduce the differences between the largest and smallest transport time scales (Elias, 1993). Therefore, an adaptive agglomeration method based on fine mesh coefficients should be used.

The principle of an adaptive agglomeration scheme is to add fine grid cells with small timescale of the transport process (large coefficients), obtaining cells with small variation in the timescale of transport information process.

The adaptive agglomeration scheme begins with a single fine grid cell, and through a set of rules it is determined which of the neighbors should be included in the coarse grid cell. The new members of the coarse cell are examined to determine which of their neighbors should be included, and this is done either until the coarse cell reach the desired size

or there are no more neighbors to be added. Then, another coarser grid can be built agglomerating the new cells in the same way, and so on.

Following the nomenclature of a typical family tree as proposed by Elias et al (1997), we call the fine grid cell, whose neighbors are being examined, as the father, and its neighbors included in the same coarse cell as its children. The current father's father is known as the grandfather. This nomenclature is very convenient because the growth of a coarse cell occurs in a tree-like fashion.

There are two main rules used to decide which fine grid cells are added to the new coarse grid cells considered (Elias, 1993). The first agglomeration rule states that the coefficient that connects the father with the possible son must be of the same order or larger than half of the coefficient that connects the father with the grandfather. Representing the father by the index  $f$ , the possible son by  $s$  and the grandfather by  $g$ , we have that  $s$  is a possible son of  $f$  if:

$$\max(a_{fs}, a_{sf}) \geq \max(a_{gf}, a_{fg}) / 2. \quad (7)$$

The second rule states that a cell is excluded if the interface timescale is very large. The concept of what would be "large" is defined relative to all other timescales which affect the cell in question. A possible son can be agglomerated with the father if the coefficient that connects the father with the son will be of the same order or larger than half of the coefficient that connects the possible son with its neighbors. Thus, taking the coefficients again and representing the possible son's neighbors by the index  $n$ ,  $s$  is agglomerated with  $f$  if:

$$\max(a_{fs}, a_{sf}) \geq \max(a_{sn}, a_{ns}) / 2. \quad (8)$$

In order to improve the efficiency of this agglomeration scheme some details must be also considered. For instance, to simplify the choice of the first fathers for coarse grid cells and ensure that all cells are agglomerated in an only sweep of the fine grid, the first fathers are selected in the same order of sweep of the iterative solver. Moreover, as the initial father does not have any grandfather and any preferential timescale, the rule one can be ignored.

It is possible to find in the end of this process remaining cells in the fine grid without any neighbor that can be used as a son. To eliminate wasted computational effort with those single cells, any father cells which do not have children are forced to join the coarse grid cell of their most strongly connected neighbor.

### 3.2 Geometric agglomeration scheme

The geometric agglomeration scheme consists in adding fine grid cells in order to obtain, as much as possible, regular coarse blocks with good aspect ratio. Unlike the previous scheme, the cells coefficients are not important in this task. A cell is added in a block if it is a neighbor of other cell or cells that lie in this block.

## 4. The linear solvers used in ACM

The kind of solver used in each grid level is other important decision. A good relaxation scheme is one that is well able to reduce error components in the large coefficient direction of a wavelength extending as far as possible. Three different solvers will be considered here: Lower Upper (LU) factorization to the coarsest grid, and Gauss-Seidel and Incomplete Lower Upper factorization – ILU to the other grids.

### 4.1 Lower Upper Factorization (LU)

The LU factorization is a direct method, where the  $U$  matrix is obtained with the same algorithm used in Gauss Elimination and the  $L$  matrix is constructing with the multiplicative factors used in the building of the  $U$  matrix.

Considering the linear system

$$[A][\phi] = [L][U][\phi] = [B] \quad (9)$$

where  $A$  is the coefficient matrix,  $B$  is the independent terms vector, the  $\phi$  is unknowns vector,  $L$  is the lower triangular matrix and  $U$  is the upper triangular matrix.

The system of Eq. (9) can be solved finding  $D$  in the following linear system:

$$[L][D] = [B] \quad (10)$$

and, after, determining the unknown  $\phi$  with the Eq. (11)

$$[U][\phi] = [D]. \quad (11)$$

As the LU factorization deals with the full matrix, it is useful only to solve small linear systems. For this reason this solver is used only in the coarsest grid of ACM method.

The iterative linear solvers used in other grid levels of ACM are presented in the next section.

#### 4.2 Gauss-Seidel

This method belongs to the point-to-point iterative methods because it solves the linear system in serial manner, iteratively, using the values of variables at the previous iterative level and the values already calculated. The point-to-point iterative methods have convergence extremely slow when a large equation system must to be solved. However, they have been applied in multigrid methods due to the fact that there is no need of obtaining the complete convergence in each grid level (only a correction factor is calculated).

The iterative process consists in the calculation of  $\phi$ , Eq. (1) using an initial estimative. After, the convergence must be checked and this procedure must be repeated case the convergence has not been reached. Other possibility is the procedure of Gauss-Seidel to be repeated a certain number of times, This number is defined by the user and it occurs when Gauss-Seidel is used as the solver base of the ACM method.

#### 4.3 Incomplete Lower Upper Factorization (ILU)

This method is usually used as preconditioner for non-stationary methods like GMRES, conjugated-gradient and others. However, due to its strong implicit characteristics, the Incomplete Lower Upper Factorization – ILU can be used also as a base solver in multigrid methods. This method deals with a storage system where only no-null values are saved, and consequently, only these values can be modified by LU factorization, justifying the name: “Incomplete”. Some details about how it works are given in sequence.

Considering the linear system:

$$[A][\phi] = [B], \quad (12)$$

where  $A$  is the coefficient matrix,  $B$  is the independent terms vector and  $\phi$  is unknowns vector.

The philosophy of ILU methods is given by

$$[L][U] = [A] + [A']. \quad (13)$$

where  $L$  is the lower triangular matrix,  $U$  is the upper triangular matrix and  $A'$  is a matrix created to accelerate the convergence and that do not influence in the final solution.

When the LU factorization originates the  $A$  matrix, the method is a direct method, but how is created the matrix  $A'$ , the method is iterative and has the following form

$$[A + A'][\phi]^{k+1} = [A + A'][\phi]^k - \{[A][\phi]^k - [B]\} \quad (14)$$

where  $k$  is the iterative level. When the solution converged is obtained, the second term of the right hand of Eq. (14) becomes zero, resulting in  $\phi^{k+1} = \phi^k$ .

Defining a correction to  $\phi$  given by

$$[\delta]^{k+1} = [\phi]^{k+1} - [\phi]^k \quad (15)$$

and the solution residue as

$$[R]^k = [A][\phi]^k - [B] \quad (16)$$

we can substitute Eq. (15) and Eq. (16) in Eq. (14) resulting in

$$[A + A'][\delta]^{k+1} = -[R]^k, \quad (17)$$

and substituting the Eq. (13) in Eq. (17), we obtain

$$[L][U][\delta]^{k+1} = -[R]^k. \tag{18}$$

which is a equation to determine the correction. The solution is obtained through two process of successive replacements, since the matrices  $L$  and  $U$  are lower triangular and upper triangular matrices, respectively. Thus, defining a new vector  $V$ , it can be determined by

$$[L][V] = -[R]^k \tag{19}$$

and knowing  $V$ , we can obtain  $\delta$  by

$$[U][\delta]^{k+1} = -[V]^k. \tag{20}$$

Thus,  $\phi$  is updated in the following form

$$[\phi]^{k+1} = [\phi]^k + [\delta]^{k+1}. \tag{21}$$

The residue is calculated again by Eq. (16) and the iterative process follows until the residue will be less than tolerance specified, or for a definite number of times.

In the next section, it is shown the method used to discretize the differential equation in all problems analyzed in this paper.

### 5. Element-based Finite Volume Method (EbFVM)

In a finite volume methodology the computational domain is covered by non-overlapping control-volumes where the balances are made. In the EbFVM, the element concept and the geometric representation inherited from the finite element method are utilized, and its definition precedes the creation of the control volume. Figure (2) shows an example of an EbFVM grid, emphasizing the differences between grid elements (entity defined by the nodes) and control-volumes (built around the nodes). Besides, it is shown the integration points where the conservation equations are integrated.

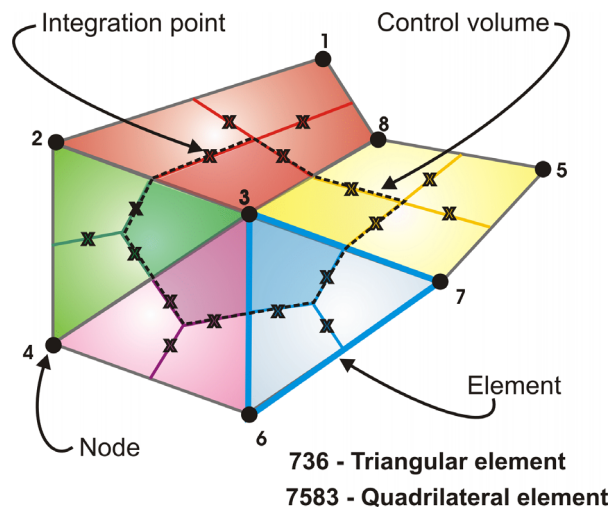


Figure 2. Control volume in a unstructured grid of triangular and quadrilateral elements.

The control volumes are created by the method of the medians, which consists of joining the center of the elements to their medians. The resulting control volume is formed by portions (sub-control volumes) of neighboring elements.

More details involving the deduction of EbFVM discretized equations can be found elsewhere (Raw, 1985; Maliska, 2004).

## 6. Results

Two example problems will be presented to demonstrate the efficiency of the ACM method using different base solvers and agglomeration schemes. Considering grids with 1024, 4096, 16384 and 65536 nodes for these two problems, each block is agglomerated with four cells on average in each level. Depending on the number of cells that the original grid has, the number of levels is defined in order to obtain always the coarsest grid with around 64 cells, which is solved directly with LU method. The maximum allowed RMS residue to the solution in the original grid was  $10^{-5}$ . The results are obtained with the use of the V, W and F cycles (Trottenberg et al., 2001) in all cases analyzed here.

### 6.1 Comparison between the results from ACM with different base solvers

**Test-Case 1.** The first problem considered is a diffusion problem with  $x$ -boundaries of the domain insulated and the  $y$ -boundaries partly insulated, as represented in Fig. (3). The EbFVM was used for discretizing this problem because it treats the grids in an unstructured fashion, even though it is used here grids geometrically structured with nodes or control-volumes equally in both  $x$  and  $y$  directions. Because of the problem geometry, the coefficients related to nodes in the interior of domain are greater in the  $y$  direction than in  $x$  direction.

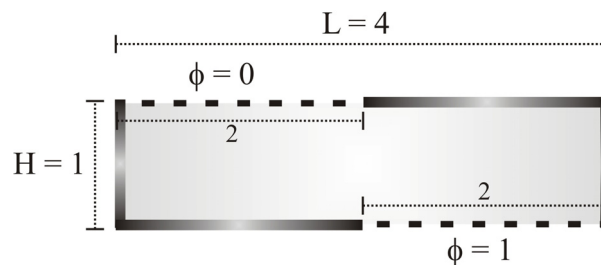
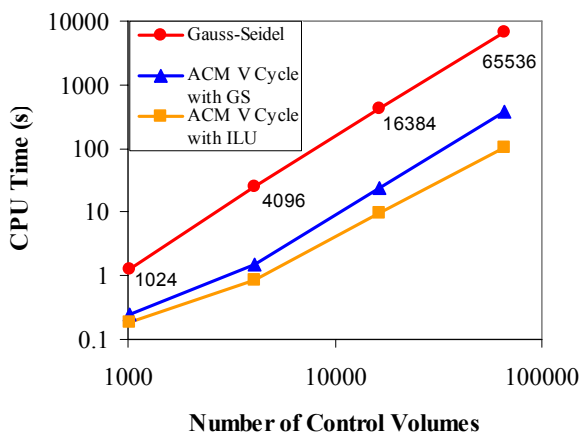
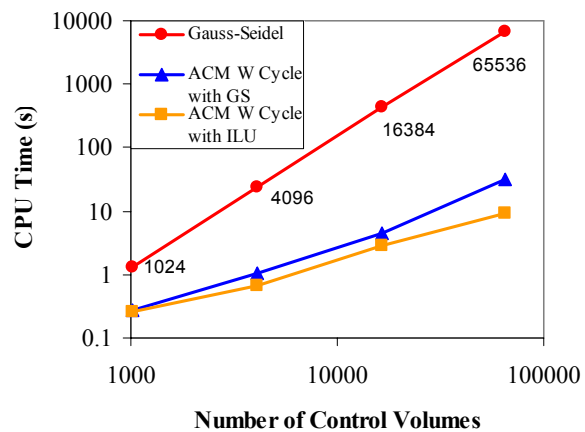


Figure 3. Geometry of the first example problem.

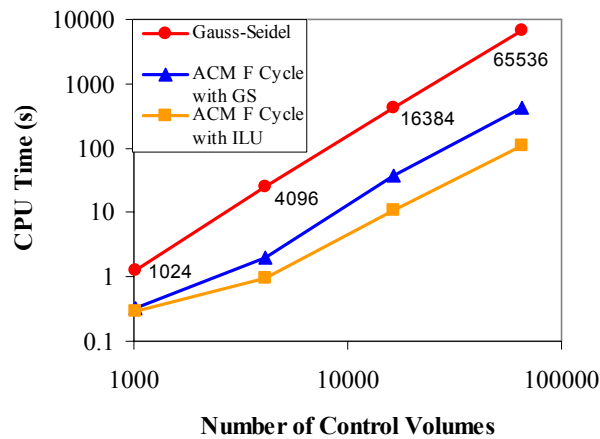
In the results presented in Fig. (4) is used the adaptive agglomeration scheme. Thus, the agglomeration is always made in the direction of the greater coefficients. We can see for the cycles V, W and F, that the ACM method with ILU solver presented the smallest computational cost.



(a)



(b)



(c)

Figure 4. Comparison between iterative solvers used in ACM method to the first conduction problem in (a) V cycle, (b) W cycle and (c) F cycle, in relation to the traditional Gauss-Seidel solver.

**Test-Case 2.** The second example is similar to the first one, except for the boundary conditions on  $x$  and  $y$  coordinates that are interchanged, as shown in Fig. (5). In this example, the information about the boundary conditions is transmitted mainly in the direction of smaller coefficients, so we expect a larger computational effort required for Gauss-Seidel to reach the convergence.

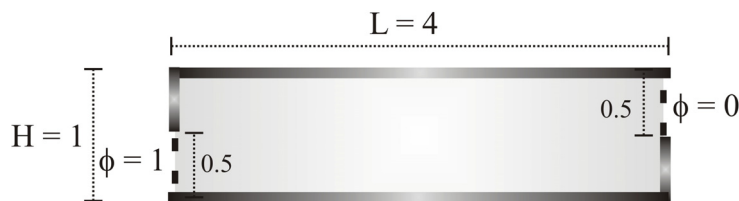
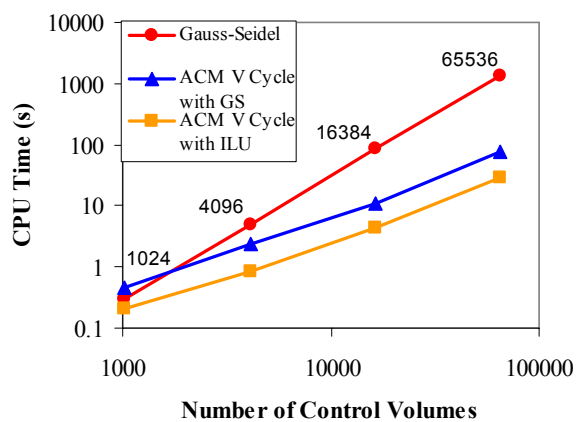
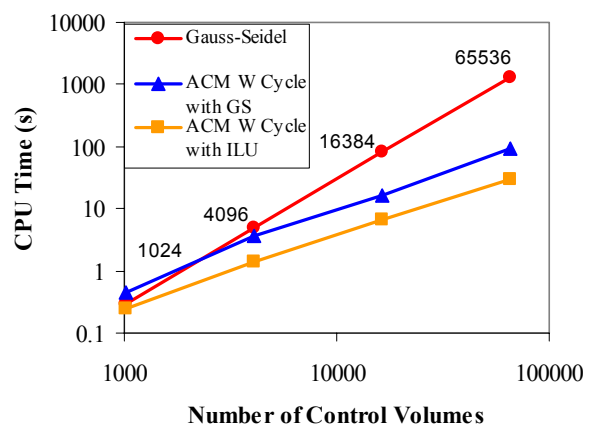


Figure 5. Geometry of the second example of diffusion problem.

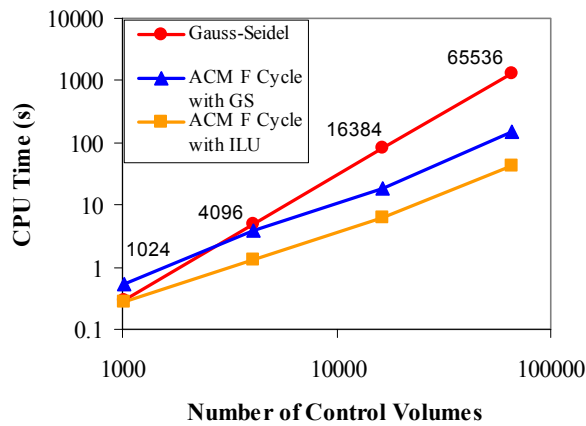
The results presented in Fig. (6) show again that the ILU solver used in ACM method has the smallest computational cost and thus it is better than ACM with Gauss-Seidel and the Gauss-Seidel alone, despite of the convergence behavior to be different of the first example because the boundary conditions are interchanged. Here, is used too the adaptive agglomeration scheme.



(a)



(b)



(c)

Figure 6. Comparison between iterative solvers used in ACM method to the second conduction problem in (a) V cycle, (b) W cycle and (c) F cycle, in relation to the traditional Gauss-Seidel solver.

**Discussion.** We can see too that the ACM W cycle showed the better performance in the problem 2 (Fig. (6.b)), while in problem 1 the three cycles presented similar performance (Fig. (4)).

In problem 1, Fig. (4), the traditional Gauss-Seidel solver had a good performance only in the mesh with 1024 cells. It occurs because the coefficients are large in the  $y$  direction and the effect of the Dirichlet boundary conditions is rapidly transmitted to interior of domain. Thus, the residue to be eliminated in the  $x$  direction is small and Gauss-Seidel is sufficient to solve this diffusion problem in relatively coarse meshes. However, with the increasing of the grid size, it is necessary the use of ACM method to reduce the computational cost.

In Fig. (6), we can see the high computational time gain of ACM over Gauss-Seidel solver in all grids considered. It occurs because the Dirichlet boundary conditions are prescribed in the weak coefficient direction (see Fig. (5)). Thus, Gauss-Seidel solver propagates the boundary condition information in this direction one node per iteration, while ACM propagates the information over the whole domain in only one iteration.

## 6.2 Comparison between the results from ACM with different agglomeration schemes

Now, we will compare two forms of performing the agglomeration: adaptive and geometric, both explained in section 3. From now on, the ACM method with ILU solver is the default, since this showed to be the best configuration to the ACM method in the cases analyzed. The two cases considered earlier and described in Figs. (3) and (5) will be considered again. As the only difference among them is the boundary conditions, their coarse grids, being calculated either by the adaptive or geometrically, must be the same and are presented in the sequence.

At first, in Fig (7) is shown the original grid with 1024 cells, where the control volumes are represented by the blue lines, the nodes by the circles and the elements by the hatched lines.

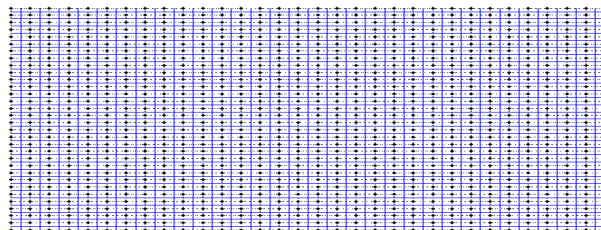


Figure 7. Original grid of two examples with 1024 cells.

Figure (8) shows two levels of geometric agglomeration with four cells, resulting in 256 new coarse cells, Fig. (8.a) and 64 new coarse cells, Fig. (8.b).

The two coarse grids, with 256 and 64 cells, determined following the rules of the adaptive agglomeration are shown in Fig. (9). Since this problem presents anisotropic coefficients due to the geometry with the greater coefficients in  $y$  direction, the adaptive agglomeration will occur in that direction, as shown in Figs. (9.a) and (9.b).



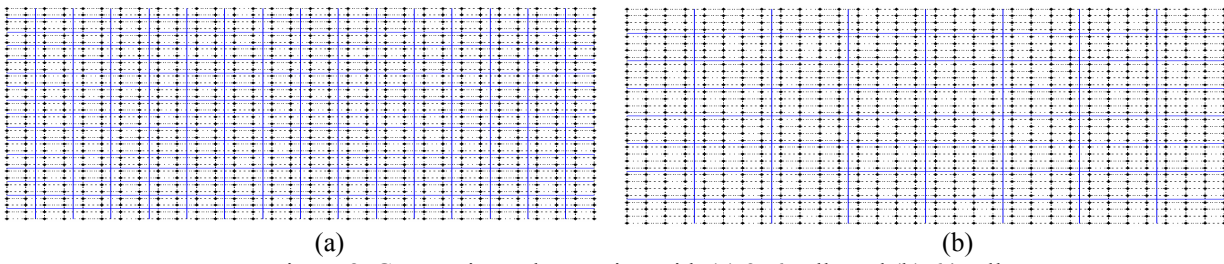


Figure 8. Geometric agglomeration with (a) 256 cells and (b) 64 cells.

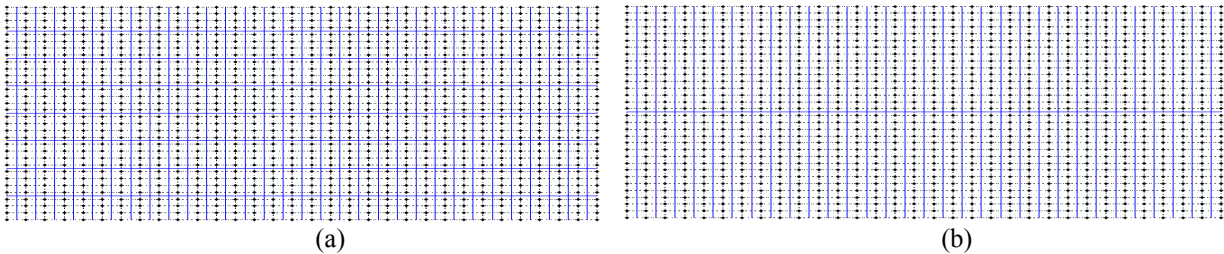


Figure 9. Adaptive agglomeration with (a) 256 cells and (b) 64 cells.

In the graphics presented in Figs. (10) and (11) are shown the total CPU time needed to reduce the RMS residual to a pre-determined value (in this case  $10^{-5}$ ) in different size grids obtained using the ACM with adaptive agglomeration, ACM with geometric agglomeration and an iterative solver, in this case Gauss-Seidel. Taking the example 1 for each cycle we obtain the results showed in Fig. (10).

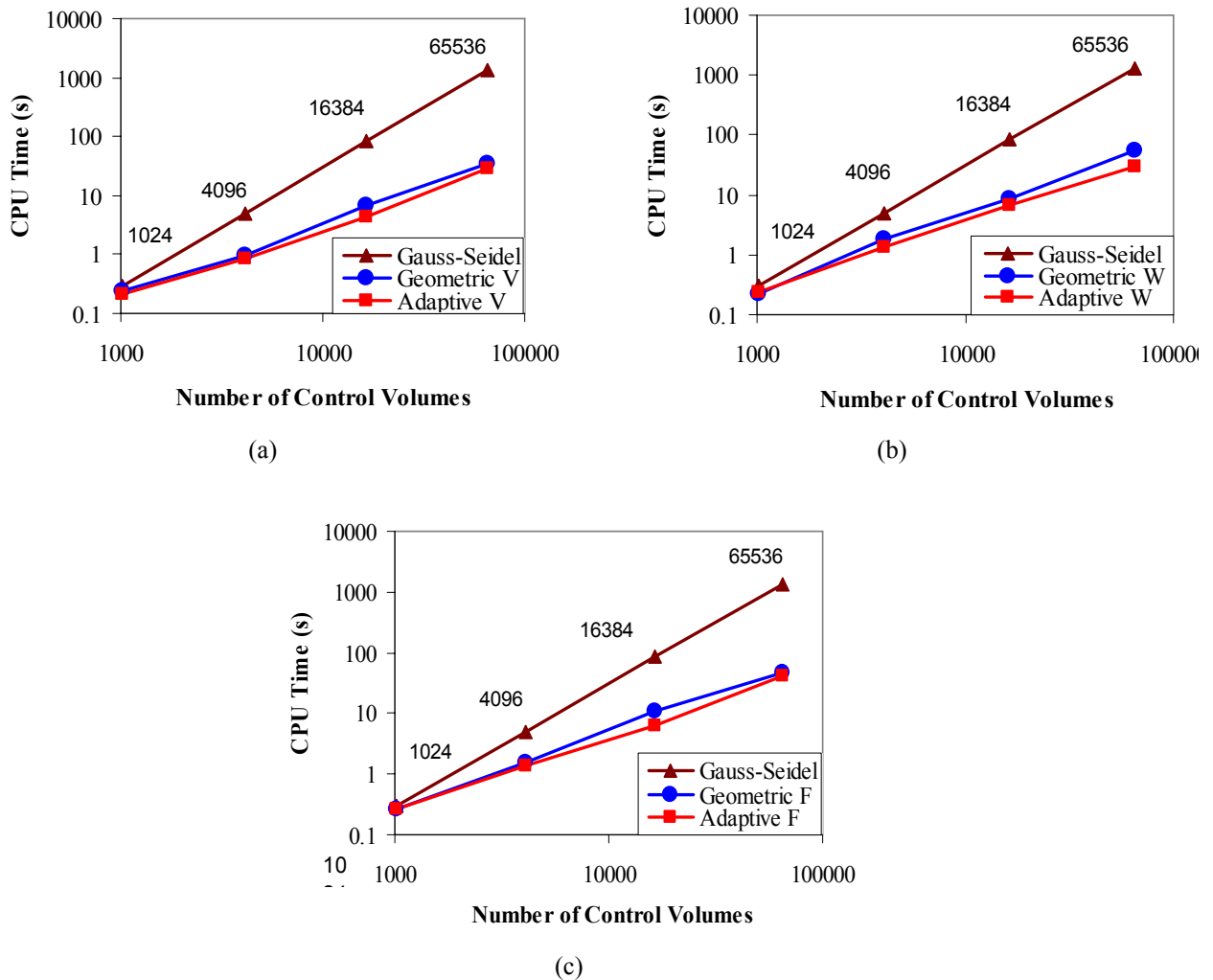


Figure 10. Comparison between agglomeration schemes used in ACM method to the first conduction problem in (a) V cycle, (b) W cycle and (c) F, in relation to the traditional Gauss-Seidel solver.

We can see that ACM with adaptive agglomeration scheme reach the converged solution more quickly than the ACM with geometric scheme and for this it is considered better. However the two agglomeration schemes had gotten good results (considering the CPU time), especially in comparison with Gauss-Seidel solver that it is a slow solver.

In Fig. (11) will be shown the results obtained to example 2, Fig. (5), comparing again ACM with adaptive agglomeration, ACM with geometric agglomeration and Gauss-Seidel solver.

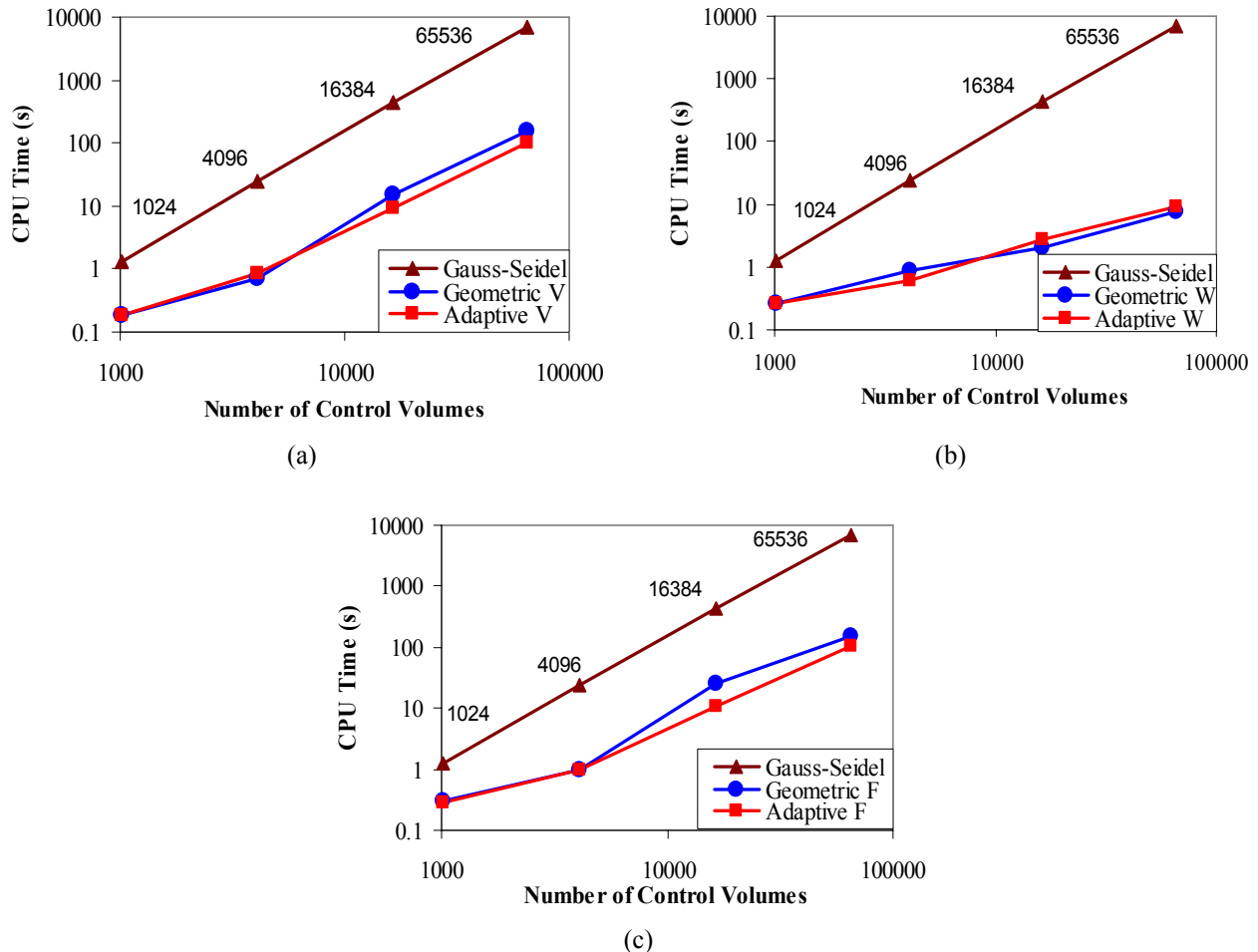


Figure 11. Comparison between agglomeration schemes used in ACM method to the second conduction problem in (a) V cycle, (b) W cycle and (c) F cycle, in relation to the traditional Gauss-Seidel solver.

In Fig. (11.b), where is shown the performance of the two agglomeration schemes in W cycle, it is not defined which scheme is the best. In Fig. (11.a) and (11.c) the adaptive agglomeration scheme had better performance that the geometric scheme in the more large grids.

## 7. Conclusions

This paper has outlined some of the ACM main features. It was shown that the choice of which iterative method will be used in the grid levels influences the performance of the ACM method. It was seen that the solution obtained with the multigrid is accelerated even more when the ILU factorization is used as a base solver.

Two agglomeration types were studied: adaptive and geometric. The adaptive scheme considers the nature of the coefficients, joining cells with small timescale of the transport process (large coefficients). The geometric scheme joins cells trying to obtain only a regular coarse blocks with good aspect ratio.

In the cases analyzed here, the two agglomeration schemes had similar performance. However the schemes must be exhaustingly tested in other kinds of anisotropic problems to obtain a more conclusive idea regarding their behavior.

Finally, we can say that this paper studied some important features of ACM method that must to be chosen when one decides to implement and use this kind of method. However more analyses in unstructured meshes must be done, mainly foreseeing its application to a coupled linear equation system.

## **8. References**

- Brandt, A., 1977, "Multi-Level Adaptive Solutions to Boundary-Value Problems", *Mathematics of Computation*, vol 31, pp. 333-390.
- Briggs, W. L., Henson, V. E., McCormick, S. F., 2000, "A Multigrid Tutorial", *Society for Industrial and Applied Mathematics*, EUA, 193p.
- Elias, S. R., 1993, "Enhancements to Additive Correction Multigrid", Master thesis, Waterloo, Ontario, Canada.
- Elias, S.R., Stubbley, G.D., Raithby, G.D., 1997, "An Adaptive Agglomeration Scheme for Additive Correction Multigrid", *International Journal of Numerical Methods in Engineering*, Vol. 4 (5), pp. 887-903.
- Francescato, J., Dervieux, A., A Semi-Coarsening Strategy for Unstructured Multigrid based on Agglomeration, *International Journal for Numerical Methods in Fluids*, Vol. 26, pp. 927-957, 1998.
- Hutchinson, B. R., Raithby, G. D., 1986, "A Multigrid Method Based on the Additive Correction Strategy", *Numerical Heat Transfer*, vol. 9, pp. 511-537.
- Keller, S. C., Cordazzo, J., Hinckel, P. H., Maliska, C. R., 2004, "The Additive Correction Multigrid Method for Unstructured Grids", *CILAMCE – XXV Iberian Latin-American Congress on Computational Methods in Engineering*, 10-12 de Novembro, Recife, Pernambuco, Brazil.
- Keller, S. C., Cordazzo, J., Hinckel, P. H., Maliska, C. R., 2004, "Additive Correction Multigrid Method Applied to Diffusion Problems with Unstructured Grids", *ENCIT – 10<sup>th</sup> Brazilian Congress of Thermal Engineering and Sciences*, 29 de Novembro - 3 de Dezembro, Rio de Janeiro, Brazil.
- Maliska, C. R., 2004, "Transferência de Calor e Mecânica dos Fluidos Computacional", Ed. LTC, Rio de Janeiro, Brazil, 453 p.
- Mesquita M. S., Lemos, M. J. S., 2005, "Effect of Medium Properties on Convergence Rates of Multigrid Solutions of Laminar Flows in Permeable Structures", *International Journal of Dynamics of Fluids* Vol. 1, No.1, pp. 69-86.
- Raw, M. J., 1985, "A New Control-Volume-Based Finite Element Procedure for the Numerical Solution of the Fluid flow and Scalar Transport Equations", Doctor thesis, Waterloo, Ontario, Canada.
- Rabi, J. A., Lemos, M. J. S., 2003, "Multigrid correction-storage formulation applied to the numerical solution of incompressible laminar recirculating flows". *Applied Mathematical Modelling* 27, pp. 717-732.
- Stüben, A., 1999, "Algebraic Multigrid (AMG)", *An Introduction with Applications*, GMD-Report 70.
- Stüben, A., 2001, "A Review of Algebraic Multigrid", *Journal of Computational and Applied Mathematics*, Vol. 128, pp. 281 –309.
- Trottenberg, U., Oosterlee, C., Schüller, A., 2001. "Multigrid", Academic Press, 631p.
- Vanka, S. P., 1986, "Block-Implicit Multigrid Calculation of Two-Dimensional Recirculating Flows", *Computer Methods in Applied Mechanics and Engineering*, 59, pp. 29- 48.
- Venkatakrishnan, V., Mavriplis, D. J., 1996 "Implicit Method for the Computation of Unsteady Flows on Unstructured Grids", *Journal of Computational Physics*, Vol. 127, pp. 380-397.
- Volker, S., Burton, T., Vanke, S. P., 1996, "Finite Volume Multigrid Calculation of Natural-Convection Flows on Unstructured Grids", *Numerical Heat Transfer, Part B*, 30:1-22.
- Wesseling, P., Oosterlee, C. W., 2001, "Geometric multigrid with applications to computational fluid dynamics", *Journal of Computational and Applied Mathematics*, 128, pp. 311-334.

## NUMERICAL SIMULATION OF THE DISPERSION OF A CONTAMINANT SPECIE OVER OBSTACLES USING A CVFEM – FORMULATION

**Clodoaldo Borges Chagas (\*)**

[clodoaldo@dem.feis.unesp.br](mailto:clodoaldo@dem.feis.unesp.br)

**Odacir Almeida Neves (\*\*)**

[odacir@fem.unicamp.br](mailto:odacir@fem.unicamp.br)

**Luiz Felipe Mendes de Moura (\*\*)**

[felipe@fem.unicamp.br](mailto:felipe@fem.unicamp.br)

**Rosiane Cristina de Lima**

Instituto Tecnológico de Aeronáutica, Departamento de Engenharia Aeronáutica e Mecânica, Área de Aerodinâmica, Propulsão e Energia – Praça Marechal Eduardo Gomes, 50, Vila das Acácias, CEP 12228-900, São José dos Campos, São Paulo, Brasil  
[rosiane@ita.br](mailto:rosiane@ita.br)

**João Batista Campos Silva (\*)**

[jbcampos@dem.feis.unesp.br](mailto:jbcampos@dem.feis.unesp.br)

(\*) Department of Mechanical Engineering, Faculty of Engineering of Ilha Solteira, Av. Brazil, 56 – P.O. Box 31 State University of São Paulo, 15385-000 – Ilha Solteira, São Paulo, Brazil

(\*\*) Department of Thermal and Fluids Engineering, Faculty of Mechanical Engineering, P.O. Box 6122 State University of Campinas, 13083-970 – Campinas, São Paulo, Brazil

*Abstract. The main purpose of this work is the numerical computation of dispersion of a contaminant specie over a region with obstacles, using a code based in Control Volume-Finite Element Method (CVFEM). Thus, the flow, temperature and concentration fields are resolved. The turbulence treatment have been done though large eddy simulation (LES) and the sub-grid scales stresses, heat and mass fluxes have been modeled through the Smagorinsky's eddy viscosity model. The domain is discretized using nine-node finite elements and the equations are discretized into control volumes around the nodes of the finite elements. Some benchmark problems are solved to validate the numerical code and the preliminary results are presented and compared to available results from the literature.*

*Keywords: contaminant dispersion, CVFEM, large eddy simulation, incompressible flow, Smagorinsky's model*

### 1. Introdução

Fluid flows are of interest in several engineering practical applications as: diffusers, airfoils with separation, buildings, combustors, turbines blades and many flow-relevant systems (Driver and Seegmiller, 1985).

Most of the fluid flows are turbulent; they occur in complex domains and are governed by a non-linear partial differential equations set of the convection-diffusive type. So, practical solutions can be obtained through numerical techniques such as: The Finite Difference Method (FDM), the Finite-Volume Method (FVM) and/or the Finite Element Method (FEM). The finite volume method is the most popular method used to calculate fluid flows. However, last decades, the finite element method has been improved; enabling its usage in Computational Fluid Dynamics (CFD) and it becomes a powerful tool in the fluid flow simulations in complex geometries.

In previous works, Campos Silva & Moura (1997, 2001), Campos Silva (1998) and Campos Silva *et al.* (1999) presented a development of a control volume finite element method using a quadratic, quadrilateral nine-noded element to simulate unsteady, incompressible and viscous fluid flows. In those works no turbulence model was considered, so the results were obtained for relatively low Reynolds numbers.

The main goal of this work is twofold. Primarily, it is desired to evaluate the large eddy simulation methodology (LES) implemented in the cited code previously. A validation of the code has been done by the simulations of classical benchmark flows: lid-driven-cavity and the backward facing-step flows, (Lima, 2005). After, it is desired to simulate the dispersion of a passive pollutant by a flow of the wind, utilizing a two-dimensional geometry with blocks similar to buildings in urban areas. Some authors as: Neofytou *et al* (2006), Wong (2002), Savii (1998) investigated this kind of problem named pollutant dispersion in urban canyon.

Turbulent flows are characterized by eddies with a wide range of length and time scales. The largest eddies are typically comparable in size to the characteristic length of the mean flow. The smallest scales are responsible for the dissipation of turbulence kinetic energy. It is possible, in theory, to directly resolve the whole spectrum of turbulent

scales either using an approach known as direct numerical simulation (DNS), or by the usage of turbulence models (algebraic, two-equations models, second order models) with the Reynolds Average Navier-Stokes equations (RANS), or filtering the Navier-Stokes equations by the large eddy simulation methodology. In LES, large eddies are resolved directly, while small eddies (or subgrid-scales stresses) are modeled. By the increase of the computational powerful, the use of LES to simulate fluid flows has gained many adepts. Walton *et al.* (2002) utilized the LES in a mean flow and turbulent problem in cubic street canyons, obtaining results with good agreement to experimental data.

In next sections, it will be presented the mathematical model, the numerical method and the preliminary results of the simulation of pollutant dispersion around solids like urban street canyons.

## 2. Mathematical model

In this section the set of equations that describe the dispersion of a passive scalar by the action of a non-isothermal flow is presented. Also, some aspects of the large-eddy simulation are presented. The filtered set of partial differential equations are:

### Continuity Equation

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0 \quad (1)$$

### Momentum Equation

$$\frac{\partial(\rho \bar{u}_i)}{\partial t^*} + \frac{\partial(\rho \bar{u}_j \bar{u}_i)}{\partial x_j} = -\frac{\partial \bar{p}_i}{\partial x_i} + \frac{\partial}{\partial x_j} \left( \mu \frac{\partial \bar{u}_i}{\partial x_j} \right) - \frac{\partial \tau_{ij}}{\partial x_j} + \bar{S}_i \quad (2)$$

### Equation of a Scalar Variable (Concentration, temperature, etc.)

$$\frac{\partial(\bar{\varphi})}{\partial t^*} + \frac{\partial(\bar{u}_j \bar{\varphi})}{\partial x_j} = \frac{\partial}{\partial x_j} \left( \Gamma_{\varphi}^* \frac{\partial \bar{\varphi}}{\partial x_j} \right) - \frac{\partial q_{j\varphi}}{\partial x_j} + \bar{S}_{\varphi} \quad (3)$$

where the properties and some variables are

- $\rho$  density;
- $\mu$  dynamic viscosity;
- $\bar{\varphi} = \bar{T}$  in temperature equation;
- $\bar{\varphi} = \bar{C}$  in concentration equation;
- $\Gamma_{\varphi}^* = k / \rho c_p$  thermal diffusivity in temperature equation;
- $\Gamma_{\varphi}^* = D$  mass diffusivity in concentration equation;
- $S_i$  and  $S_{\varphi}$  source terms accounting for the effects of buoyancy, heat generation (temperature equation) or chemical reaction (concentration equation);
- $*$  dimensional variables;
- (overbar) filtered or large scale variables.

In large-eddy simulation, the large-scale variables representing the velocity, pressure or other scalar fields are defined by a filter function in the form:

$$\bar{f}(\bar{x}) = \int_D f(\bar{x}') G(\bar{x}, \bar{x}', \Delta) d\bar{x}' \quad (4)$$

where  $G$  is a filter function, the most common filter functions are: Gaussian, Top-hat or Sharp Fourier cut-off filters, Chidambaram (1998); and  $\Delta$  is the filter width, generally,  $\Delta = (\Delta_x \Delta_y \Delta_z)^{1/3}$  with  $\Delta_{x_i}$  being the grid size in the  $x_i$  axis.

The sub-grid-scale (sgs) stress tensor  $\tau_{ij}$  appearing in Eq. (2) and the sub-grid flux  $q_{j\varphi}$  in Eq. (3) come from the filtering process, they are defined as:

$$\tau_{ij} = \rho(\overline{u_i u_j} - \bar{u}_i \bar{u}_j); \quad q_{j\varphi} = -\rho(\overline{u_j \varphi} - \bar{u}_j \bar{\varphi}) \quad (5)$$

which must be modeled. In this work it has been used the Smagorinsky model and the sgs stress tensor and fluxes are modeled in the form:

$$\tau_{ij} = \frac{2}{3} \rho k \delta_{ij} - 2\mu_t^* \bar{S}_{ij}; \quad q_{i\phi} = -\Gamma_{\phi t}^* \frac{\partial \bar{\phi}}{\partial X_i} \quad (6)$$

where the eddy viscosity,  $\mu_t$ , the turbulent kinetic energy,  $k$  and the deformation rate are defined by the following expressions:

$$\mu_t^* = \rho (C_s \Delta)^2 (2\bar{S}_{kl} \bar{S}_{kl})^{1/2}; \quad k = \frac{\tau_{ii}}{2}; \quad \bar{S}_{ij} = \frac{1}{2} \left( \frac{\partial \bar{u}_i}{\partial X_j} + \frac{\partial \bar{u}_j}{\partial X_i} \right) \quad (7)$$

The squared Smagorinsky's constant after some tests with lid-driven cavity flow was choose to be  $C_s^2 = 0,026$ , Lima (2005). The filter width was choose to be equal to the length of control volume face where the convective and diffusive fluxes are considered ( $\Delta = \int_{\xi} \sqrt{(\partial x / \partial \xi)^2 + (\partial y / \partial \xi)^2} d\xi$  or  $\Delta = \int_{\eta} \sqrt{(\partial x / \partial \eta)^2 + (\partial y / \partial \eta)^2} d\eta$ ). See Fig. 2 for reference.

After the substitution of Eq. (7) and Eq. (6) into Eq. (2) and Eq. (3) one obtains the following equations in dimensionless form:

$$\frac{\partial U_i}{\partial X_i} = 0 \quad (8)$$

$$\frac{\partial U_i}{\partial t} + \frac{\partial (U_j U_i)}{\partial X_j} = -\frac{\partial P_t}{\partial X_i} + \frac{\partial}{\partial X_j} \left( \left( \frac{1}{Re} + v_t \right) \frac{\partial U_i}{\partial X_j} \right) + \frac{\partial}{\partial X_j} \left( v_t \frac{\partial U_j}{\partial X_i} \right) + F_i \quad (9)$$

$$\frac{\partial \Phi}{\partial t} + \frac{\partial (U_j \Phi)}{\partial X_j} = \frac{\partial}{\partial X_j} \left( \left( \frac{\Gamma_{\Phi}}{Re} + \Gamma_{\Phi t} \right) \frac{\partial \Phi}{\partial X_j} \right) + S_{\Phi} \quad (10)$$

where the term of kinetic energy was included in pressure term resulting the turbulent pressure,  $P_t = \bar{P} + \frac{2}{3} \frac{k}{u_0^2}$ . The dimensionless variables are defined as:

$$X_i = \frac{x_i}{L}; \quad U_i = \frac{\bar{u}_i}{u_0}; \quad \bar{P} = \frac{\bar{p} - p_0}{\rho u_0^2}; \quad t = \frac{t^*}{L/u_0}; \quad v_t = \frac{v_t^*}{u_0 L} = \left( C_s \frac{\Delta}{L} \right)^2 (2\bar{S}_{kl} \bar{S}_{kl})^{1/2}; \quad Re = \frac{\rho u_0 L}{\mu}; \quad Pr = \frac{\mu c_p}{k}; \quad Sc = \frac{\nu}{D},$$

$$\Phi = \frac{(\bar{T} - T_0)}{\Delta T} \text{ or } \Phi = \frac{\bar{C}}{C_0}; \quad \Gamma_{\Phi} = \frac{1}{Pr}; \quad \Gamma_{\Phi t} = \frac{v_t}{Pr_t} \text{ or } \Gamma_{\Phi} = \frac{1}{Sc}; \quad \Gamma_{\Phi t} = \frac{v_t}{Sc_t} \quad (11)$$

where  $L$  is a characteristics length and variables with a subscript zero are reference variables.

### 3. Numerical method

The CVFEM was firstly presented by Baliga and Patankar (1980, 1983) and later by Raw and Schneider (1986). They used triangular and linear quadrilateral elements, respectively. Several authors have enhanced the CVFEM since that time till nowadays. Banaszek (1989) compared both the Galerkin and CVFEM methods in diffusion problems using six-noded and nine-noded elements. Campos Silva (1998) and Campos Silva & Moura (2001) presented the procedure of application of a nine-noded CVFEM.

The CVFEM formulation involves five basic steps: (1) discretization of the domain of interest into elements; (2) further discretization of the domain into control volumes that surround the nodes in the finite element mesh, as shown in Figure 1; (3) definition of element-based interpolation functions for variables and physical properties of the fluid; (4) derivation of algebraic equations by the usage of the sub-domain weighted residual method; and (5) assembling of the element equations forming the global matrix and choice of a procedure to solve the system of algebraic equations. In this method, each node of the finite element mesh is inside a control volume like in FVM. An element and its respective control volumes are showed in Figure 2, where are also showed the faces with convective and diffusive fluxes.

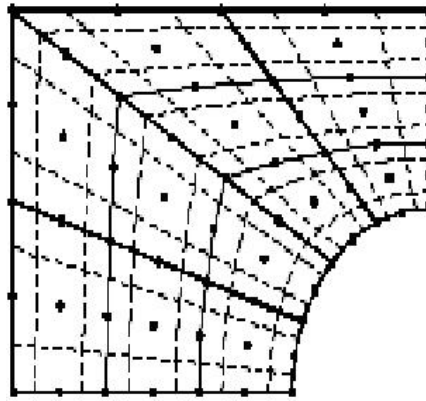


Figure 1 - Meshes of finite elements e control volumes.

Now, the integration of the Equations (8) - (10) inside each control volumes yields:

$$\int_V \frac{\partial U_i}{\partial X_i} dV = 0 \tag{12}$$

$$\int_V \frac{\partial U_i}{\partial t} dV + \oint_S \left( U_j U_i - v_e \frac{\partial U_i}{\partial X_j} \right) \cdot n_j dS + \int_V \frac{\partial P_t}{\partial X_i} dV = \oint_S v_t \frac{\partial U_j}{\partial X_i} n_j dS + \int_V F_i dV \tag{13}$$

$$\int_V \frac{\partial \Phi}{\partial t} dV + \oint_S \left( U_i \Phi - \Gamma_e \frac{\partial \Phi}{\partial X_i} \right) \cdot \bar{n}_i dS = \int_V S_\Phi dV \tag{14}$$

where  $S$  and  $V$  denoting the surface area and the volume of a control volume around a node in the element, respectively,  $n_j$  is the outward normal vector to the area of a control volume where there are convective and diffusive fluxes. The effective viscosity and diffusivity are:  $v_e = \left( \frac{1}{Re} + v_t \right)$ ;  $\Gamma_e = \frac{\Gamma_\Phi}{Re} + \Gamma_{\Phi t}$

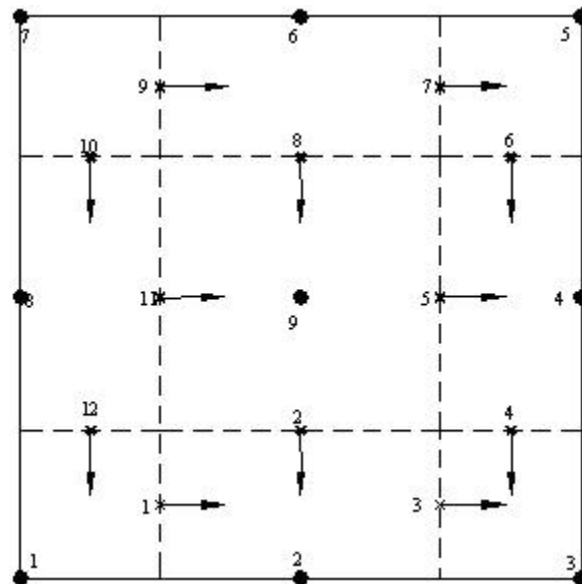


Figure 2 – Finite element divided in control volumes and integration points.

In order to transform the integrals of Eqs. (12), (13) and (14) to algebraic equations, the variables and the coordinates are interpolated like in classical Galerkin finite element method, as it's defined bellow:

$$U_i = \sum_{\alpha=1}^{n_{nep}} N_\alpha U_{i\alpha} \tag{15}$$

$$P = \sum_{\alpha=1}^{nnel} N_{\alpha} P_{\alpha} \tag{16}$$

$$\Phi_i = \sum_{\alpha=1}^{nnep} N_{\alpha} \Phi_{i\alpha} \tag{17}$$

$$X_i = \sum_{\alpha=1}^{nnep} N_{\alpha} X_{i\alpha} \tag{18}$$

where  $N_{\alpha}$  are interpolation functions of the reference element in local coordinates (Dhatt & Touzot, 1984);  $U_{i\alpha}$  are the velocity components;  $P_{\alpha}$  is the pressure;  $\Phi_i$  is any scalar and  $X_{i\alpha}$  are components of the coordinate system at nodes  $\alpha$  of a element; nnep and nnel are nodes numbers of quadratic (parabolic) and linear elements respectively. The velocity field and scalar variables are interpolated by quadratic functions. The pressure is interpolated by linear functions (Eq. 16).

The coefficient matrices are computed element by element in local coordinates as it's showed in Fig. 3a-b and a global system of equations is assembled like in the classical finite element method. The global system of equations is solved by the frontal method proposed by Taylor & Hughes (1981). An important characteristic of this method is that the system is assembled element by element; therefore the global system never is assembled totally in memory. So, personal computers may be used for solve the cases studied.

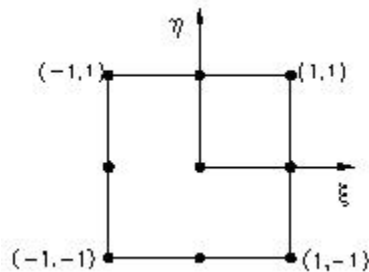


Figure 3.(a) - Element in local coordinates  $\xi$ - $\eta$

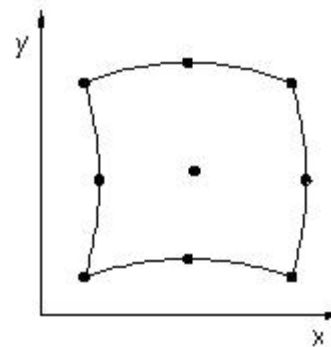


Figure 3.(b) - Element in global coordinates x-y

#### 4. Results

The computational program was validated by Lima (2005) for classical benchmark problems: lid-driven cavity and backward-facing step flows. Dispersion of a pollutant in a cavity was also simulated by Lima with the same boundary conditions of a free convection problem with good agreement of the results with results from the literature. In that work some simulations of dispersion in urban canyons was initiated. The simulations here have been done for a flow inside a street canyon based in Savii (1998). In this case the interest is in the horizontal dispersion of pollutant specie around the obstacles.

The discretized domain for this flow is shown in Figure 4 (top view). The original dimensions of the mesh are  $0 \leq X \leq 65$  and  $0 \leq Y \leq 46$ . The length of the buildings is 10.0; the width of upper and lower buildings is 6.0 and 2.0, respectively. The boundary conditions in the flow are  $U=1, V=0$  and  $T=0$  at the inflow section, upper and lower boundaries;  $T=1$ , in the buildings boundaries,  $P=0$  in the outlet section;  $C=1$  between the buildings, in the center horizontal line of the domain. In this work:  $Re=100, Sc=0.2, Ra=0, Pr=0.7$ .

A qualitative comparison of the simulated concentration of this work with results from Savii (1998) is shown in Figure 5. Savii considered only transient and diffusive terms in concentration equation, while in this work the convective terms were also included. Maybe, it was the cause of differences between the results, as shown in Figure 5. In the work of Savii it was not possible to get all information about the domain of calculation and numerical values of the velocity and concentration fields.



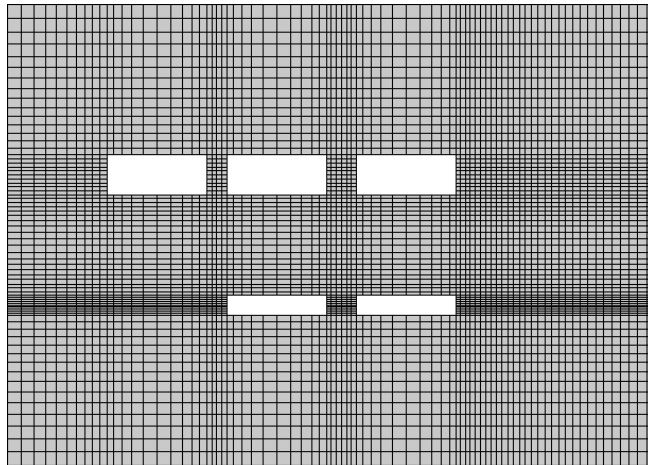


Figure 4. Discretized domain with five obstacles.

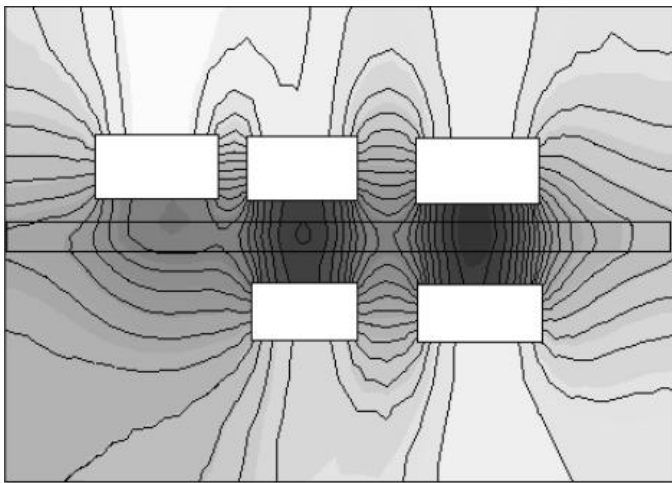


Figure 5a. Simulated concentration field, Savii (1998)

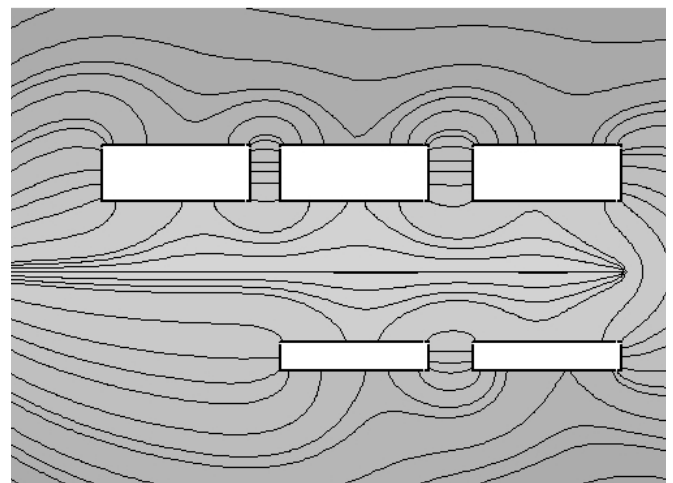


Figure 5b. Simulated concentration field, this work

The Figures 6-13 show the X-velocity profiles before the first building, between buildings and after last buildings at several time steps. Qualitatively the behavior of the velocity has been simulated; however, the size of the computational domain has to be increased. As can be seen in Figure 13 the presence of obstacles still has much influence on the outlet velocity and the length of computational domain after the obstacles seems to have been short. If the computational dimensions are appropriated, most probably the outlet velocity profile will be parabolic or more uniform. The size of the domain could be insufficient for imposed boundary conditions.

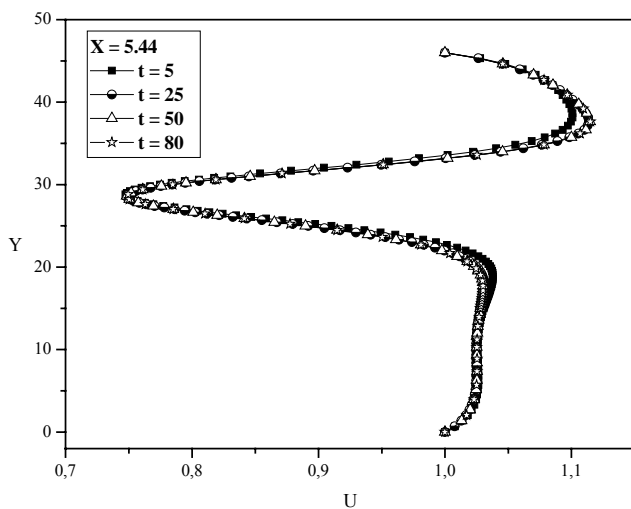


Figure 6. X-Velocity before the first building,  $X = 5.4$ .

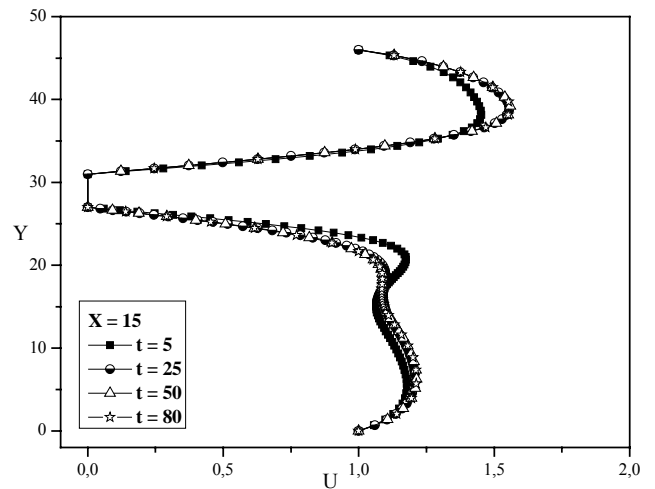


Figure 7. X-Velocity in the half of first building,  $X = 15$ .

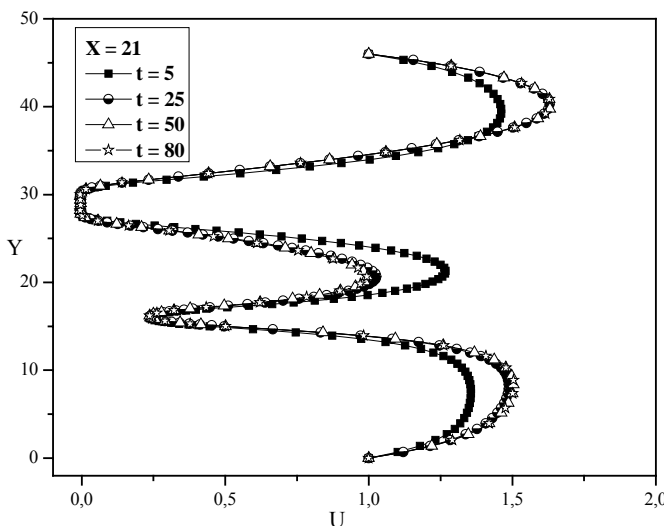


Figure 8. X-Velocity between the first and second building, X = 21.

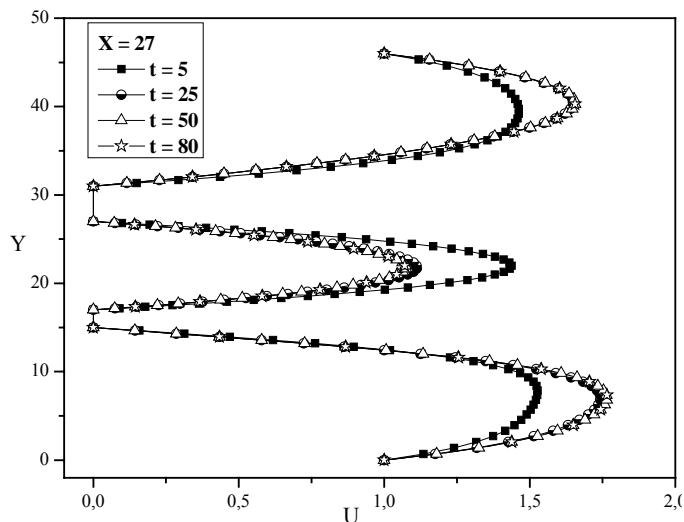


Figure 9. X-Velocity in the half of second building, X = 27.

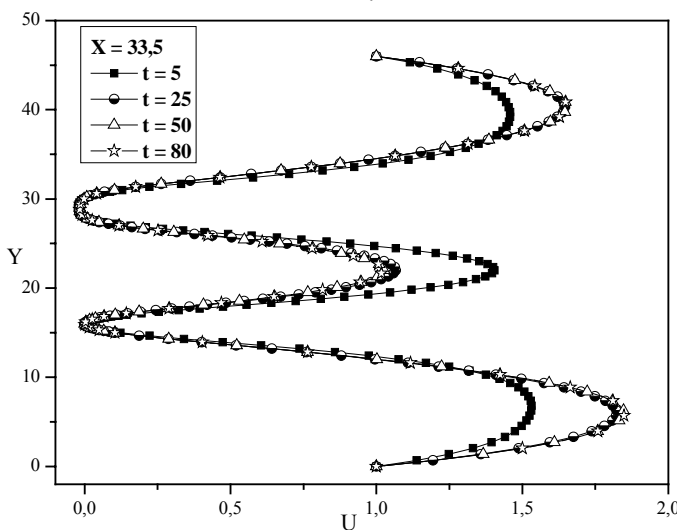


Figure 10. X-Velocity between the second and third building, X = 33.5.

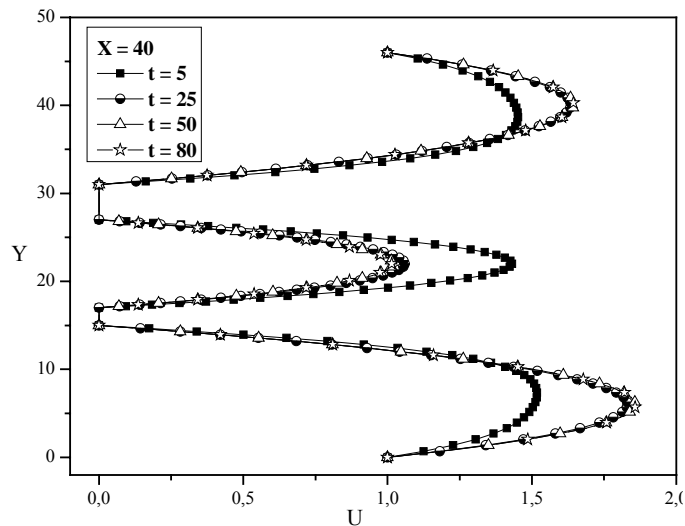


Figure 11. X-Velocity in the half of third building, X = 40.

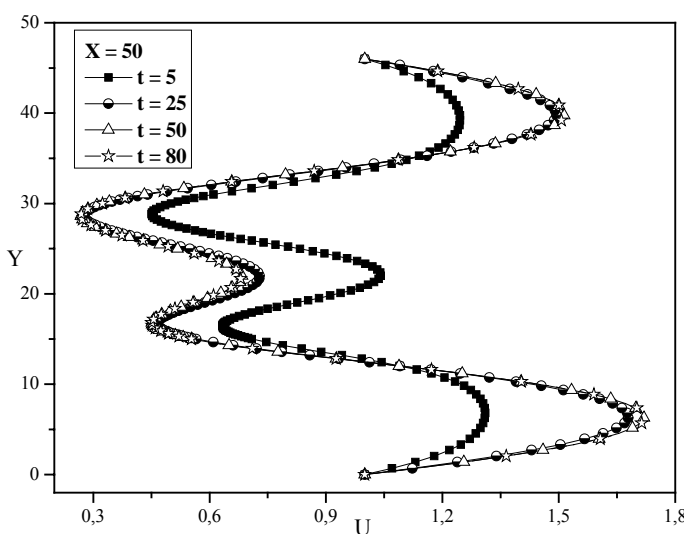


Figure 12. X-Velocity after the third building, X = 50.

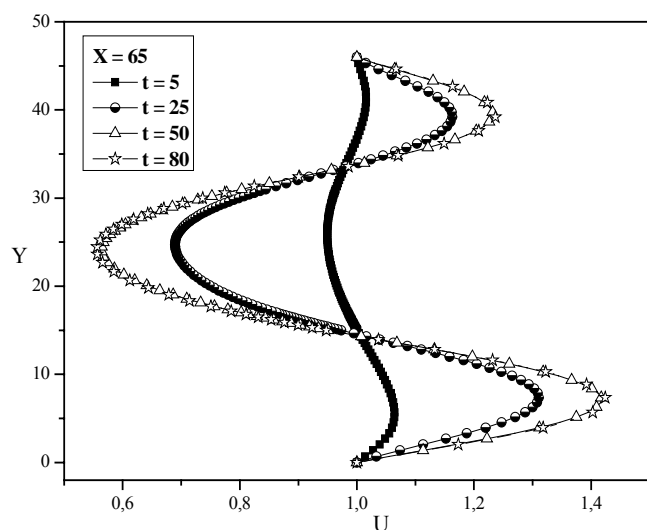


Figure 13. X-Velocity in the exit of flow, X = 65.

The evolution with time of the pollutant concentration in the domain is shown in Figures 14 to 21. Qualitatively, the results seem to be simulated with the expected behavior. It was not possible to do comparisons with experimental

and other numerical results, because the authors didn't find available such results. However, the model for this type of simulation needs more investigation, mainly, for high Reynolds number flows. The stream functions and temperature field are shown in Figure 22 and 23, respectively. They seem to present the waited behavior.

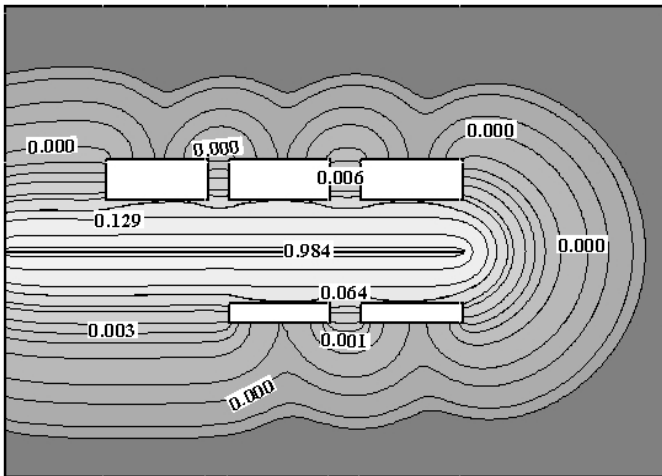


Figure 14. Pollutant concentration field at t=3

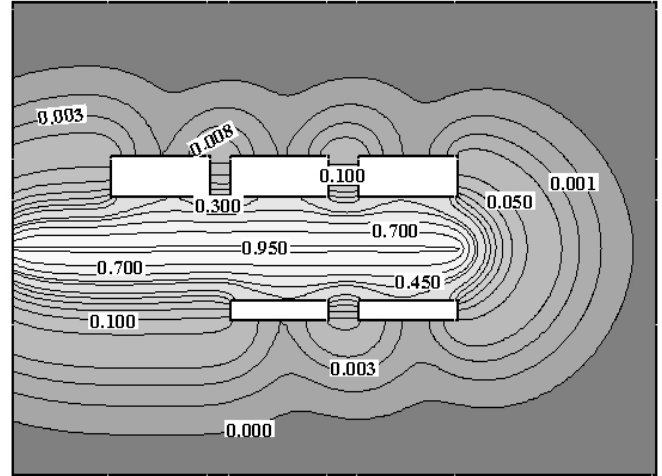


Figure 15. Pollutant concentration field at t = 12

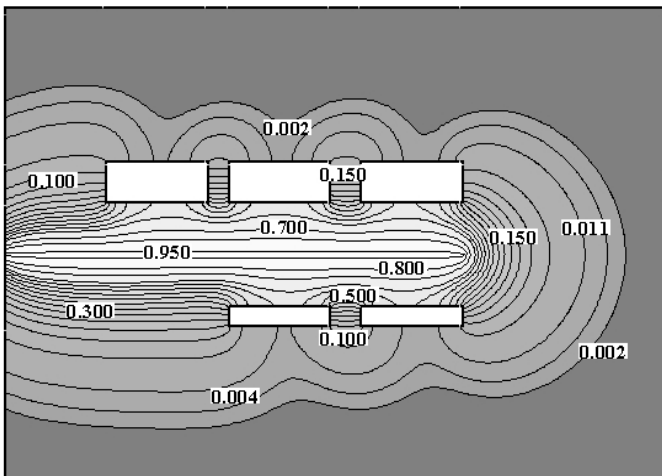


Figure 16. Pollutant concentration field at t=20

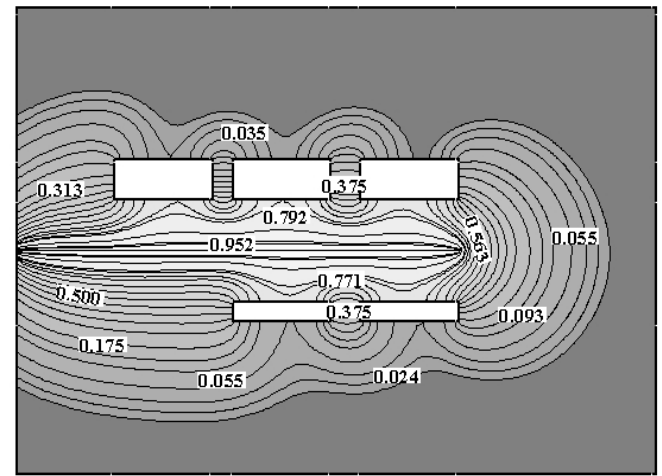


Figure 17. Pollutant concentration field at t=30

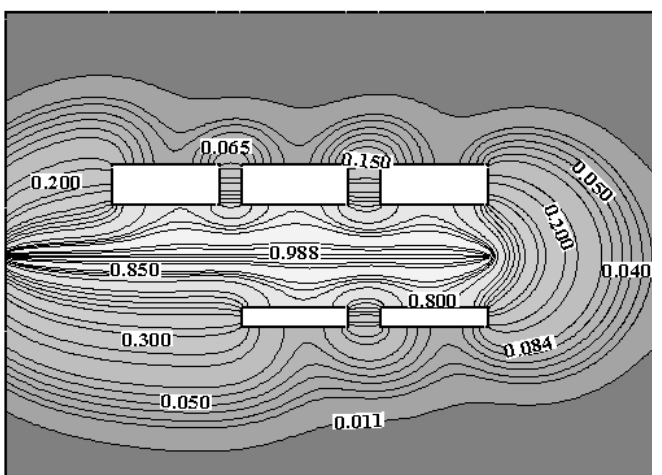


Figure 18. Pollutant concentration field at t = 35

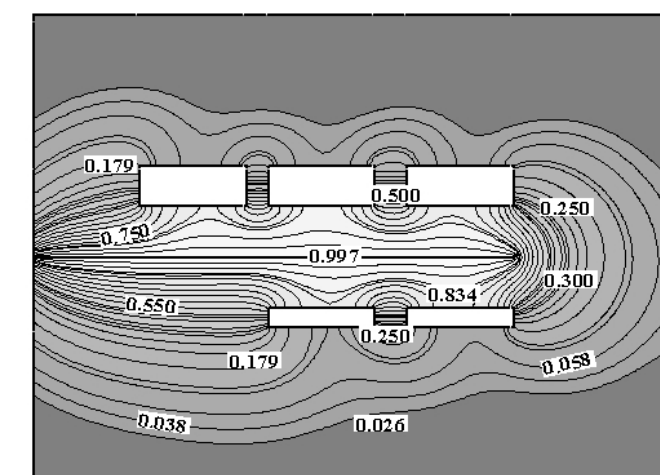


Figure 19. Pollutant concentration field at t = 40

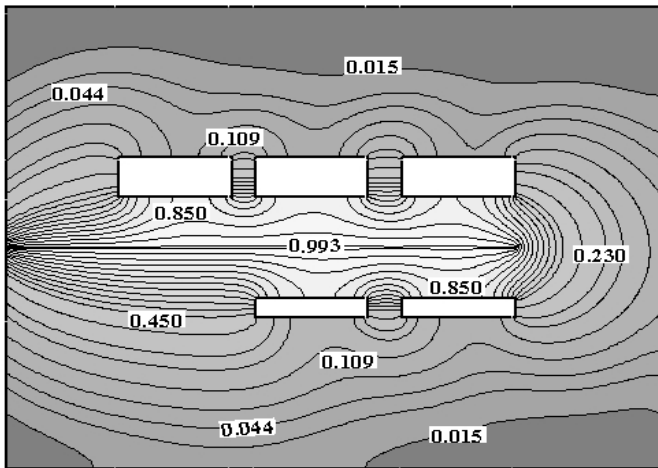


Figure 20. Pollutant concentration field at  $t=45$

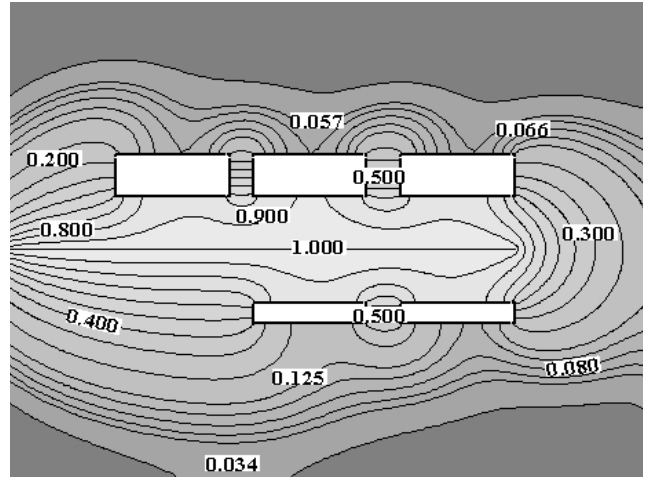


Figure 21. Pollutant concentration field at  $t=50$

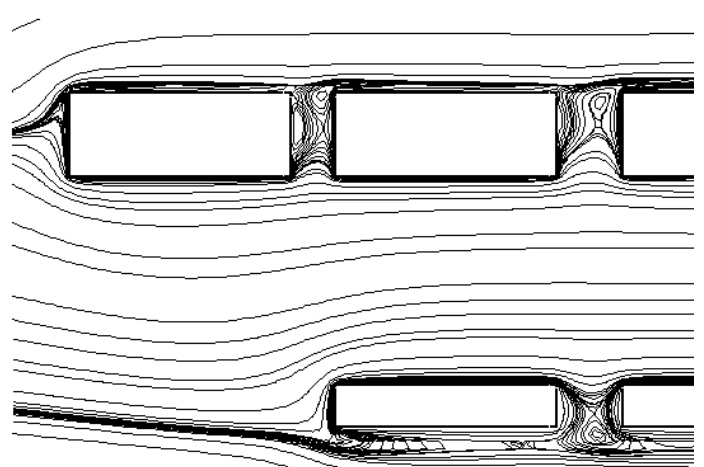
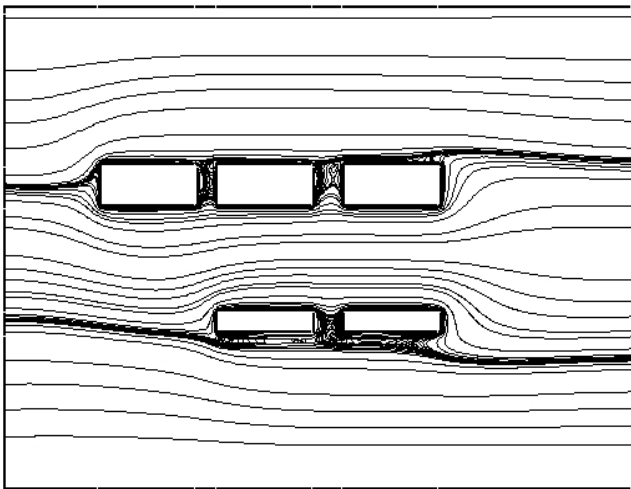


Figure 22. Stream functions

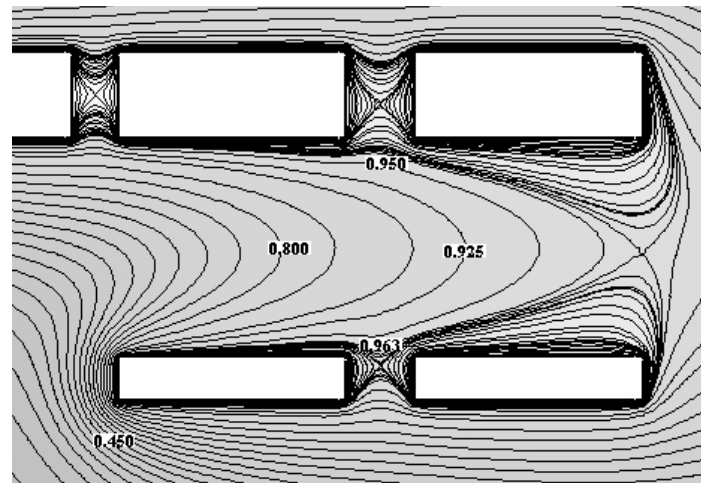
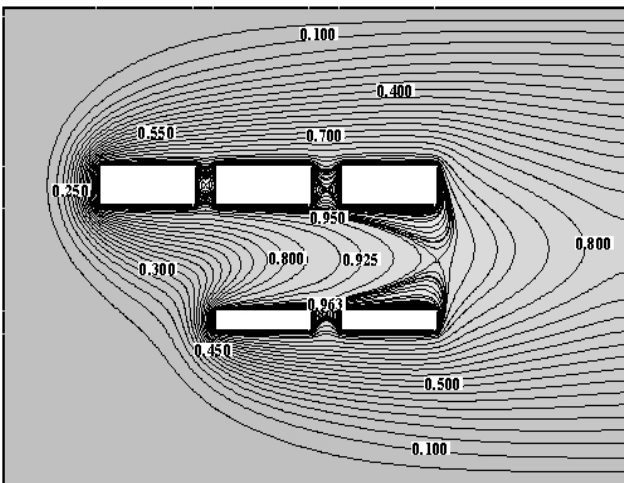


Figure 23. Temperature field

## 5. Conclusions

A simulation of pollutant dispersion horizontally around obstacles has been done. Qualitatively, the waited behavior of phenomenon seems have to be simulated. Some anomalous behavior of the concentration at the outflow section may be due to an insufficient size of the computational domain and a not appropriated boundary condition for velocity in the domain size considered. This has to be investigated in future works. Other arranges of obstacles, more

high Reynolds number flows, effect of natural convection and different boundary conditions are subjects for the sequence of this work as well as more realistic 3D flows have to be simulated.

## **6. Acknowledgement**

The authors would like to acknowledge the CNPq and FAPESP, process nº. 05/02301-1. The second author is CNPq – Brazil Scholarship.

## **7. References**

- Baliga, B.R. and Patankar, S.V., 1980, "A new finite-element formulation for convection-diffusion problems", *Numerical Heat Transfer*, Vol. 3, pp. 393-409.
- Baliga, B.R. and Patankar, S.V., 1983, "A control volume finite-element method for two-dimensional fluid flow and heat transfer", *Numerical Heat Transfer*, Vol. 6, pp. 245-261.
- Banaszek, J., 1989, "Comparison of control volume and Galerkin finite element methods for diffusion-type problems", *Numerical Heat Transfer*, Vol. 16, p. 59-78.
- Campos Silva, J. B., and Moura, L. F. M., 2001, "A Control-Volume-Finite-Element Method (CVFEM) for Unsteady, Incompressible, Viscous Fluid Flows", *Numerical Heat Transfer Part B: Fundamentals*, Vol. 40 N<sup>o</sup> 1, pp 61-82.
- Campos Silva, J. B., Aparecido, J. B. & Moura, L. F. M., 1999, "A Control Volume-Finite Element Method (CVFEM) for Unsteady Fluid Flows", *Anais do XV Congresso Brasileiro de Engenharia Mecânica - COBEM'99*, CD ROM, Águas de Lindóia/SP.
- Campos Silva, J.B. and Moura, L.F.M., 1997, "Numerical simulation of fluid flow by the control volume-finite element method", *Proceedings (in CD-ROM) of the XIV Brazilian Congress of Mechanical Engineering - COBEM97*, December 8-12, Bauru-SP, Brazil, paper code 041, 8p.
- Campos Silva, J.B., 1998, "Numerical simulation of fluid flow by the finite element method based on control volumes", *Doctorate Thesis (in Portuguese)*, State University of Campinas, Campinas, São Paulo, Brazil.
- Chidambaram, N., 1998, "Colocated-grid Finite Volume Formulation for the Large Eddy Simulation of Incompressible and Compressible Turbulent Flows", *Graduate College, Department of Mechanical Engineering, Iowa State University, Ames, Iowa, USA. M.Sc. Thesis.*
- Dhatt, G. and Touzot, G., 1984, "Une Presentation de la Méthode des Éléments Finis", *Deuxieme Édition*, Maloine S.A. Éditeur, Paris.
- Driver, D. M. and Seegmiller, H. L., 1985, "Features of a reattaching turbulent shear layer in divergent channel flow". *AIAA Journal*, Vol. 23, n<sup>o</sup> 2, pp. 163-173.
- Lima, R.C., 2005, "Simulação de Grandes Escalas de Escoamento Incompressíveis com Transferência de Calor e Massa por um Método de Elementos Finitos de Subdomínio", *Dissertação de Mestrado*, Universidade Estadual Paulista "Júlio de Mesquita Filho", Faculdade de Engenharia, Ilha Solteira, São Paulo, Brasil.
- Neofytou, P., Venetsanos, A.G., Rafailidis, S., Bartzis, J.G., 2006, "Numerical investigation of the pollution dispersion in an urban street canyon", *Environmental Modelling & Software* 21, pp. 525–531.
- Raw, M.J. and Schneider, G.E., 1986, "A skewed, positive influence coefficient up-winding procedure for control-volume-based finite-element convection-diffusion computation", *Numerical Heat Transfer*, Vol. 9, pp. 1-26.
- Savii, G.G., 1998, "Simulation of Air Pollution in Urban Area using Finite Element Methods", *Proceedings of 3rd International Conference on Technical Informatics – CONTI'98*, Timisoara, Romania, October, 1998. *Sci.Bull."Politehnica" Univ. Timisoara, Trans. Comp. Autom.*, Vol.43 (57), 1998, pp.96-99.
- Taylor, C., and Hughes, T.G., 1981, "Finite Element Programming of the Navier-Stokes Equations", *Pineridge Press Ltd, Swansea, U.K.*
- Walton, A., Cheng, A.Y.S., Yeung, W.C., 2002, "Large-eddy simulation of pollution dispersion in an urban street canyon – Part I: comparison with field data". *Atmospheric Environment* 36, pp. 3601–3613.
- Wong, A.Y.T., Chan, A.T. and So, E.S.P., 2002, "Computational Analysis of Wind Flow and Pollutant Dispersion in an Urban Street Canyon", *The 15th Engineering Mechanics Division Conference of the American Society of Civil Engineers at Columbia University in the City of New York, New York, NY June 2-5.*

## A TWO-PHASE ONE-DIMENSIONAL MODEL FOR SIMULATION OF FCC RISERS

**Ary Saad Brum**

Dep. de Física, Fundação Universidade Federal do Rio Grande, Av. Itália, km 8, Rio Grande-RS, 96201-900, Brazil  
arysaad@ibestvip.com.br

**Jeferson Avila Souza**

Dep. de Física, Fundação Universidade Federal do Rio Grande, Av. Itália, km 8, Rio Grande-RS, 96201-900, Brazil  
jasouza@furg.br

**José Viriato Coelho Vargas**

Dep. de Engenharia Mecânica, Fundação Universidade Federal do Paraná, CP. 19011, Curitiba-PR. Brazil  
jvargas@demec.ufpr.br

**Abstract.** *The FCC (Fluidized Catalytic Cracking) is one of the most important processes in a petroleum refinery plant. The numerical modeling of this process has been performed by several authors who have proposed different mathematical models and reported them in the literature. With the constant increase of computational capabilities, such models have become even more complex and with wider application. The different models address both fluid flow and cracking kinetic, varying from simple one phase and one-dimensional models to three-dimensional and three-phase models. Therefore, there is no common ground regarding the most adequate formulation, and advantages and drawbacks may be identified in each available model. In the present work, a relatively complex model reported in the literature is reproduced. Even though it is a one dimensional model, it includes many physical phenomena, like the dependence of the fluid properties on temperature and the transport equations formulation for both phases (gas and particulate). In a following stage, some simplifications in the mathematical formulation are included to the model and the results obtained with both formulations (with and without simplifications) are compared. The main goal of the present work is to establish a relationship between each included physical phenomenon and its real influence on the capability of the model to predict the products formation inside the riser reactor.*

**Keywords.** *FCC, Catalytic cracking, Finite differences, Fluid flow*

### 1. Introduction

The FCC (Fluidized Catalytic Cracking) is a key process for the petroleum refinery industry. With this process, heavy hydrocarbon molecules can be broken into lighter ones, and products of commercial interest such as gasoline and LPG are obtained in large scale and with relatively low costs. Another advantage of the FCC process is that the gasoil (feedstock), is a sub-product of the atmospheric and vacuum distillation and other petrochemical processes. This “sub-product” (gasoil), with a low commercial value, is then transformed into a high aggregated value products. The environmental advantages may also be cited; since industrial sub-products may be used as the feedstock for the FCC process, the amount of industrial trash discarded to the environment may be considerably diminished. Other important issue to be considered, is that the most important variables of the process, as mass flow and input temperature of the feedstock, among others, can be easily controlled, allowing a flexible adjusting of the production to the market variations.

The numerical modeling of the FCC (Fluidized Catalytic Cracking) has been reported in the literature by several authors. Some of these models simulates the whole unity, and are built with a mathematical formulation where the riser, the stripper and the regenerator were all considered in the simulation. Other simple models concentrates its analysis only on a specific equipment, developing for it a more detailed formulation. In this work, only the riser will be considered.

There are different kinds of models for the FCC riser. These models address its formulation for the fluid flow, for the cracking kinetic, or both of them. The complexity of these models may vary from a simple one phase and one-dimensional model to a three-dimensional and three-phase model. Depending on its industrial application, Souza, (2006) proposed that the different FCC models may be used for: (i) the development of a major model where the simulations of the entire conversion FCC unit is considered; (ii) testing or studying of specific catalytic cracking phenomena, such as catalyst deactivation, adsorption, kinetic models etc.; (iii) the search for better operating conditions (unit optimization), and (iv) phenomenological studies of the physical problem.

For the first type of model, the simulation is performed not only for the riser, but also for other equipments like the regenerator and the stripper. It is very important to have a fast solution for the riser model. This kind of model is normally used in the development of control systems, where the most desired feature is a low computational time solution. In these applications, it is not necessary an exact determination of the values of the variables, but only to determine qualitatively correct the system response to changes in the operating conditions. Examples of these kind of

models are the works of Han and Chung (2001) and Ali *et al.* (1997). The second type encompasses models used when the main goal is to study a specific physical phenomenon of the gasoil catalytic cracking, such as analysis of a catalyst deactivation function (Cerqueira *et al.* 1997a), the coke formation in the cracking process (Cerqueira *et al.* 1997b), the adsorption phenomena in the catalytic cracking [Martignoni, and Lasa, 2001) and the building of kinetic models (Ancheyta *et al.*, 1999). The third type of models strives for unit optimization. In this case, it is necessary to have a fast and sufficiently precise model that will be used to run several simulations searching for the best values for the input variables (mass concentrations, temperatures, etc.). The fourth and last type refers to models which try to include all important physical phenomena (Gao *at al.*, 2001; Chang and Zhou, 2003). These models are generally three-dimensional with two or three phases flow. The conservation equations are written for each phase of the fluid flow and the physical properties are not necessarily assumed constant. Turbulent models are normally used to describe the fluid flow.

Based on the great number of models, with different levels of complexities shown in the above paragraph, it is reasonable to conclude that there is no common ground regarding about the most adequate formulation for the FCC modeling. The four types of models discussed above may be used as a reference for choosing the most adequate model for a specific need, therefore great difference in the complexity may be observed among the models classified in a same case of application. For example, the one-dimensional models of Han and Chung (2001) and Ali *et al.* (1997) were considered both of first type, but conceptually they are completely different. While the Ali *et al.* (1997) formulates the riser as an one phase mixture fluid flow model, Han and Chung (2001) uses a two-phase model for the fluid flow simulation.

In the present work, part of the Han and Chung (2001) work is reproduced. In this model, an one-dimensional formulation for the riser, the regenerator and the stripper is presented, therefore only the riser formulation is reproduced in this first simulation. This model includes many physical phenomena, like the dependence of the fluid (and particle) properties (specific heat, viscosity, etc.) on temperature and the transport equations formulation for both phases (gas and particulate). These phenomena are normally simplified in simpler models.

At this moment, the species conservation, gasoil energy equation and catalyst energy equation are already implemented. The specific heat of the gasoil dependence on the temperature and the feedstock was the first property calculation added to the model. Two simulations were performed: one with the full specific heat temperature and feedstock dependence and another with a fixed average value for the gas phase specific heat. The comparison of the results showed a small difference in the products mass fraction predictions at the riser output section. The momentum equations for the gasoil and the catalyst are also already implemented and working, therefore they where not validated yet. The comparison solution between the two-phase model and the one-phase model will be performed in a following work.

## 2. Mathematical model

Figure 1 shows a schematic diagram of the studied system, with its simple geometry, where  $H$  is the length of the riser in the flow direction. Regenerated catalyst and gasoil enter the system from the bottom of the riser, whereas the product lumps and the deactivated catalyst leave it from the top of the riser.

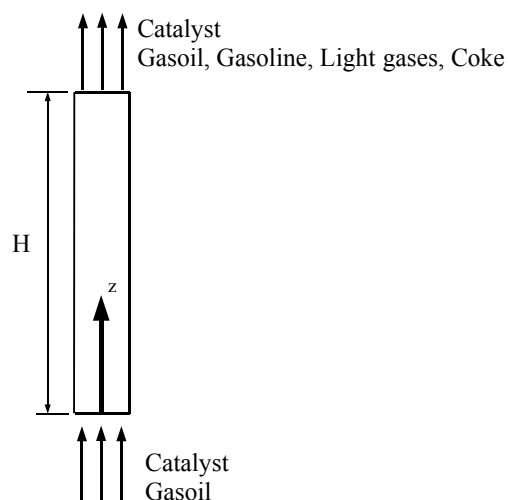


Figure 1 – Problem sketch

In this first simulation, only the riser reactor model will be simulated. The present formulation also considers the following assumptions: incompressible, laminar, one-dimensional flow and homogeneous mixture. The gasoil density

and bed porosity were also considered constants, but the gas phase specific heat was considered as to be a function of the bed temperature.

The kinetic model that will determine the products mass fraction profiles along the riser was based on a 4-lumps, as shown in Fig. 2. All the pre-exponential constants ( $k_{ij}$ ) of the kinetic model and heat of reactions ( $\Delta H$ ) are available in Han and Chung (2001).

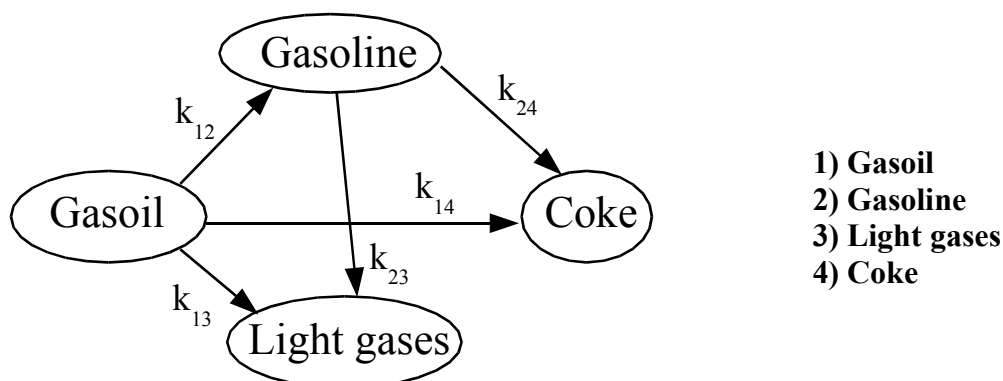


Figure 2 – Lumped kinetic model

According with Han and Chung (2001), the lumps mass fraction profiles along the riser reactor can be predicted by

$$\frac{\partial y_i}{\partial z} = \frac{\rho_c \epsilon_c A \phi_c}{\dot{m}_g} \Omega_i \quad \text{with } i = 1 \text{ to } 4 \quad (1)$$

where

$$\begin{aligned} \Omega_1 &= -(k_{12} + k_{13} + k_{14}) y_1^2 \\ \Omega_2 &= k_{12} y_1^2 - k_{23} y_2 - k_{24} y_2 \\ \Omega_3 &= k_{13} y_1^2 + k_{23} y_2 + k_{34} y_3 \\ \Omega_4 &= k_{14} y_1^2 + k_{24} y_2 + k_{34} y_3 \end{aligned} \quad (2)$$

In Eqs. (1-2)  $y$  is the lump mass fraction,  $\rho$  the density,  $\epsilon$  the volume fraction,  $\phi$  the catalyst deactivation function,  $z$  the axial coordinate,  $A$  the riser cross section area,  $\dot{m}$  the mass flow rate,  $\Omega$  the reaction term and  $k$  the pre-exponential constants. The sub-indexes  $c$ , and  $g$ , indicates catalyst and gas, respectively.

The temperature profiles for the gasoil and catalyst are determined by the following energy equations

$$\frac{\partial T_c}{\partial z} = \frac{A h_p}{\dot{m}_c C p_c} (T_g - T_c) \quad (3)$$

$$\frac{\partial T_g}{\partial z} = \frac{A}{\dot{m}_g C p_g} [h_p (T_g - T_c) + \rho_c \epsilon_c \Omega_T] \quad (4)$$

where

$$\Omega_T = -\phi_c (\Delta H_1 k_{12} y_1^2 + \Delta H_2 k_{13} y_1^2 + \Delta H_3 k_{14} y_1^2 + \Delta H_4 k_{23} y_2 + \Delta H_5 k_{24} y_2) \quad (5)$$



In Eqs. (3-5),  $T$  is the temperature,  $h_p = h A_p$  the interface heat transfer coefficient between gas phase and catalyst phases,  $\Omega_T$  energy reaction term,  $C_p$  the specific heat and  $\Delta H$  heat of reaction.

The Han and Chung (2001) work considers all physical properties as functions of temperature, pressure or feedstock. In the present stage of development of this work, only the gaseous phase specific heat dependency with the feedstock and temperature was incorporated to the model.

The gas-phase heat capacities of gasoil and gasoline lump are calculated as (Lee and Kesler, 1988):

$$Cp_g = \beta_1 + \beta_2 T + \beta_3 T^2 \quad (6)$$

where

$$\beta_1 = -1.492343 + 0.124432 K_f + \beta_4 \left( 1.23519 - \frac{1.04025}{S_g} \right) \quad (7)$$

$$\beta_2 = -7.53624 \times 10^{-4} \left[ 2.9247 - (1.5524 - 0.05543 K_f) + \beta_4 \left( 6.0283 - \frac{5.0694}{S_g} \right) \right] \quad (8)$$

$$\beta_3 = 1.356523 \times 10^{-6} (1.6946 + 0.0884 \beta_4) \quad (9)$$

$$\beta_4 = \left[ \left( \frac{12.8}{K_f} \right) \left( 1 - \frac{10}{K_f} (S_g - 0.885)(S_g - 0.7) \times 10^4 \right) \right]^2 \quad \text{for } 10 < K_f < 12.8 \quad (10)$$

$\beta_4 = 0$  for all others cases

The density  $S_g$  and  $K_f$  are defined as

$$S_g = \frac{141.5}{\text{API} + 131.5} \quad (11)$$

$$K_f = \frac{(1.8 T_{Me})^{1/3}}{S_g} \quad (12)$$

where  $T_{Me}$  is the mean average boiling temperature.

For the light gases lumps, the specific heat is calculated by

$$Cp_g = 0.2457 + 5.3 \times 10^{-3} T - 2.1527 \times 10^{-6} T^2 \quad (13)$$

### 3. Results

The ordinary differential equation system formed by Eqs. (1), (3) and (4) was solved with a fourth-order Runge-Kutta algorithm (Kincaid, 1991). Fig. 3. shows the computational domain with the appropriated boundary conditions for the problem. For the transport equation of the kinetic model (Eq. (1)), at the input section of the riser, it is considered that only gasoil is entering. Therefore the mass fraction of gasoil is set to 1, while for the other three lumps the mass fractions are set to zero. The catalyst and gasoil (gas) temperatures are a known operating condition at the input section of the riser, thus a prescribed value is set to this variables in the mathematical model. The operating conditions for all simulations are presented in Table 1.

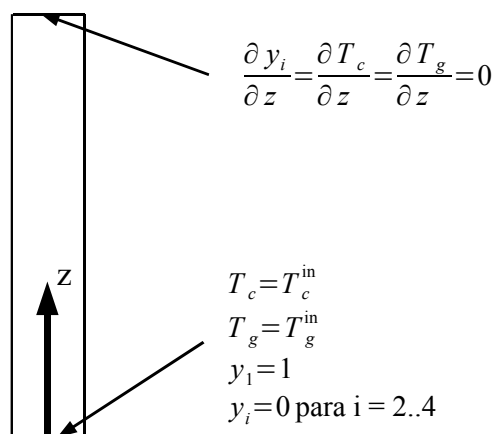


Figure 3 – Boundary conditions and computational domain

At the riser output section, it was considered that the mass fractions of the lumps are no longer varying, and the derivative in the flow direction is set to zero for each lump. The same boundary condition is prescribed for both gas phase and catalyst temperatures at the top of the riser, as also shown in Fig. 3.

Table 1: General operating conditions for all simulations

Geometry	
Length (m)	30
Diameter (m)	1,1
Feedstock	
Gas oil mass flux (kg/s)	50
Catalyst mass flux (kg/s)	350
Physical parameters	
Catalyst density (kg/m <sup>3</sup> )	1410
Catalyst specific heat (kJ/kg K)	1,15
Catalyst input temperature (°K)	910
Gasoil specific heat (kJ/kg K)	4,5*
Gasoil input temperature (°K)	670
Heat exchange coefficient between phases (kJ/s K)	1 x 10 <sup>2</sup>

\* for the simulations with constant gas phase specific heat.

The comparison of the Han and Chung (2001) results and those obtained with the present formulation are shown in Fig. 4. Even though, at this stage of the simulations both models are considerably different, a qualitative agreement between the results for both the mass fractions and temperatures profiles along the riser may be observed in Fig. 4.

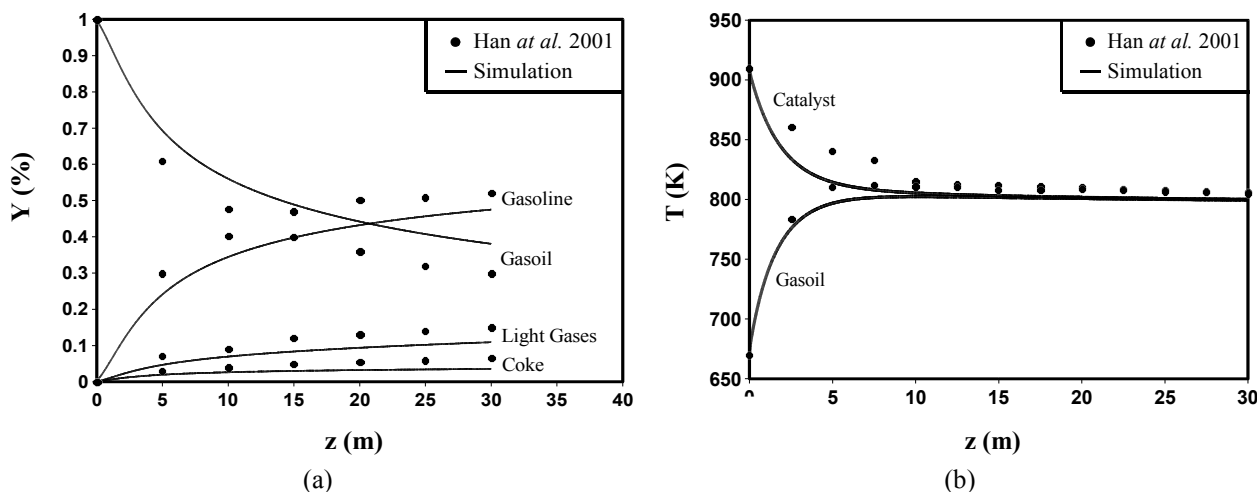


Figure 4 – Comparison between the model and the Han and Chung (2001) results

The differences between the two models do not allow a quantitative validation of the present program, therefore the good qualitative results obtained are sufficient for the analysis performed in this paper.

The main goal of the present paper is to analyze the influence of the dependence of the specific heat of the gas phase with the temperature. This dependence of the gas phase specific heat on the temperature is added to the model by the Eqs. (6-12). This set of equations also incorporates to the model the dependence of the gasoil specific heat with the feedstock, therefore this influence will not be explored in the present work.

The set of equations needed to incorporate the gas phase specific heat dependence on the temperature (Eqs. (6-12)) also include some complications to the simulation solution. Even though only algebraic equations are added to the model, these equations are dependent of the gas phase temperature and for this reason, also included in the differential system of equations to be solved (Eqs. (1, 3-4)). The model will be considerably simplified if we set a constant average value for the gas phase specific heat, and if no significant differences between the two solutions were observed, this simplification should be the best option for the simulation.

In order to quantify this influence, the program developed with the model presented in this work was run with and without the specific heat dependence on the temperature. Fig. 5. shows the comparison of both results for the mass fraction and temperature profiles along the riser.

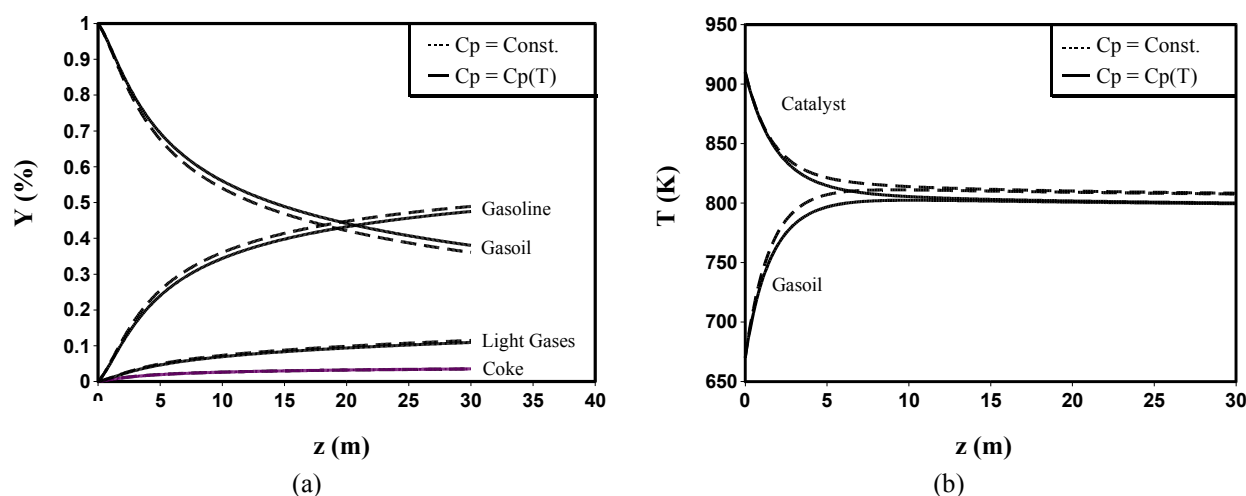


Figure 5 – Mass fraction and temperature profiles along the riser: (a)  $C_p = \text{constante}$ , (b)  $C_p = C_p(T)$

As can be seen in Fig. 5, a small deviation between the results obtained with the two simulations can be observed. Comparing Figs. 5a and 5b, it can be noticed that at the input section of the riser (up to 5m approximately) no significant differences may be observed between the two simulations. In this region, with high catalyst temperature and low gasoil temperature, both mass fraction and temperature profiles are almost identical. After this input section, a small deviation between the profiles for both mass fractions and temperatures are observed, therefore this difference remains less than 7%. Table 2 shows the comparison between the mass fraction predictions at the riser output section of the riser. Table 2 shows that only a small difference between the predicted mass fractions of the lumps were found.

Table 2: Gas phase specific heat influence on the mass fraction at the riser output section

Y	For $C_p = \text{constant}$	For $C_p = C_p(T)$	%error
			$\left  \frac{C_p - C_p(T)}{C_p} \right $
Gasoil	0,3600	0,3803	5,64%
Gasoline	0,4893	0,4751	2,90%
Light Gases	0,1149	0,1092	4,96%
Coke	0,0352	0,0354	0,57%

#### **4. Conclusions**

In this work, a simplified fluid flow and kinetic model for a FCC riser was used to study the influence of gas mixture specific heat on the product profiles along the riser reactor. Two simulation were performed: one with a constant value for the specific heat of the gas phase and another with the specific heat formulated as a function of the gas phase temperature. The results obtained with this comparison showed that the difference between the two models remains less than 7%, what considering the difficult and uncertainty of the corrected prediction of these profiles when they are compared with experimental data, the use of such a complex formulation for the specific heat mixture calculation may not aggregate significant improvement to the model prediction. Since the two models were not compared with the experimental data, but only among them, and the difference between the two predictions is quite small, it is not possible to concern about the best formulation (with or without specific heat dependence on temperature), but only about their advantages and disadvantages. The constant specific heat formulation has a more simplified mathematical model, and consequently an easy and fast solution, while the temperature dependent formulation is more phenomenological corrected and may be the best choice for theoretical studies of the physics of catalytic cracking phenomenal inside the riser reactor, but according with the results obtained in this work do not increase the products profiles prediction along the riser.

#### **5. Acknowledgment**

The authors acknowledge with gratitude the FAPERGS – Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul.

#### **6. References**

- Ali, H., Rohani, S. and Corriou, J. P., 1997, "Modeling and control of a riser type fluid catalytic cracking (FCC) unit", *Trans IchemE*, vol. 75, Part A, 1997.
- Ancheyta J, Lopez-Isunza F, Aguilar-Rodriguez E., 1999, "5-Lump kinetic model for gasoil catalytic cracking", *Applied Catalysis A-General*, vol 177, n. 2 pp. 227-235.
- Cerqueira H. S., Biscaia E. C. Jr., Aguiar E. F. S. 1997a, "Mathematical modeling of deactivation by coke formation in the cracking of gasoil", *Studies in Surface Science and Catalysis*, vol. 111, pp. 303-310.
- Cerqueira H. S., Biscaia E. C., Sousa-Aguiar E. F., 1997b, "Mathematical modeling and simulation of catalytic cracking of gasoil in a fixed bed: Coke formation", *Applied Catalysis A-General*, vol. 164, n 1-2, pp. 35-45.
- Chang S. L. and Zhou C. Q. 2003, "Simulation of FCC riser flow with multiphase heat transfer and cracking reactions", *Computational Mechanics*, vol. 31, n. 6, pp. 519-532.
- Gao J. S., Xu C. M., Lin S. X., Yang G. H., Guo Y. C., 2001, "Simulations of gas-liquid-solid 3-phase flow and reaction in FCC riser reactors", *AIChE Journal*, vol. 47, n. 3, pp. 677-692.
- Han, I. S. and Chang, B. C., 2001, Dynamic modeling and simulation of a fluidized catalytic cracking process Part I: process modeling, *Chemical Engineering Science*, vol. 56, pp. 1951-1971.
- Kincaid, D. and Cheney, W., 1991, "Numerical analysis mathematics of scientific computing", Cole Publishing Company, Pacific Grove, California.
- Lee, B. I., and Kesler, M. G., 1988, "Isobaric teat capacity of petroleum fraction liquids/vapors", *Technical data book-petroleum refining*, Tulsa, OK: American Petroleum Institute.
- Martignoni W, de Lasa H. I., 2001, "Heterogeneous reaction model for FCC riser units", *Chemical Engineering Science*. vol. 56, n. 2, pp. 605-612.
- Souza, J. A., Vargas, J. V. C., Meien, O. F. von, Martignoni, W., Amico, S. C., 2006, "A simplified two-dimensional model for the simulation, control and optimization of FCC risers". *AIChE Journal*, vol. 52, n. 5, pp. 1895-1905.

## PERFORMANCE OF THE $k$ - $\epsilon$ AND REYNOLDS STRESS MODELS IN TURBULENT FLOWS WITH VISCOELASTIC FLUIDS.

### P. R. Resende

Centro de Estudos de Fenómenos de Transporte  
Faculdade de Engenharia da Universidade do Porto  
Rua Dr. Roberto Frias s/n,  
4200 – 465 Porto, Portugal  
e-mail: [resende@fe.up.pt](mailto:resende@fe.up.pt)

### F. T. Pinho

Universidade do Minho  
Largo do Paço  
4704-553 Braga, Portugal  
e-mail: [fpinho@fe.up.pt](mailto:fpinho@fe.up.pt)

### D. O. Cruz

DEM – Universidade Federal do Pará  
Campus Universitário do Guamá  
66075 – 900 Belém, Pará, Brazil  
e-mail: [doac@ufpa.br](mailto:doac@ufpa.br)

**Abstract:** The performance of a newly developed low Reynolds number second order closure for viscoelastic fluids is compared with that of an existing  $k$ - $\epsilon$  model using experimental data for fully-developed flow of various polymer solutions in circular pipes of Escudier *et al.* (1999) and Resende *et al.* (2005). New developments were made to account separated flows, removing the dependence of the velocity gradient by the friction velocity in the recirculation zone. The fluids simulated are the following aqueous polymeric solutions: 0.125% by weight of polyacrilamide (PAA), 0.2% of xantham gum (XG), 0.25% of carboxymethyl cellulose (CMC) and a blend of 0.09% CMC and 0.09% XG. As far as the rheological constitutive equation is concerned, both models are based on modified generalized Newtonian model developed by Pinho (2003), and wall approximation effects required by the low Reynolds number approach are taken into account by the damping function developed by Cruz and Pinho (2003), which includes effects of shear thinning and Trouton ratio thickening.

The low Reynolds Reynolds stress model is seen to perform better than the  $k$ - $\epsilon$  model both in terms of mean flow quantities (friction factor and mean velocity) as well as the Reynolds stress predictions.

**Keywords:** Turbulence model, drag reduction, polymer solutions, second order closure.

### 1. Introduction

A Reynolds stress model is developed to predict turbulent flows with viscoelastic fluids and is tested here in fully-developed channel flows of polymer solutions. First-order turbulence models have shortcomings when it comes to predicting flows with separation or streamline curvature, amongst other things (see an early revision in Patel *et al.* (1984)). In addition, viscoelastic fluids in duct flows exhibit stronger anisotropy of the Reynolds stress tensor than Newtonian fluids do, which further accentuates the shortcomings of some first-order closures to properly deal with them. The use of anisotropic first-order models can offset some of these disadvantages (Park *et al.* (2003), Craft *et al.* (1996)). Now that simple first-order turbulence closures are available for viscoelastic fluids (Resende *et al.* (2006)), and have been tested in duct flow, it is time to evolve to higher-order closures which will enable the handling of more complex flows, such as flows with separation.

The first turbulence models for viscoelastic fluids date from the 1970's with Mizushima *et al.* (1973), Durst and Rastogi (1977) and Poreh and Hassid (1978). Their scope was rather limited because they depended on parameters that needed to be selected for each fluid in each flow situation. In the 1980's new turbulence models appeared (Politis (1989), Malin (1997)), but were limited to the inelastic fluids of variable viscosity. To overcome this limitation Pinho (2003) and Cruz and Pinho (2003) developed a turbulence model using a modified version of the generalized Newtonian constitutive equation in order to account elastic effects and therefore introduce rheological parameters into the turbulence model. Subsequent developments of that model were introduced by Cruz *et al.* (2004) and Resende *et al.* (2006).

The present Reynolds stress closure is a step forward in the hierarchy of models for viscoelastic fluids and is based on the model of Lai and So (1990) for Newtonian fluids. This model was selected because it combined simplicity with a low Reynolds number capability, essential for use with viscoelastic fluids for which no universal law of the wall exists.

The performance of the model is tested against experimental data for dilute polymeric aqueous solutions of Escudier *et al.* (1999) and Resende et al (2006).

The next section presents the governing equations for viscoelastic turbulent flow. The terms requiring modeling are identified in the Section 3 with the corresponding closures. The results of the numerical simulations and their discussion are presented in Section 4. The paper closes with a summary of the main conclusions.

## 2. Governing equations

The governing equations are the continuity and momentum equations and the Reynolds stress is calculated by its transport equation. The extra stress of the fluid is given by a generalized Newtonian constitutive equation modified by Pinho (2003). The momentum equation is

$$\rho \frac{\partial U_i}{\partial t} + \rho U_k \frac{\partial U_i}{\partial x_k} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_k} \left( 2\bar{\mu} S_{ik} - \overline{\rho u_i u_k} + 2\overline{\mu' s_{ik}} \right) \quad (1)$$

where  $p$  is the pressure,  $\rho$  is the densit,  $\bar{\mu}$  is the average molecular viscosity,  $u_i$  is the  $i$ -velocity component and  $S_{ij}$  is the rate of deformation tensor defined as  $S_{ij} \equiv \left( u_{i,j} + u_{j,i} \right) / 2$ . Here, and elsewhere, small letters or a prime indicate fluctuations, capital letters or an overbar designate time-average quantities and a hat is used for instantaneous values. The average molecular viscosity ( $\bar{\mu}$ ) is given by equation (2) which combines the pure viscometric shear viscosity contribution ( $\eta_v$ ) in Eq. (3), with the high Reynolds number time-average molecular viscosity contribution ( $\bar{\mu}_h$ ) of Eq. (4).

$$\bar{\mu} = f_v \bar{\mu}_h + (1 - f_v) \eta_v, \quad (2)$$

$$\eta_v = K_v [\dot{\gamma}^2]^{(n-1)/2}, \quad (3)$$

$$\begin{aligned} \bar{\mu}_h = & \left( C_\mu \rho \right)^{3m(m-1)A_2/(8+3m(m-1)A_2)} \times 2^{4m(m-1)A_2/(8+3m(m-1)A_2)} \\ & \times k^{6m(m-1)A_2/(8+3m(m-1)A_2)} \times \varepsilon^{[(8-3(M-1)A_2)m]/(8+3m(m-1)A_2)} \times B^{8/(8+3m(m-1)A_2)} \end{aligned} \quad (4)$$

where  $B = [K_v K_e / A_e]^{1-m} 2^{[(n-1)-m(n+1)]/2} \rho^m$ ,  $m = (n + p - 2) / (n + p)$ ,  $K_v$  and  $n$  are the power law parameters,  $f_v$  is a damping function equal to  $f_\mu$  and  $k$  and  $\varepsilon$  represent the turbulent kinetic energy and its rate of dissipation, respectively. This model for  $\bar{\mu}$  was derived by Pinho (2003).

The pseudo- elastic stress in the momentum equation ( $2\overline{\mu' s_{ik}}$ ) and the new terms of the transport equation for Reynolds stress ( $\overline{\rho u_i u_k}$ ) require specific modelling which is presented in the next section.

The transport equation for Reynolds stress is

$$\begin{aligned} \rho \frac{D\overline{u_i u_j}}{Dt} + \overline{\rho u_j u_k} \frac{\partial U_i}{\partial x_k} + \overline{\rho u_i u_k} \frac{\partial U_j}{\partial x_k} = & -\rho \frac{\partial}{\partial x_k} \overline{u_i u_j u_k} - \left( \frac{\partial}{\partial x_i} \overline{p' u_j} + \frac{\partial}{\partial x_j} \overline{p' u_i} \right) + \\ & + p' \left( \frac{\partial \overline{u_j}}{\partial x_i} + \frac{\partial \overline{u_i}}{\partial x_j} \right) + \overline{\mu} \frac{\partial^2 \overline{u_i u_j}}{\partial x_k \partial x_k} - 2\overline{\mu} \frac{\partial \overline{u_i}}{\partial x_k} \frac{\partial \overline{u_j}}{\partial x_k} + \frac{\partial \overline{\mu}}{\partial x_k} \frac{\partial \overline{u_i u_j}}{\partial x_k} \\ & + \frac{\partial \overline{\mu}}{\partial x_k} \left( \frac{\partial \overline{u_k u_j}}{\partial x_i} + \frac{\partial \overline{u_k u_i}}{\partial x_j} - 2\overline{u_k s_{ij}} \right) + \overline{\mu'} \frac{\partial^2 \overline{u_i u_j}}{\partial x_k \partial x_k} - 2\overline{\mu'} \frac{\partial \overline{u_i}}{\partial x_k} \frac{\partial \overline{u_j}}{\partial x_k} \\ & + \frac{\partial \overline{\mu'}}{\partial x_k} \frac{\partial \overline{u_i u_j}}{\partial x_k} + \frac{\partial \overline{\mu'}}{\partial x_k} \left( u_j \frac{\partial \overline{u_k}}{\partial x_i} + u_i \frac{\partial \overline{u_k}}{\partial x_j} \right) + \overline{\mu' u_j} \frac{\partial^2 \overline{U_i}}{\partial x_k \partial x_k} \\ & + \overline{\mu' u_i} \frac{\partial^2 \overline{U_j}}{\partial x_k \partial x_k} + u_j \frac{\partial \overline{\mu'}}{\partial x_k} \left( \frac{\partial \overline{U_i}}{\partial x_k} + \frac{\partial \overline{U_k}}{\partial x_i} \right) + u_i \frac{\partial \overline{\mu'}}{\partial x_k} \left( \frac{\partial \overline{U_j}}{\partial x_k} + \frac{\partial \overline{U_k}}{\partial x_j} \right) \end{aligned} \quad (5)$$

## 3. Closures for non-Newtonian terms

We start with the two molecular related stresses in the momentum equation. Term  $2\overline{\mu's'_{ik}}$  was designated a pseudo-elastic stress by Cruz *et al.* (2004) where a closure was proposed in the context of their  $k-\varepsilon$  model. Inspired by their derivation, we propose here an extended version consistent with the use of a full Reynolds stress model. Therefore, the expression for the pseudo-elastic stress used here is

$$2\overline{\mu's'_{ij}} = \tilde{C} \frac{K_v K_e}{A_\varepsilon^{p-1}} \left[ \frac{\rho |u_i u_j|}{2\bar{\mu}} \frac{\partial U_i}{\partial x_j} \right]^{\frac{(p+n-2)}{2}} \times \frac{1}{L_c} \times \frac{\overline{u_i u_j}}{\sqrt{|\overline{u_i u_j}|}} \quad (6)$$

with

$$\tilde{C} = (1 + C_0)^{p+n-2} - 1 \quad (7)$$

The pseudo-elastic stress vanishes in the Newtonian limit ( $n=1$  and  $p=1$ ), an effect properly accounted for by parameter  $\tilde{C}$ , which depends on parameter  $C_0$ . This parameter takes on a new value different from that in Cruz *et al.* (2004) and was obtained from optimization of the Reynolds stress model predictions.

$L_c$  is a spatial scale of turbulence,  $L_c = u_R^3/\varepsilon$ , accounting high Reynolds number flow away from the wall and the damping effect of the approaching wall. The velocity scale,  $u_R$ , is defined by

$$u_R^2 = \frac{k}{\left| \exp\left(-\left(k/u_w^2\right)^\alpha\right) - 1 \right|^{1/\alpha}} \quad (8)$$

where  $\alpha=2$ .  $u_w$  is a wall velocity scale here given by  $u_w = (\nu_w \cdot \varepsilon_w)^{1/4}$ , based in the viscous length scale,  $u = \nu/l$ , and related to the energy-dissipating eddies.

In the Reynolds stress transport equation we identify terms which are identical in form to those for Newtonian fluids and new terms associated with the non-Newtonian fluid characteristics, which are those involving fluctuating viscosities. Of these, according to the order of magnitude analysis made by Pinho (2003), the following terms could be neglected:

$$u_j \frac{\partial \mu'}{\partial x_k} \left( \frac{\partial U_i}{\partial x_k} + \frac{\partial U_k}{\partial x_i} \right) + u_i \frac{\partial \mu'}{\partial x_k} \left( \frac{\partial U_j}{\partial x_k} + \frac{\partial U_k}{\partial x_j} \right) \quad (9)$$

$$\overline{\mu' u_j} \frac{\partial^2 U_i}{\partial x_k \partial x_k} + \overline{\mu' u_i} \frac{\partial^2 U_j}{\partial x_k \partial x_k} \quad (10)$$

The remaining non-Newtonian terms are modelled as follows assuming high Reynolds number turbulence:

$$\frac{\partial \bar{\mu}}{\partial x_k} \frac{\partial \overline{u_i u_j}}{\partial x_k} + \frac{\partial \mu'}{\partial x_k} \frac{\partial \overline{u_i u_j}}{\partial x_k} \sim C_{V1} \times \frac{\partial \bar{\mu}}{\partial x_k} \frac{\partial \overline{u_i u_j}}{\partial x_k} \quad (11)$$

$$\frac{\partial \bar{\mu}}{\partial x_k} \left( \frac{\partial \overline{u_k u_j}}{\partial x_i} + \frac{\partial \overline{u_k u_i}}{\partial x_j} - 2\overline{u_k s_{ij}} \right) + \frac{\partial \mu'}{\partial x_k} \left( u_j \frac{\partial \overline{u_k}}{\partial x_i} + u_i \frac{\partial \overline{u_k}}{\partial x_j} \right) \sim C_{V2} \times \frac{\partial \bar{\mu}}{\partial x_k} \left( \frac{\partial \overline{u_k u_j}}{\partial x_i} + \frac{\partial \overline{u_k u_i}}{\partial x_j} \right) \quad (12)$$

where  $C_{V1}$  and  $C_{V2}$  are parameters to be quantified later.

All the other terms are Newtonian and are modelled according to the original model of Lai and So (1990):

- Turbulent diffusion of the Reynolds stresses,  $D_{ij}^T$ ,

$$-\rho \frac{\partial}{\partial x_k} \overline{u_i u_j u_k} = \rho \frac{\partial}{\partial x_k} \left\{ C_s \frac{k}{\varepsilon} \left[ \overline{u_i u_l} \frac{\partial \overline{u_j u_k}}{\partial x_l} + \overline{u_j u_l} \frac{\partial \overline{u_k u_i}}{\partial x_l} + \overline{u_k u_l} \frac{\partial \overline{u_i u_j}}{\partial x_l} \right] \right\} \quad (13)$$

- The molecular diffusion of the Reynolds stresses,  $D_{ij}^v$ ,

$$\bar{\mu} \frac{\partial^2 \overline{u_i u_j}}{\partial x_k \partial x_k} + \overline{\mu'} \frac{\partial^2 u_i u_j}{\partial x_k \partial x_k} \approx \bar{\mu} \frac{\partial^2 \overline{u_i u_j}}{\partial x_k \partial x_k}, \quad (14)$$

assuming at this stage a lack of correlation between fluctuating viscosity and the fluctuating second derivative of the Reynolds stress.

- The dissipation of the Reynolds stresses,  $\varepsilon_{ij}$ ,

$$-2\bar{\mu} \frac{\partial u_i}{\partial x_k} \frac{\partial u_j}{\partial x_k} - 2\overline{\mu'} \frac{\partial u_i}{\partial x_k} \frac{\partial u_j}{\partial x_k} = \rho \varepsilon_{ij}, \quad (15)$$

is modelled considering anisotropy and directional effects near walls, by

$$\varepsilon_{ij} = \frac{2}{3} \varepsilon (1 - f_{w,1}) \delta_{ij} + \frac{f_{w,1} (\varepsilon/k) [\overline{u_i u_j} + \overline{u_i u_k n_k n_j} + \overline{u_j u_k n_k n_i} + n_i n_j \overline{u_k u_l n_k n_l}]}{1 + 3\overline{u_k u_l n_k n_l} / 2k}, \quad (16)$$

- The pressure-strain,  $\phi_{ij}^*$ ,

$$-\left( \frac{\partial}{\partial x_i} \overline{p' u_j} + \frac{\partial}{\partial x_j} \overline{p' u_i} \right) + \overline{p' \left( \frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j} \right)} = \phi_{ij}^*, \quad (17)$$

is modelled by equation (18) considering the high Reynolds number contribution,  $\phi_{ij}$ , and the wall approximation contribution,  $\phi_{ij,w} f_{w,1}$ .

$$\phi_{ij}^* = \phi_{ij} + \phi_{ij,w} f_{w,1}, \quad (18)$$

with

$$\phi_{ij,1} = -C_1 \frac{\varepsilon}{k} \left( \overline{u_i u_j} - \frac{2}{3} k \delta_{ij} \right) - \alpha \left( P_{ij} - \frac{2}{3} \tilde{P} \delta_{ij} \right) - \beta \left( D_{ij} - \frac{2}{3} \tilde{P} \delta_{ij} \right) - \gamma k \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right), \quad (19)$$

$$\phi_{ij,w} = C_1 \frac{\varepsilon}{k} \left( \overline{u_i u_j} - \frac{2}{3} k \delta_{ij} \right) - \frac{\varepsilon}{k} \left( \overline{u_i u_k n_k n_j} + \overline{u_j u_k n_k n_i} \right) - \alpha^* \left( P_{ij} - \frac{2}{3} \tilde{P} \delta_{ij} \right), \quad (20)$$

where

$$D_{ij} = \left[ \overline{u_i u_k} \frac{\partial U_k}{\partial x_j} + \overline{u_j u_k} \frac{\partial U_k}{\partial x_i} \right], \tilde{P} = \frac{1}{2} P_{ii}, \alpha = \frac{(8 + C_2)}{11}, \beta = \frac{(8C_2 - 2)}{11}, \gamma = \frac{(30C_2 - 2)}{55}, \quad (21)$$

The transport equation for the rate of dissipation of turbulent kinetic energy of Lai and So's model (1990) is used without any modification and is given by



$$\frac{D\varepsilon}{Dt} = \frac{\partial}{\partial x_k} \left( \bar{\mu} \frac{\partial \varepsilon}{\partial x_k} \right) + \frac{\partial}{\partial x_k} \left( C_s \frac{k}{\varepsilon} u_k u_i \frac{\partial \varepsilon}{\partial x_i} \right) + C_{\varepsilon_1} (1 + \sigma f_{w,2}) \frac{\varepsilon}{k} \tilde{P} - C_{\varepsilon_2} f_\varepsilon \frac{\varepsilon \tilde{\varepsilon}}{k} + f_{w,2} \left[ \left( \frac{7}{9} C_{\varepsilon_2} - 2 \right) \frac{\varepsilon \tilde{\varepsilon}}{k} - \frac{1}{2k} \left( \varepsilon - \frac{2\bar{\mu}k}{\rho x_2^2} \right)^2 \right], \quad (22)$$

with

$$\tilde{\varepsilon} = \varepsilon - 2 \frac{\bar{\mu}}{\rho} \left( \frac{\partial k^{1/2}}{\partial x_2} \right)^2, \quad (23)$$

Finally, the various parameters constants and damping functions are given in the next table.

Table 1. Constants and damping functions of the Lai and So (1990) Reynolds stress model

Constants				
$C_1 = 1.5$	$C_2 = 0.4$	$C_{\varepsilon_1} = 1.35$	$C_{\varepsilon_2} = 1.8$	$C_s = 0.11$
$\alpha^* = 0.45$	$C_\varepsilon = 0.15$	$C_{V1} = -1.8$	$C_{V2} = 0.2$	$C_0 = -0.95$
Damping functions				
$f_{w,2} = \exp \left[ - \left( \frac{R_T}{64} \right)^2 \right]$		$f_\varepsilon = 1 - \left( \frac{2}{9} \right) \exp \left[ - \left( \frac{R_T}{6} \right)^2 \right]$		

The turbulent Reynolds number,  $R_T$ , is defined as

$$R_T = \rho k^2 / \bar{\mu} \varepsilon, \quad (24)$$

The damping function  $f_{w,1}$  of Lai and So (1990) was modified to account for viscoelastic effects and is given by

$$f_{w,1} = \exp \left[ - (f_\mu)^{0.86} \right], \quad (25)$$

where  $f_\mu$  is the original damping function derived by Cruz and Pinho (2003) in the context of their  $k$ - $\varepsilon$  model.

$$f_\mu = \left\{ 1 - \left[ 1 + \left| \frac{1-n}{1+n} \right| x_2^* \right]^{-|1+n|/A^+} \right\} \times \left\{ 1 - \left[ 1 + \left| \frac{p-1}{3-p} \right| x_2^* C^{\frac{1-p}{2-p}} \right]^{-|3-p|/A^+} \right\}, \quad (26)$$

This function accounts for wall effects very much as Van Driest's function do for Newtonian fluids and is influenced by the rheological properties of the fluids measured by the shear-thinning intensity ( $n < 1$ ) and Trouton ratio thickening ( $p > 1$ ) of the extensional viscosity. The parameters, which were quantified by performing extensive calculations, are identical to  $C=25$  and  $A^+=40$ , and  $x_2^*$  is a wall coordinate normalised by the wall viscosity ( $x_2^* = u_w x_2 / \nu_w$ ) and using  $u_w = (\nu_w \cdot \varepsilon_w)^{1/4}$ .

#### 4. Results and discussion

The program used to carry out the numerical simulations is based on a finite-volume discretization and the TDMA solver is used to calculate the solution of the discretized algebraic governing equations. The mesh is non-uniform with 199 cells across the pipe, giving mesh-independent results for Newtonian and non-Newtonian fluids within 0.1%. The full domain in the transverse direction is mapped, hence only the following wall boundary conditions need to be imposed:

$$U_1 = \overline{u_1 u_1} = \overline{u_2 u_2} = \overline{u_3 u_3} = \overline{u_1 u_2} = 0 \quad \text{and} \quad \varepsilon = 2 \frac{\bar{\mu}}{\rho} \left( \frac{\partial k^{1/2}}{\partial x_2} \right)^2 \quad \text{at} \quad r = R$$

The turbulence model was calibrated using the experimental data from Escudier *et al.* (1999) for their 0.125% PAA aqueous solution, following the philosophy of Cruz and Pinho (2003), Cruz *et al.* (2004) and Resende *et al.* (2006). Then, the model is tested for the remaining viscoelastic fluids, 0.2% XG, 0.25% CMC and 0.09% / 0.09% XG / CMC without any change to the turbulence model.

#### 4.1. Newtonian fluids

The modification of the damping function  $f_{w,1}$  can severely affect predictions for Newtonian fluids, so here we assess that effect. The modification of  $f_{w,1}$  was carried out to bring into the model the capability to predict flows of viscoelastic fluids, but the changes relative to the predictions with the original formulation of Lai and So (1990) were minimized. For Newtonian fluids the error in the Darcy friction coefficient for fully-developed pipe turbulent flows, for a Reynolds numbers of 7430, is negligible, relative to the predictions with the original model of Lai and So (1990). This can be observed in the comparison between the mean velocity profiles in wall coordinates in Figure 1. At larger Reynolds numbers, the differences relative to the predictions of the original Lai and So model are also negligible.

The corresponding profiles of the normalized turbulent kinetic energy and Reynolds normal stresses can be observed in the Figure 2. Comparing the results of the present model with those model of Lai and So (1990) we can see a small deterioration of the turbulent kinetic energy predictions away from the wall and consequently a small decreased of quality of the normalised Reynolds normal stresses distribution, defined in Eq. (27),

$$u_1^{*+} = \sqrt{\overline{u_1^2}} / u_\tau ; u_2^{*+} = \sqrt{\overline{u_2^2}} / u_\tau ; u_3^{*+} = \sqrt{\overline{u_3^2}} / u_\tau , \quad (27)$$

where  $u_\tau$  is the friction velocity. The spatial coordinate  $x_2^+$  is normalized with the wall viscosity and the friction velocity ( $x_2^+ = u_\tau x_2 / \nu_w$ ).

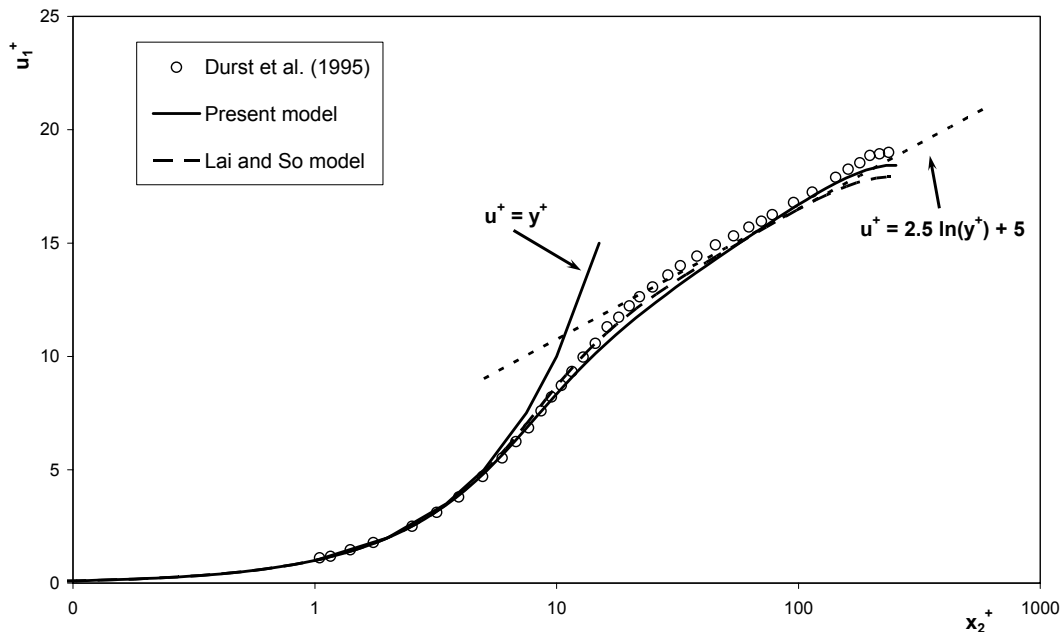


Figure 1. Comparison between the predicted and the measured mean velocity profile for fully-developed turbulent pipe flow of Newtonian fluid at Re=7430 in wall coordinates.

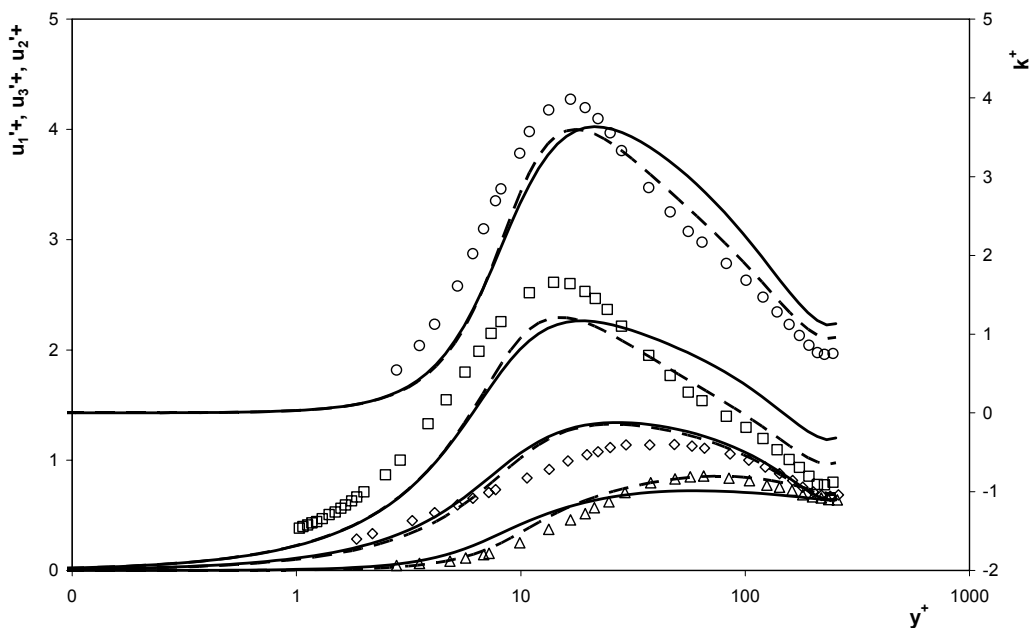


Figure 2. Comparison between the predicted and the measured profiles of normalized turbulent kinetic energy and Reynolds normal stresses for fully-developed turbulent pipe flow of Newtonian fluid at  $Re=7430$  in wall coordinates:  $\circ$   $k^+$ ,  $\square$   $u_1'^2$ ,  $\diamond$   $u_3'^2$ ,  $\Delta$   $u_2'^2$  data of Durst *et al.* (1995); — Present model; - - Lai and So model (1990).

#### 4.2. Non-Newtonian fluids

We compare predictions with the experimental data for turbulent fully-developed pipe flow of Escudier *et al.* (1999) and Resende *et al.* (2006). The various non-Newtonian terms in the momentum and Reynolds stress equation have an impact at different locations in the flow. The pseudo-elastic stress in the momentum equation is basically relevant in the buffer layer, but this is sufficient to affect the flow across the whole pipe and is especially important to create drag reduction. Indeed, and in contrast to the  $k - \varepsilon$  model of Cruz *et al.* (2004), where the drag reduction was basically achieved by a reduction of the eddy viscosity, this Reynolds stress model is truer to the real behaviour of polymer solutions because the drag reduction is achieved by the increasing importance of this new stress as it should according to DNS simulations that show the drag reduction being achieved by the increasing role of the polymer stress ( $\tau_p$ ). In this Reynolds stress model, the pseudo-elastic stress values are larger than in the model of Cruz *et al.* [2], but they still have a negative sign. This negative sign is not a deficiency of the model because the polymer contribution to the total extra stress equals the sum of the pseudo-elastic stress with part of the molecular shear stress, i.e.  $\tau_p = 2\bar{\mu}S_{xy} + 2\bar{\mu}'s_{xy} - 2\mu_s S_{xy}$  where  $\mu_s$  is the solvent viscosity (in the present case water). Therefore, the polymer shear stress remains positive, increases with drag reduction, as it should, and when added to the positive Reynolds shear stress and positive solvent shear stress the total equals the linear stress variation across the pipe. This is so even though the deduction of the model for the pseudo-elastic stress was based in the same philosophy of Cruz *et al.* (2004).

As the pseudo-elastic stress increases with drag reduction there is also a small increase of  $k^+$ , an improvement over the  $k - \varepsilon$  closure developed by Cruz *et al.* (2004). Finally, and in contrast to the  $k - \varepsilon$  model where the pseudo-elastic stress helped to improve the predictions, but was not essential to obtain drag reduction, in the Reynolds stress closure its incorporation in the balance of momentum is essential to obtain drag reduction, and the new non-Newtonian terms in the transport equation are also required.

##### 4.2.1. Measured polymer solutions

The mean velocity profile in wall coordinates for the 0.125% PAA solution at  $Re=42900$  can be observed in Figure 3. Comparing with the previous model of Cruz *et al.* (2004) there is a better behaviour of the mean velocity profile in spite of both models predict well the Darcy friction factor.

The profiles of turbulent kinetic energy and of the Reynolds normal stresses in Figure 4 show that  $k$  and  $\overline{u_1'^2}$  are underpredicted near to the wall, especially in the region of the peak stress. The prediction of  $\overline{u_3'^2}$  is good, but there is also an underprediction of  $\overline{u_2'^2}$ .

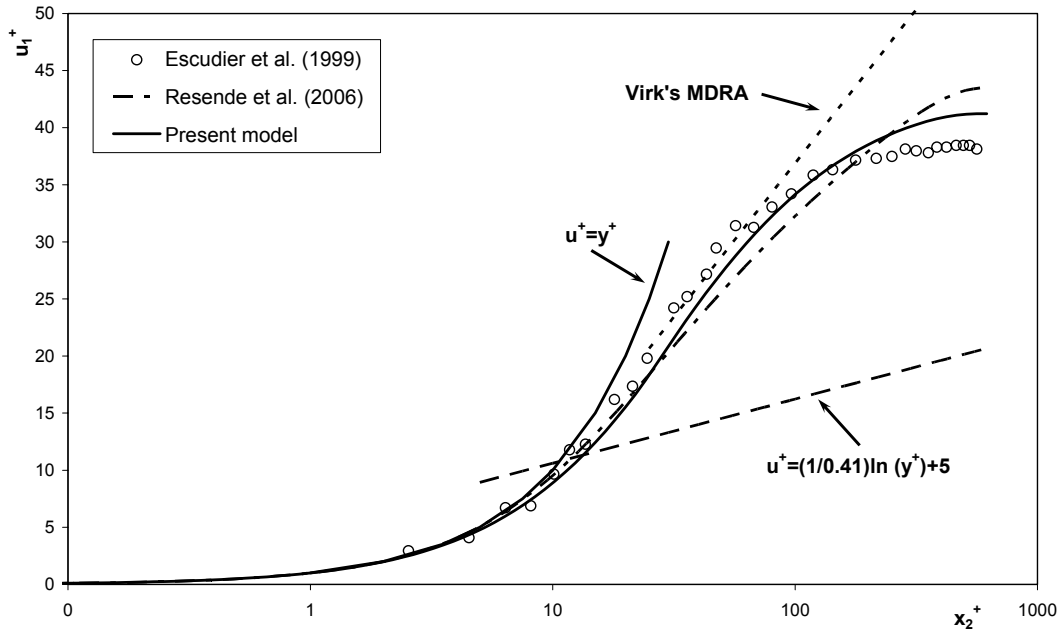


Figure 3. Comparison between the predicted and measured mean velocity profile for fully-developed pipe turbulent flow with the 0.125% PAA solution at  $Re=42900$  in wall coordinates.

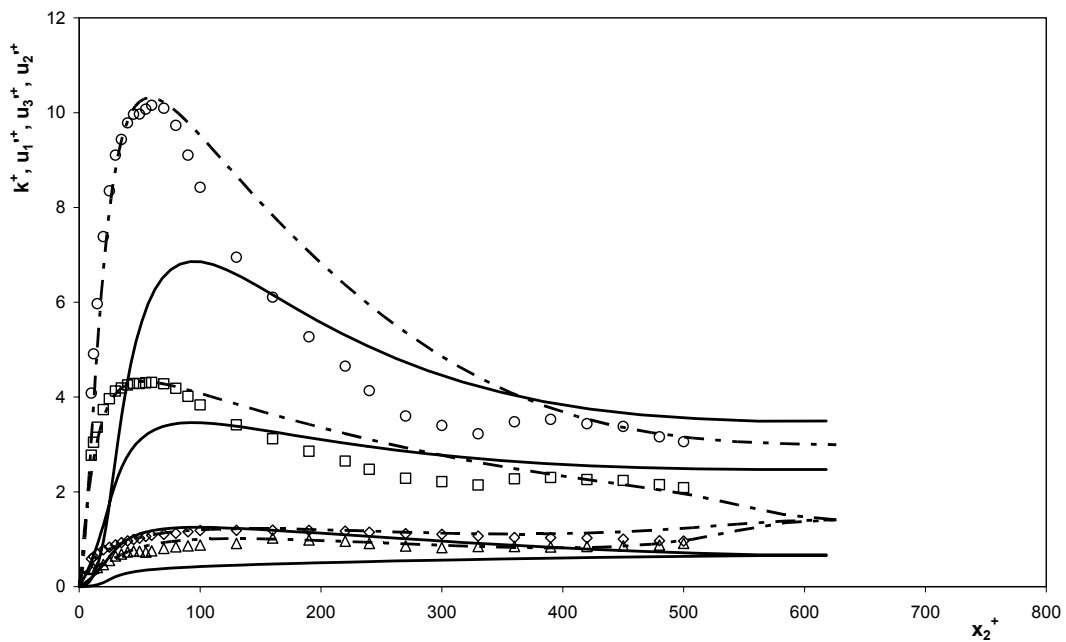


Figure 4. Comparison between the predicted and the measured profiles of normalized turbulent kinetic energy and Reynolds normal stresses for fully-developed turbulent pipe flow of 0.125% PAA fluid at  $Re=42900$  in wall coordinates:  $\circ$   $k^+$  data of Escudier *et al.* (1999);  $\square$   $u_1^+$ ,  $\diamond$   $u_3^+$ ,  $\Delta$   $u_2^+$  data of Resende *et al.* (2006); — Present model; - - - Resende *et al.* (2006).

For the 0.25% CMC fluid the predictions of the mean velocity profile and turbulent quantities at  $Re=16600$  are presented in Figure 5 and Figure 6, respectively. The mean velocity profile shows a good agreement with the experiments. In terms of the turbulent quantities, these are well predicted in terms of magnitude, but the peak axial normal stress and  $k$  are shifted to higher values of  $x_2^+$ .

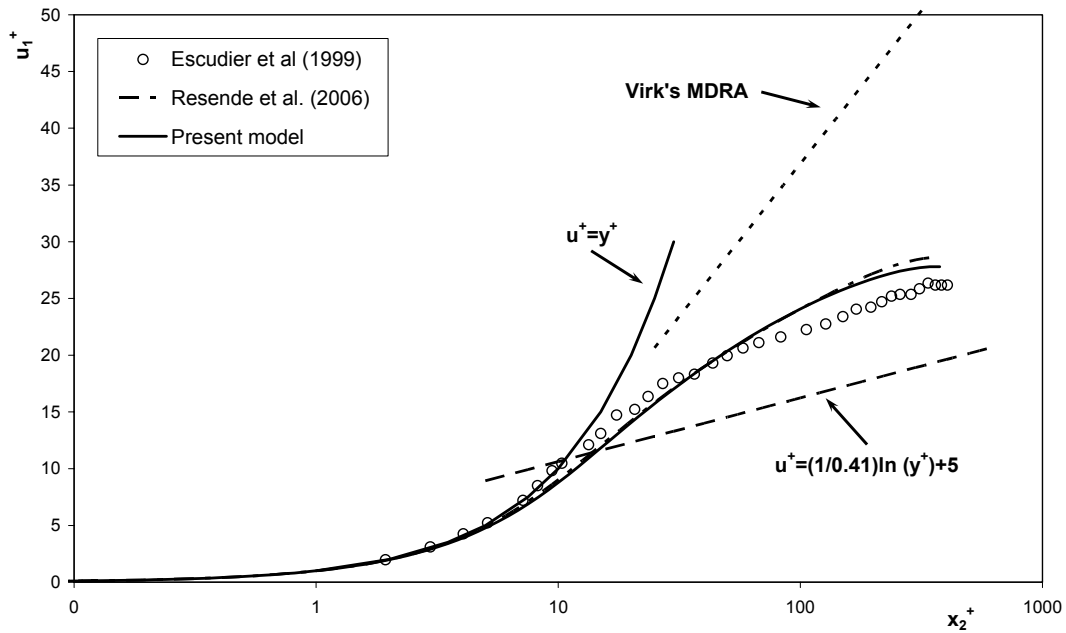


Figure 5. Comparison between the predicted and measured mean velocity profile for fully-developed pipe turbulent flow with the 0.25% CMC solution at  $Re=16600$  in wall coordinates.

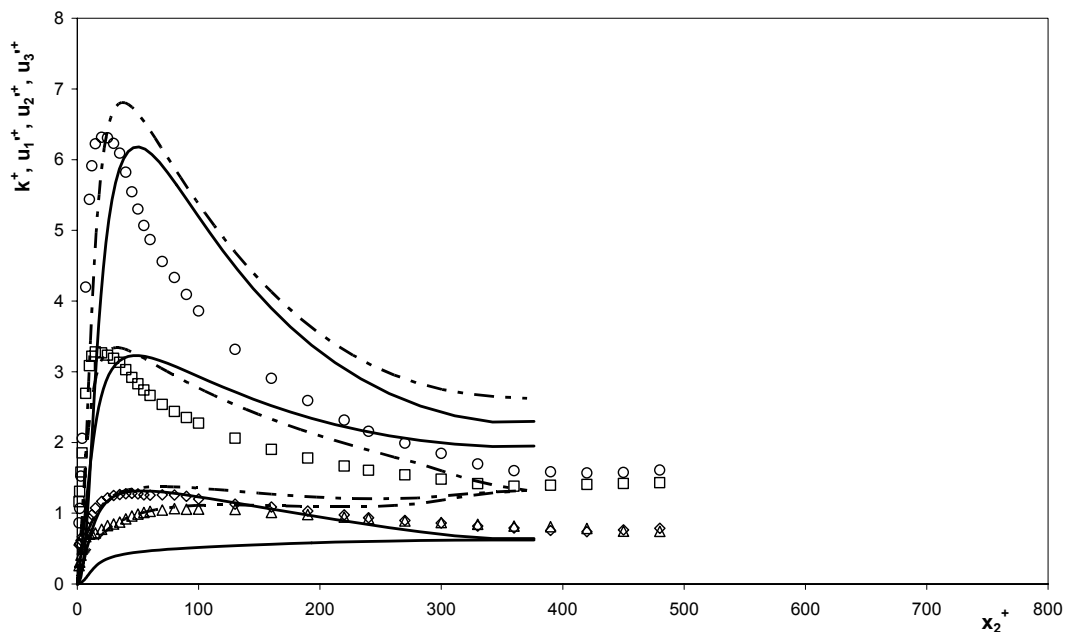


Figure 6. Comparison between the predicted and the measured profiles of normalized turbulent kinetic energy and Reynolds normal stresses for fully-developed turbulent pipe flow of 0.25% CMC fluid at  $Re=16600$  in wall coordinates:  $\circ$   $k^+$  data of Escudier *et al.* (1999);  $\square$   $u_1^{+}$ ,  $\diamond$   $u_3^{+}$ ,  $\Delta$   $u_2^{+}$  data of Resende *et al.* (2006); — Present model; — - Resende *et al.* (2006).

The prediction of the mean velocity profile for the blend (0.09% / 0.09% CMC / XG) solution, at  $Re=45200$ , match the experimental data, as seen in Figure 7. For the 0.2% XG solution, at  $Re=45200$ , exist a small deficit of the mean velocity profile, Figure 8. Comparing with the previous model of Cruz *et al.* (2004) there were significant improves, especially with the 0.2% XG fluids.

It must be emphasised at this stage that the predictions for these two fluids, and in particular for the 0.2% XG solution, are significantly better than was previously achieved by any of the first-order closures developed in the past for viscoelastic fluids (2003; 2004; 2006), an important success of the current Reynolds stress model.

Figure 9 for the blend and Figure 10 for the 0.2% XG show that, as for the previous two non-Newtonian fluids, the axial and radial Reynolds normal stresses and  $k$  are underpredicted near the wall, with the peak of the axial normal stress and  $k$  shifted to higher values of  $x_2^+$ . For the 0.2% XG solution the tangential Reynolds normal stress is slightly over-predicted.

So, and as a general conclusion regarding predictions of the turbulent quantities, there is in almost all cases an underprediction in  $k$ ,  $\overline{u_1^2}$  and  $\overline{u_2^2}$  near the wall, whereas  $\overline{u_3^2}$  is usually well predicted.

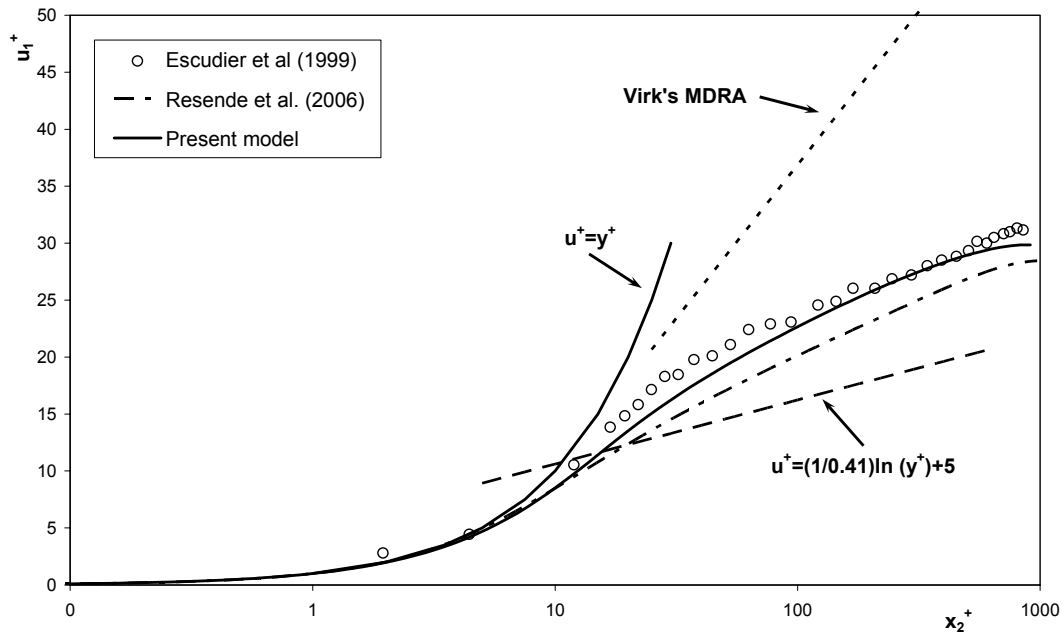


Figure 7. Comparison between the predicted and measured mean velocity profile for fully-developed pipe turbulent flow with the 0.09% / 0.09% CMC / XG solution at Re=45300 in wall coordinates.

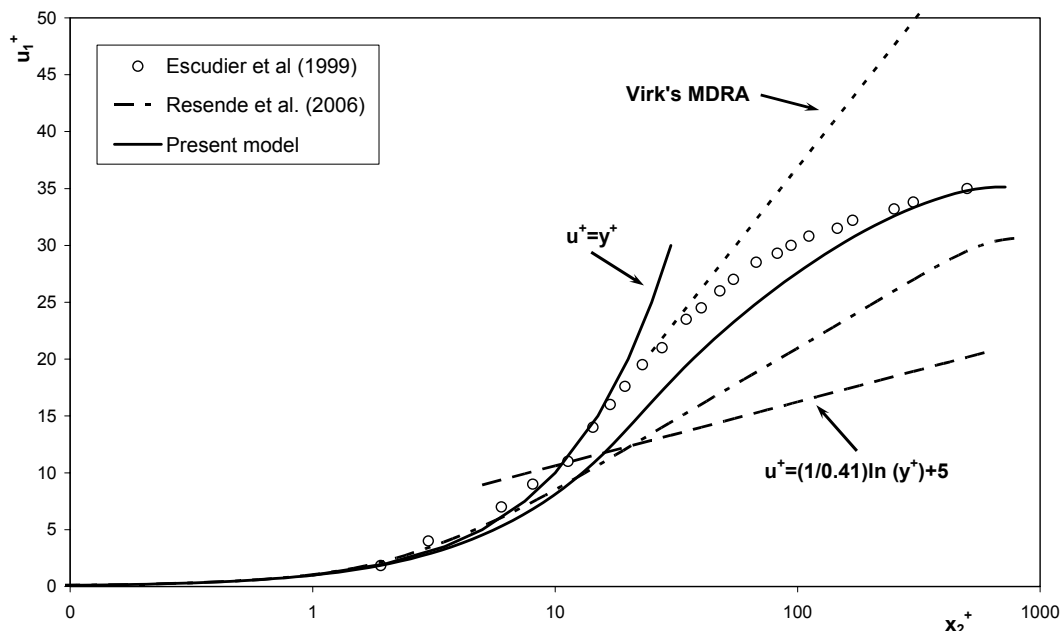


Figure 8. Comparison between the predicted and measured mean velocity profile for fully-developed pipe turbulent flow with the 0.2% XG solution at Re=3900 in wall coordinates.

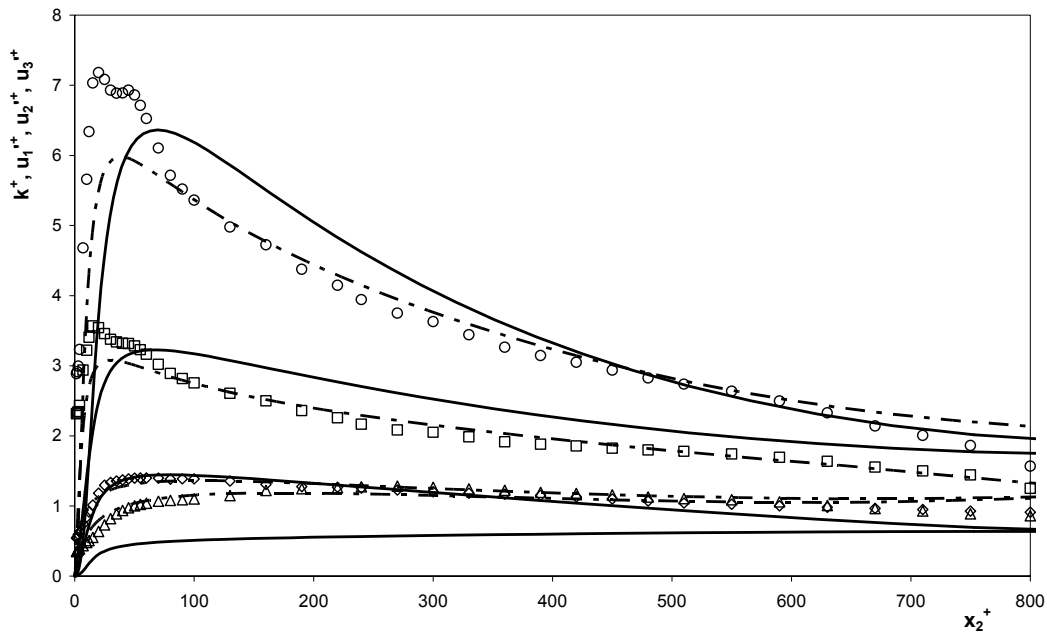


Figure 9. Comparison between the predicted and the measured profiles of normalized turbulent kinetic energy and Reynolds normal stresses for fully-developed turbulent pipe flow of 0.09% / 0.09% CMC / XG fluid at  $Re=45300$  in wall coordinates:  $\circ$   $k^+$  data of Escudier *et al.* (1999);  $\square$   $u_1^+$ ,  $\diamond$   $u_3^+$ ,  $\Delta$   $u_2^+$  data of Resende *et al.* (2006); — Present model; - - Resende *et al.* (2006).

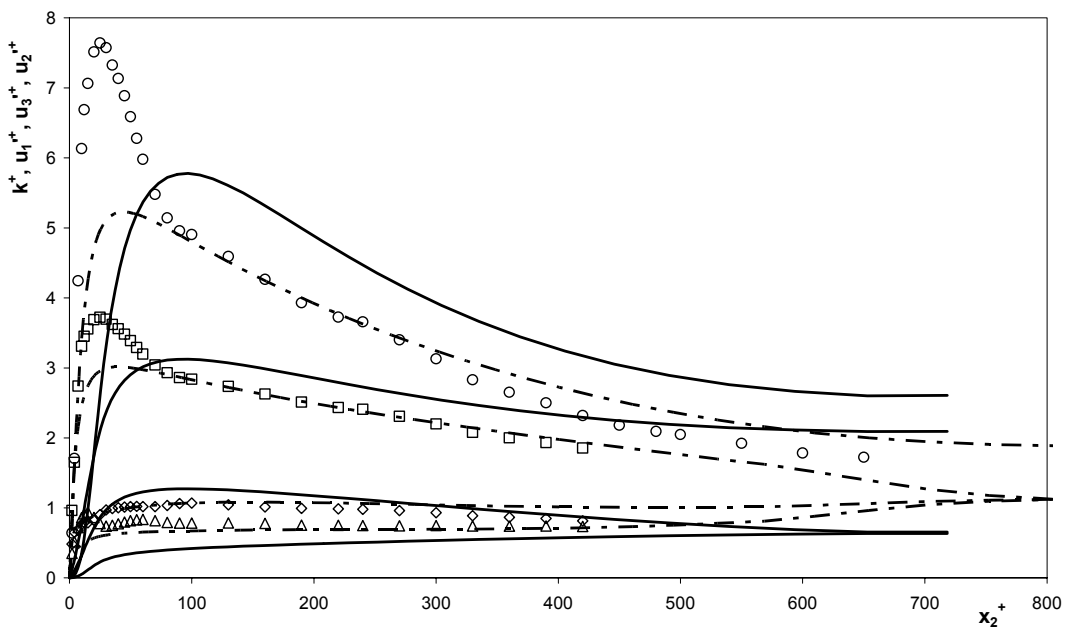


Figure 10. Comparison between the predicted and the measured profiles of normalized turbulent kinetic energy and Reynolds normal stresses for fully-developed turbulent pipe flow of 0.20% XG fluid at  $Re=39000$  in wall coordinates:  $\circ$   $k^+$  data of Escudier *et al.* (1999);  $\square$   $u_1^+$ ,  $\diamond$   $u_3^+$ ,  $\Delta$   $u_2^+$  data of Resende *et al.* (2006); — Present model; - - Resende *et al.* (2006).

## 5. Conclusions

A Reynolds stress model has been developed to predict the flow of viscoelastic solutions based on a generalised Newtonian constitutive equation modified to account for elastic effects. The Reynolds stress model is a modified version of the Lai and So (1990) low Reynolds turbulence model which includes several new non-Newtonian terms.

Closures for all these new terms were developed as well as for the pseudo-elastic stress term appearing in the momentum equation.

The predictions of this model are remarkably good for all fluids tested, four different aqueous solutions of polymer and this was assessed in terms of the friction factor, mean velocity and the three Reynolds normal stresses. The less successful achievement was in predicting these Reynolds stresses were in general the model underpredicted  $\overline{u_1^2}$  near the wall and  $\overline{u_2^2}$ , but it was able to predict well the  $\overline{u_3^2}$  component.

A significant improvement over previous models for viscoelastic fluids are the successful predictions for the two solutions based on the semi-rigid xanthan gum molecule, for which the linear and nonlinear  $k - \varepsilon$  models of Cruz *et al.* (2004) and Resende *et al.* (2006) underpredicted the measured levels of drag reduction.

Therefore, the present model represents a significant improvement over previous turbulence models for viscoelastic solutions, all of which are of first-order.

## 6. Acknowledgements

The authors would like to acknowledge funding of FEDER via projects POCI/56342/EQU/2004 and POCI/59338/EME/2004 of Fundação para a Ciência e Tecnologia.

## 7. References

- Craft, T. J., Launder, B. E. and Suga, K., 1996, "Development and application of a cubic eddy-viscosity model of turbulence", *Int. Journal of Heat and Fluid Flow*, Vol. 17, pp. 108-115.
- Cruz, D. O. A. and Pinho, F. T., 2003, "Turbulent pipe flow predictions with a low Reynolds number  $k-e$  model for drag reducing fluids", *Journal of Non-Newtonian Fluid Mechanics*, Vol. 114, pp. 109-148.
- Cruz, D. O. A., Pinho, F. T. and Resende, P. R., 2004, "Modeling the new stress for improved drag reduction predictions of viscoelastic pipe flow", *Journal of Non-Newtonian Fluid Mechanics*, Vol. 121, pp. 127-141.
- Durst, F., Jovanovic, J. and Sender, J., 1995, "LDA measurements in the near-wall region of turbulent pipe flow", *Journal of Fluid Mechanics*, Vol. 295, pp. 305-335.
- Durst, F. and Rastogi, A. K., 1977, "Calculations of turbulent boundary layer flows with drag reducing polymer additives", *Physics of Fluids*, Vol. 20, N° 12, pp. 1975-1985.
- Escudier, M. P., Presti, F. and Smith, S., 1999, "Drag reduction in the turbulent pipe flow of polymers", *Journal of Non-Newtonian Fluid Mechanics*, Vol. 81, pp. 197-213.
- Hassid, S. and Poreh, M., 1978, "A turbulent energy dissipation model for flows with drag reduction", *Journal of Fluids Engineering*, Vol. 100, pp. 107-112.
- Lai, Y. G. and So, R. M. C., 1990, "On near-wall turbulent flow modelling", *Journal of Fluid Mechanics*, Vol. 221, pp. 641-673.
- Malin, M. R., 1997, "Turbulent pipe flow of power-law fluids", *Int. Commun. Heat Mass Transfer*, Vol. 24, N° 7, pp. 977-988.
- Mizushima, T., Usui, H. and Yoshida, T., 1973, "Turbulent pipe flow of dilute polymer solutions", *Journal Chem. Eng. Japan*, Vol. 7, N° 3, pp. 162-167.
- Park, T. S., Sung, H. J. and Suzuki, K., 2003, "Development of a nonlinear near-wall turbulence model for turbulent flow and heat transfer", *Int. Journal of Heat and Fluid Flow*, Vol. 24, pp. 29-40.
- Patel, V. C., Rodi, W. and Scheuerer, G., 1984, "Turbulence Models for Near-Wall and Low Reynolds: A Review", *AIAA Journal*, Vol. 23, N° 9, pp. 1308-1319.
- Pinho, F. T., 2003, "A GNF framework for turbulent flow models of drag reducing fluids and proposal for a  $k-e$  type closure", *Journal of Non-Newtonian Fluid Mechanics*, Vol. 114, pp. 149-184.
- Politis, S., 1989. *Turbulence modelling on inelastic power-law fluids*. London, UK, Department of Mechanical Engineering, Imperial College of Science, Technology and Medicine.
- Resende, P. R., Escudier, M. P., Presti, F., Pinho, F. T. and Cruz, D. O. A., 2006, "Numerical predictions and measurements of Reynolds normal stresses in turbulent pipe flow of polymers", *Int. Journal of Heat and Fluid Flow*, Vol. 27, pp. 204-219.



## APLICAÇÃO DE ESQUEMAS $W_p$ DE TERCEIRA ORDEM EM ANÁLISES TRANSIENTES PELO MÉTODO DOS ELEMENTOS FINITOS

### Vicente Luiz Scalon

Depto. de Engenharia Mecânica, FEB, UNESP,17033-360, Bauru, SP  
e-mail: scalon@feb.unesp.br

### Alcides Padilha

Depto. de Engenharia Mecânica, FEB, UNESP,17033-360, Bauru, SP  
e-mail: padilha@feb.unesp.br

### Fernando Massucheto Jorge

Depto. de Engenharia Mecânica, FEB, UNESP,17033-360, Bauru, SP  
e-mail: fjorge20@yahoo.com

### Marcos Paulo de Carvalho Oliveira

Depto. de Engenharia Mecânica, FEB, UNESP,17033-360, Bauru, SP  
e-mail: mpcoliveira@feb.unesp.br

**Resumo.** O desenvolvimento recente dos modelos numéricos aliado à grande evolução ocorrida nos processadores desencadearam um processo de desenvolvimento de uma técnica de formulação de problemas alternativa. Desse modo, de maneira totalmente diferente de 20 anos atrás, atualmente é muito mais econômico desenvolver um modelo numérico para estudo de um determinado fenômeno, do que realizar um estudo experimental a respeito do mesmo. Os problemas envolvendo o comportamento transitório de um dado fenômeno, são geralmente, complexos e o uso de intervalos de tempo menores implicam em um aumento significativo no tempo necessários para a obtenção da solução. Por conta disto, neste trabalho será realizada uma análise comparativa dos modelos de primeira, segunda e terceira ordem usando o esquema  $W_p$  para solução por elementos finitos. Neste tipo de implementação, para os sistemas parabólicos, estes são apresentados como sendo compostos por uma série de operadores integrais oriundos da mesma família com o objetivo específico de realizar a integração ao longo do tempo. Ao contrário dos métodos tradicionalmente utilizados, esta família de esquema de solução propõe o uso de uma função peso específica para o tempo na análise transiente. Segundo este esquema é possível obter um sistema de equações através de sucessivas integrações por partes de maneira que a dependência do tempo, passe para a função peso pré-estabelecida. Para verificação dos resultados, foi escolhido um problema de condução transiente unidimensional numa parede plana, do qual se conhece a solução analítica. Os resultados usando o esquema  $W_p$  de terceira ordem foram comparados à solução analítica do problema, e aos esquemas  $W_p$  de primeira e segunda ordem, e o desvio das soluções foram analisados em função do tempo e da malha utilizada.

**Palavras chave:** Análise transiente, Método dos Elementos Finitos, Esquemas transientes  $W_p$ .

### 1. Introdução

No passado, a grande maioria dos trabalhos realizados abordando fenômenos físicos, principalmente nas áreas de engenharia, se baseava na obtenção de dados experimentais que justificassem ou ajudassem a entender o como e o porque de tais fenômenos ocorrerem. O desenvolvimento recente dos modelos numéricos aliado à grande evolução ocorrida nos processadores desencadearam um processo de mudança de mentalidade. Desse modo, de maneira totalmente diferente de 20 anos atrás, atualmente é muito mais econômico desenvolver um modelo numérico para estudo de um determinado fenômeno, do que realizar um estudo experimental a respeito do mesmo. Os problemas envolvendo o comportamento transitório de um dado fenômeno, são geralmente são complexos e o uso de intervalos de tempo menores implicam em um incremento muito significativo no tempo necessário para a obtenção da solução. Neste sentido, um grande número de trabalhos publicados recentemente tem objetivado a melhoria dos modelos numéricos para a solução de problemas físicos, contando, inclusive, com uma série de periódicos dedicados exclusivamente a este assunto.

Além disso, existem outras vantagens do modelo matemático devem ser levadas em consideração como o fato de se tratar de uma técnica não invasiva, na qual não existe nenhum tipo de sensor físico que possa perturbar o fenômeno. Outro fator importante é que os resultados podem ser obtidos em qualquer parte do domínio do problema com bom nível de confiabilidade e, se houver necessidade, pode-se inclusive utilizar um maior refinamento nas áreas de interesse.

Uma série de trabalhos recentes tem abordado este tema e variações. Com relação aos novos esquemas de discretização, Zhong *et al.*, 1996 propuseram a utilização de um incremento na ordem da aproximação no tempo dos tradicionais

esquemas transientes. No trabalho, são tratadas uma série de considerações sobre as equações diferenciais parciais (EDP) e, com base nesta técnica, apresentaram um esquema explícito de elevada ordem baseando-se na solução analítica de equações diferenciais de primeira ordem.

Em outro trabalho, Tamma *et al.*, 1997 apresentaram uma compilação de uma série de metodologias para implementação de esquemas transientes utilizando um método similar ao esquema de resíduos ponderados. Neste trabalho, foram analisadas implementações tanto para esquemas de problemas transientes como para problemas dinâmicos, envolvendo termos de aceleração. O trabalho enfatizou muito a metodologia utilizada para a formulação do problema, além de sugerir possíveis esquemas de implementação. a aplicação de cada uma das metodologias é muito pouco explorada, apenas alguns exemplos simplificados são apresentados para ilustração. Nesta mesma linha, trabalhando com as funções peso para o tempo, cabe ressaltar os avanços recentes introduzidos por Tamma *et al.*, 2002.

Embora já seja conhecida há algum tempo, Smolinshi e Wu, 1998 apresentaram algumas inovações para a técnica conhecida como *subcycling* e sua inclusão em problemas de elementos finitos de primeira ordem. Esta técnica permite que diferentes incrementos de tempo sejam usados em diferentes regiões do mesmo problema, dependendo do tamanho dos elementos utilizados. A aplicação desta técnica só tem sentido em discretizações em que a solução não é incondicionalmente estável. Alguns detalhes sobre a implementação da técnica de *subcycling* para o ANSYS podem ser vistos em ANSYS, 2002.

Utilizando outra técnica, Donea *et al.*, 2000 apresentaram a proposta de um novo esquema de solução transiente em fenômenos que envolvem problemas altamente convectivos. Este esquema se baseia nas aproximações de Padé que podem ser usadas com alguma vantagem sobre as aproximações baseadas em Séries de Taylor. Este tipo de aproximação é bastante utilizada na elaboração de esquemas de solução para problemas parabólicos no tempo usando elementos do tipo  $C^0$ . Foi verificada neste estudo grande similaridade entre a formulação obtida através das funções de aproximação de Padé e aquelas que seriam utilizadas no caso de aplicação de um método de Runge Kutta de mesma ordem.

Recentemente, Tamma *et al.*, 2003a explicaram o desenvolvimento fundamental teórico e o rigor de uma representação generalizada com respeito à estabilidade, precisão e, em Tamma *et al.*, 2003b, muita atenção foi dada às representações práticas dos operadores de segunda ordem, incluindo, também, seus atributos algorítmicos e suas extensões para situações não lineares.

Esta série de trabalhos mostra como o assunto de modelos numéricos para problemas transientes têm ganho espaço na pesquisa recente sobre o assunto. Este trabalho, pretende avançar um pouco mais no tema realizando as implementações de primeira, segunda e terceira ordem usando "família  $W_p$ ", em um problema simples de condução de calor unidimensional transiente. Serão avaliados os esquemas cuja implementação é realizada via inclusão de novas funções pesos apresentadas em Tamma *et al.*, 1997.

Embora se trate de um problema unidimensional o mesmo será solucionado utilizando-se de elementos bidimensionais de quatro nós (biquadráticos) com as condições de contorno adequadas para resolução do problema. Para esta solução foi considerado ainda que no instante inicial,  $\tau = 0$ , o material se encontra a uma temperatura  $\theta_0 = 0$ . Os resultados obtidos são, então, confrontados com a solução analítica do problema para, assim, verificar como cada um dos modelos se comporta. Um esquema do problema e suas condições de contorno pode ser visto na fig.(1).

## 2. Metodologia

Tomando por base a discretização de um problema de condução transitória em geometria bidimensional com um termo fonte associado, a equação geral que rege o problema é dada por:

$$\frac{\partial \theta}{\partial t} - \left( \frac{\partial^2 \theta}{\partial x^2} + \frac{\partial^2 \theta}{\partial y^2} \right) = f \tag{1}$$

que se discretizada em elementos finitos sem qualquer forma de aproximação para a derivada no tempo. O termo fonte  $f$  pode representar qualquer situação com possíveis termos fonte. A solução do problema dependeria ainda das inclusões das condições de contorno em nível de espaço e das condições iniciais do problema ( $\theta_0$ ).

Para este caso, será considerada uma discretização no espaço por resíduos ponderados da equação de condução bidimensional, sem nenhum tipo de termo fonte:

$$\int_{\mathcal{V}} W_e \left[ \frac{\partial \theta}{\partial t} - \left( \frac{\partial^2 \theta}{\partial x^2} + \frac{\partial^2 \theta}{\partial y^2} \right) \right] d\mathcal{V} = 0 \tag{2}$$

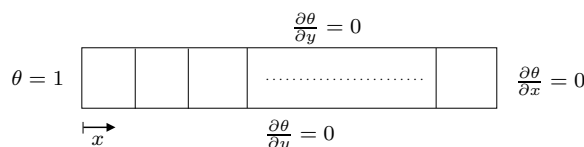


Figura 1: Esquema do problema proposto

Num primeiro momento não haverá preocupação com a discretização do termo transiente, entretanto, considerando-se que o termo está independente da posição, aplicando-se Galerkin, tem-se que:

$$\int_{\mathcal{V}} N_i \frac{\partial \theta}{\partial t} d\mathcal{V} = \int_{\mathcal{V}} N_i d\mathcal{V} \frac{\partial \theta}{\partial t} \quad (3)$$

mas considerando apenas o termo difusivo e aplicando para a sua discretização os procedimentos de redução de ordem (forma fraca), discretização por Galerkin e aproximação nodal, tem-se que:

$$\int_{\mathcal{V}} N_i \left( \frac{\partial^2 \theta}{\partial x^2} + \frac{\partial^2 \theta}{\partial y^2} \right) d\mathcal{V} = \oint_S N_i \frac{\partial \theta}{\partial n} dS - \int_{\mathcal{V}} \left( \frac{\partial N_i}{\partial x} \cdot \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \cdot \frac{\partial N_j}{\partial y} \right) d\mathcal{V} \cdot \theta \quad (4)$$

Assim a equação geral para o problema em resíduos ponderados pode ser escrita na sua forma matricial:

$$[\mathbb{M}] \cdot \frac{\partial \vec{\theta}}{\partial t} + [\mathbb{K}] \cdot \vec{\theta} = [\mathbb{S}] \quad (5)$$

onde:

$$[\mathbb{M}] = \int_{\mathcal{V}} N_i d\mathcal{V}; \quad [\mathbb{K}] = \int_{\mathcal{V}} \left( \frac{\partial N_i}{\partial x} \cdot \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \cdot \frac{\partial N_j}{\partial y} \right) d\mathcal{V}; \quad [\mathbb{S}] = \oint_S N_i \cdot \frac{\partial \vec{\theta}}{\partial n} dS$$

## 2.1. Aplicação do esquema usual de discretização

O esquema de discretização no tempo mais comumente utilizado pelos usuários do método dos elementos finitos foi herdado da técnica de diferenças finitas e consiste em utilizar a série de Taylor para aproximar o comportamento da derivada no tempo. Neste esquema:

$$\frac{\partial \vec{\theta}}{\partial t} = \frac{\vec{\theta}^{m+1} - \vec{\theta}^m}{\Delta t} \quad (6)$$

sendo que os valores de  $m$  e  $m + 1$  representam, respectivamente, as temperaturas no instante atual e no próximo instante de tempo. Desse modo, a equação (5) pode ser escrita da seguinte forma:

$$\frac{1}{\Delta t} [\mathbb{M}] \cdot (\vec{\theta}^{m+1} - \vec{\theta}^m) + [\mathbb{K}] \cdot \vec{\theta}^{m+k} = [\mathbb{S}] \quad (7)$$

sendo que de acordo com o valor de  $k$ , a formulação é realizada de maneira diferente. Para o esquema implícito, a temperatura é avaliada no instante posterior, ou seja,  $k = 1$ .

## 2.2. Implementação por função peso para o tempo ( $W_p$ )

Neste tipo de implementação para os sistemas parabólicos, estes são apresentados como sendo compostos por uma série de operadores integrais da mesma família, com o objetivo específico de realizar a integração ao longo do tempo. Ao contrário dos métodos tradicionalmente utilizados esta família de esquema de solução propõe o uso de uma função peso específica para o tempo na análise transiente. Segundo este esquema, é possível obter um sistema de equações através de sucessivas integrações por parte de maneira que a dependência do tempo, passe para a função peso pré-estabelecida. Talvez, a grande vantagem deste tipo de formulação seja justamente o fato de deixar inalterada a discretização da geometria do problema, ou seja, podem ser mantidas quaisquer tipos de características da discretização no espaço, seja no que tange a funções de aproximação, ou mesmo, esquemas adotados para função peso.

Aplicando as funções pesos anteriormente estabelecidas, na equação (5) e realizando a integração ao longo do tempo, têm-se que:

$$\int_{t_n}^{t_{n+1}} [W]^T ([\mathbb{M}] \cdot \dot{\vec{\theta}} + [\mathbb{K}] \vec{\theta} - [F]) dt = 0 \quad (8)$$

Subdividindo a equação (8) em diversas integrais e integrando o primeiro termo por partes obtém-se:

$$\left( [W]^T [\mathbb{M}] \vec{\theta} \right)_{t_n}^{t_{n+1}} - \int_{t_n}^{t_{n+1}} [W]^T [F] dt = \int_{t_n}^{t_{n+1}} [\dot{W}]^T [\mathbb{M}] \vec{\theta} dt - \int_{t_n}^{t_{n+1}} [W]^T [\mathbb{K}] \vec{\theta} dt \quad (9)$$

ou ainda:

$$\int_{t_n}^{t_{n+1}} ([\dot{W}]^T [\mathbb{M}] - [W]^T [\mathbb{K}]) \vec{\theta} dt = \left( [W]^T [\mathbb{M}] \vec{\theta} \right)_{t_n}^{t_{n+1}} - \int_{t_n}^{t_{n+1}} [W]^T [F] dt \quad (10)$$

### 2.2.1. Aproximação de Primeira Ordem

Fazendo uma aproximação linear para o tempo e o perfil de temperaturas:

$$W_1(\tau) = w_0 + w_1 \cdot \tau \quad (11)$$

$$\vec{\theta}_{n+1} = \vec{\theta}_n + \dot{\vec{\theta}} \cdot \Delta t \cdot \tau \quad (12)$$

Note que neste caso foi necessário realizar uma mudança de variáveis, definindo-se uma variável  $\tau$  de forma a:

$$\tau = \frac{t - t_n}{t_{n+1} - t_n} = \frac{t - t_n}{\Delta t} \quad (13)$$

onde pode-se ainda definir:

$$\mathbb{F} = \int_0^1 (w_0 + w_1 \tau) \cdot [F] d\tau, \quad (14)$$

Em seu trabalho, Tamma *et al.*, 1997, propõe uma expressão geral para obtenção dos coeficientes desta equação de maneira que:

$$W_{1,0} = \int_0^1 (w_0 + w_1 \tau) d\tau = w_0 + \frac{w_1}{2} \quad (15)$$

$$W_{1,1} = \int_0^1 (w_0 + w_1 \tau) \tau d\tau = \frac{w_0}{2} + \frac{w_1}{3} \quad (16)$$

e, desse modo, o valor de  $\dot{\vec{\theta}}$  é dado pela equação:

$$\dot{\vec{\theta}} = \frac{\mathbb{F} - W_{1,0} \cdot [\mathbb{K}] \cdot \vec{\theta}_n}{W_{1,1} \cdot \Delta t \cdot [\mathbb{K}] + W_{1,0} \cdot [\mathbb{M}]} \quad (17)$$

De posse do valor de  $\dot{\vec{\theta}}$  é possível encontrar facilmente os valores do perfil de temperatura no instante  $n + 1$ , usando-se a aproximação representada na equação (12) para  $\tau = 1$ .

É importante notar que, neste caso, o valor do termo fonte  $\mathbb{F}$  não foi descartado pois ele pode ser necessário na aplicação das condições de contorno. De acordo com a natureza do problema, principalmente nos casos envolvendo condições de contorno de 2ª e 3ª espécies, este torna-se fundamental para implementação das condições de superfície.

Tamma *et al.*, 1997 apresentaram uma série de expressões para se obter esquemas usuais de discretização de primeira ordem. Um resumo destes coeficientes estão apresentados na Tabela (1).

Tabela 1: Valores das funções de aproximação para os diversos algoritmos.

Algoritmo	Função de Aproximação
Crank-Nicolson	$W = 1 + 0 \cdot \tau = 1$
Euler Avançado	$W = 1 - 3/2\tau$
Euler Atrasado	$W = 1 - 3\tau$
Liniger	$W = 1 - 17/4\tau$
Galerkin	$W = 1 - \infty\tau$

Obs:  $\tau = (t - t_n)/\Delta t$ ,

### 2.2.2. Aproximação de Segunda Ordem

Fazendo-se uma aproximação quadrática para o tempo e usando os três primeiros termos da série de Taylor para expressar o perfil de temperaturas, tem-se:

$$W_2(\tau) = w_0 + w_1 \cdot \tau + w_2 \cdot (\tau)^2 \quad (18)$$

$$\vec{\theta}_{n+1} = \vec{\theta}_n + \dot{\vec{\theta}} \cdot \Delta t \cdot \tau + 1/2 \cdot (\tau)^2 \cdot (\Delta t)^2 \ddot{\vec{\theta}} \quad (19)$$

Usando estas expressões na equação (8):

$$\int_{t_0}^{t_1} [(w_0 + w_1 \cdot \tau + w_2 \cdot (\tau)^2) [\mathbb{K} \cdot \vec{\theta}_n + (\mathbb{M} + \mathbb{K} \cdot \Delta t) \cdot \dot{\vec{\theta}} + (\mathbb{M} + 1/2 \cdot \Delta t) \cdot \Delta t \cdot \ddot{\vec{\theta}}] - [F]] \cdot dt = 0 \quad (20)$$

Resolvendo a integração, nota-se que seriam necessários os seguintes termos:

$$W_{0,2} = \int_{t_0}^{t_1} (w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2) \cdot d\tau = \left[ w_0\tau + \frac{w_1\tau^2}{2} + \frac{w_2\tau^3}{3} \right]_0^1 = w_0 + \frac{w_1}{2} + \frac{w_2}{3} \quad (21)$$

$$W_{2,1} = \int_{t_0}^{t_1} (w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2) \cdot \tau \cdot d\tau = \left[ \frac{w_0\tau^2}{2} + \frac{w_1\tau^3}{3} + \frac{w_2\tau^4}{4} \right]_0^1 = \frac{w_0}{2} + \frac{w_1}{3} + \frac{w_2}{4} \quad (22)$$

$$W_{2,2} = \int_{t_0}^{t_1} (w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2) \cdot \tau^2 \cdot d\tau = \left[ \frac{w_0\tau^3}{3} + \frac{w_1\tau^4}{4} + \frac{w_2\tau^5}{5} \right]_0^1 = \frac{w_0}{3} + \frac{w_1}{4} + \frac{w_2}{5} \quad (23)$$

$$\mathbb{F}_2 = \int_0^{\Delta t} (w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2) \cdot [F] \cdot dt \quad (24)$$

Desta forma, a equação geral apresentada em (20) passa a ser expressa na forma rearranjada para se obter a solução da segunda derivada:

$$\ddot{\theta} = \frac{\mathbb{F}_2 - W_{2,0} \cdot \mathbb{K} \cdot \vec{\theta}_n - (W_{2,0} \cdot \mathbb{M} + W_{2,1} \cdot \mathbb{K} \cdot \Delta t) \cdot \dot{\theta}}{\Delta t \cdot (W_{2,1} \cdot \mathbb{M} + 1/2 \cdot W_{2,2} \cdot \mathbb{K} \cdot \Delta t)} \quad (25)$$

De posse do valor de  $\ddot{\theta}$  é possível encontrar facilmente os valores para a primeira derivada utilizando-se o método de diferenças finitas:

$$\vec{\theta}_{n+1} = \vec{\theta}_n + \dot{\theta} \cdot \Delta t + \ddot{\theta} \cdot \frac{\Delta t^2}{2} \quad (26)$$

$$\dot{\theta}_{n+1} = \dot{\theta} + \ddot{\theta} \cdot \Delta t \quad (27)$$

a partir do qual o procedimento de marcha se repete. Deve-se observar que se faz necessária uma primeira estimativa do valor inicial da primeira derivada  $\dot{\theta}_0$ , uma vez que em geral somente os valores de temperatura inicial são conhecidos.

Da mesma forma que para as expressões de primeira ordem, Tamma *et al.*, 1997 apresentam uma série de expressões para se obter esquemas usuais de discretização de segunda ordem. Um resumo destes coeficientes está apresentado na tabela (2).

### 2.2.3. Aproximação de Terceira Ordem

Fazendo-se uma aproximação de terceiro grau para o tempo e usando os quatro primeiros termos da série de Taylor para expressar o perfil de temperaturas, tem-se:

$$W_3(\tau) = w_0 + w_1 \cdot \tau + w_2 \cdot (\tau)^2 + w_3 \cdot (\tau)^3 \quad (28)$$

$$\vec{\theta}_{n+1} = \vec{\theta}_n + \dot{\theta} \cdot \Delta t + 1/2 \cdot (\tau)^2 \cdot (\Delta t)^2 \ddot{\theta} + 1/6 \cdot (\tau)^3 \cdot (\Delta t)^3 \ddot{\theta}^{(3)} \quad (29)$$

Usando estas expressões na equação (8):

$$\int_{t_0}^{t_1} [(w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2 + w_3 \cdot \tau^3) [\mathbb{K} \cdot \vec{\theta}_n + (\mathbb{M} + \mathbb{K} \cdot \Delta t) \cdot \dot{\theta} + (\mathbb{M} + 1/2 \cdot \mathbb{K} \cdot \Delta t) \cdot \Delta t \cdot \ddot{\theta} + (1/2 \cdot \mathbb{M} + 1/6 \cdot \mathbb{K} \cdot \Delta t) \cdot \Delta t^2 \cdot \ddot{\theta}^{(3)}]] - [F]] \cdot dt = 0 \quad (30)$$

Tabela 2: Valores das funções de aproximação para os diversos algoritmos de 2ª ordem.

Algoritmo	Função de Aproximação
Dupont	$W = 1 - 19/3 \cdot \tau + 20/3 \cdot \tau^2$
Gear	$W = 1 - 36/5 \cdot \tau + 8 \cdot \tau^2$
Lees	$W = 1 - 60/11 \cdot \tau + 60/11 \cdot \tau^2$
Liniger-1	$W = 1 - 146,82 \cdot \tau - 279,091 \cdot \tau^2$
Liniger-2	$W = 1 - 8,88424 \cdot \tau + 10,0961 \cdot \tau^2$
Runge-Kutta	$W = 1 - 8 \cdot \tau - 10 \cdot \tau^2$
Zlamal	$W = 1 - 236/39\tau - 80/13 \cdot \tau^2$

Obs:  $\tau = (t - t_n)/\Delta t$ ,

Resolvendo a integração, nota-se que seriam necessários os seguintes termos:

$$W_{3,0} = \int_{t_0}^{t_1} (w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2 + w_3 \cdot \tau^3) \cdot d\tau = w_0 + \frac{w_1}{2} + \frac{w_2}{3} + \frac{w_3}{4} \quad (31)$$

$$W_{3,1} = \int_{t_0}^{t_1} (w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2 + w_3 \cdot \tau^3) \cdot \tau \cdot d\tau = \frac{w_0}{2} + \frac{w_1}{3} + \frac{w_2}{4} + \frac{w_3}{5} \quad (32)$$

$$W_{3,2} = \int_{t_0}^{t_1} (w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2) \cdot \tau^2 \cdot d\tau = \frac{w_0}{3} + \frac{w_1}{4} + \frac{w_2}{5} + \frac{w_3}{6} \quad (33)$$

$$W_{3,3} = \int_{t_0}^{t_1} (w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2) \cdot \tau^3 \cdot d\tau = \frac{w_0}{4} + \frac{w_1}{5} + \frac{w_2}{6} + \frac{w_3}{7} \quad (34)$$

$$\mathbb{F}_3 = \int_0^{\Delta t} (w_0 + w_1 \cdot \tau + w_2 \cdot \tau^2 + w_3 \cdot \tau^3) \cdot [F] \cdot dt \quad (35)$$

Desta forma, a equação geral apresentada em (30) passa a ser expressa na seguinte forma, sendo rearranjada para se obter a solução da terceira derivada:

$$\vec{\theta}^{(3)} = \frac{F_3 - W_{3,0} \cdot \mathbb{K} \cdot \vec{\theta}_n + (W_{3,0} \cdot \mathbb{M} - W_{3,1} \cdot \mathbb{K} \cdot \Delta t) \cdot \dot{\vec{\theta}} - (W_{3,1} \cdot \mathbb{M} + 1/2 \cdot W_{3,2} \cdot \mathbb{K} \cdot \Delta t) \cdot \Delta t \cdot \ddot{\vec{\theta}}}{(1/2 \cdot W_{3,2} \cdot \mathbb{M} + 1/6 \cdot W_{3,3} \cdot \mathbb{K} \cdot \Delta t) \cdot \Delta t^2} \quad (36)$$

De posse do valor de  $\vec{\theta}^{(3)}$  é possível encontrar facilmente os valores para a primeira e a segunda derivada utilizando-se o método de diferenças finitas:

$$\vec{\theta}_{n+1} = \vec{\theta}_n + \dot{\vec{\theta}} \cdot \Delta t + \ddot{\vec{\theta}} \cdot \frac{\Delta t^2}{2} + \vec{\theta}^{(3)} \cdot \frac{\Delta t^3}{6} \quad (37)$$

$$\dot{\vec{\theta}}_{n+1} = \dot{\vec{\theta}} + \ddot{\vec{\theta}} \cdot \Delta t + \vec{\theta}^{(3)} \cdot \frac{\Delta t^2}{2} \quad (38)$$

$$\ddot{\vec{\theta}}_{n+1} = \ddot{\vec{\theta}} + \vec{\theta}^{(3)} \cdot \Delta t \quad (39)$$

a partir do qual o procedimento de marcha se repete. Deve-se observar que se faz necessária uma primeira estimativa do valores iniciais da primeira e da segunda derivada ( $\dot{\vec{\theta}}_0$  e  $\ddot{\vec{\theta}}_0$ , respectivamente), uma vez que em geral somente os valores de temperatura inicial são conhecidos.

Da mesma forma que para as expressões de primeira ordem, Tamma *et al.*, 1997 apresentam uma série de expressões para se obter esquemas usuais de discretização de terceira ordem. Um resumo destes coeficientes está apresentado na tabela (3).

Tabela 3: Valores das funções de aproximação para os diversos algoritmos.

Algoritmo	Função de Aproximação
Houbolt	$W = 1 - 615/46 \cdot \tau + 825/23 \cdot \tau^2 - 1155/46 \cdot \tau^3$
Wilson- $\theta$	$W = 1 - 1829/137 \cdot \tau + 4925/137 \cdot \tau^2 - 3465/137 \cdot \tau^3$
Bossak-Newmark	$W = 1 - 180/13 \cdot \tau + 495/13 \cdot \tau^2 - 350/13 \cdot \tau^3$
Hilber-Hughes-Taylor	$W = 1 - 44478/3229 \cdot \tau + 120990/3229 \cdot \tau^2 - 85680/3229 \cdot \tau^3$

Obs:  $\tau = (t - t_n)/\Delta t$ ,

### 2.3. Solução Analítica

Segundo Incropera e Dewit, 2003 a solução analítica, fundamental para efeitos de comparação, é composta por uma série de infinitos termos, obtida com base no método da separação de variáveis. Ela pode ser facilmente encontrada em qualquer livro texto de transmissão de calor e é constituída pelas seguintes equações:

$$\theta = \sum_{i=1}^{\infty} C_i \cdot \exp(-\zeta_i^2 \cdot \tau) \cdot \cos(\zeta_i \cdot x) \quad \text{onde} \quad \zeta_i \cdot \tan(\zeta_i) = Bi \quad \text{e} \quad C_i = \frac{4 \cdot \sin(\zeta_i)}{2\zeta_i + \sin(2\zeta_i)} \quad (40)$$

sendo, ainda que para o caso onde a temperatura da parede é especificada a solução da equação (40), deve ser obtida para quando  $Bi \rightarrow \infty$ .

Com todas estas informações, pode-se facilmente plotar a solução analítica do problema e, embora seja impossível usar os infinitos termos, pode-se representar a solução analítica com boa precisão. Para efeito de comparação neste trabalho, foram utilizados os 90 primeiros termos da série.

### 3. Resultados

Com base neste procedimento uma série de resultados para os modelos de primeira a terceira ordem foram obtidos e comparados com a solução analítica. Os resultados obtidos para o caso estudado podem ser vistos nas fig. (2) a (5). Nestas figuras são apresentadas a solução com base em todos os esquemas apresentados anteriormente na metodologia, de todas as ordens para as mesmas condições de malha e incremento de tempo. Desta forma, é possível uma análise comparativa mais clara do comportamento de cada uma das soluções para problema apresentado.

Os resultados para a solução do problema obtidos a partir da implementação de esquemas de primeira ordem podem ser vistos na fig.(2). Nestes gráficos, estão mostrados o desenvolvimento das temperaturas para o ponto  $x^* = 0,5$ , fig.(2a), e o desvio entre cada solução numérica e a solução analítica nos tempos analisados, fig.(2b). A comparação da solução analítica com cada um dos modelos apresentados foi realizada usando valores para  $\Delta x^* = 0,1$  e  $\Delta \tau = 0,1$ . Nota-se claramente através da análise destes gráficos que a qualidade do resultado obtido pode ser melhorada, de forma significativa, em função do esquema a ser utilizado.

Com base nesta análise os métodos que apresentaram melhor precisão do problema estudado: Ímplicito (Euler avançado), Liniger, Galerkin e Crank-Nicholson, de acordo com a fig.(2). Para cada caso, foram estudados o comportamento do desvio em relação à solução analítica em função da variação nos incrementos de tempo e na malha utilizada. Maiores detalhes dos desenvolvimentos deste estudo podem ser encontrados em Scalon *et al.*, 2005b, onde diversos gráficos do comportamento da solução com refinamento da malha foram analisados e discutidos. Entretanto, a conclusão mostra o que pode ser visto neste gráfico de que, para o problema de difusão de calor proposto, o esquema de Crank-Nicholson foi o que apresentou os melhores resultados.

De maneira análoga, a fig.(3) apresenta um comparativo entre os modelos de segunda ordem e a solução analítica para os 90 primeiros termos da série e  $\Delta x^* = 0,1$  e  $\Delta t = 0,1$  e a fig.(4) apresenta um comparativo de desvios entre os modelos numéricos de segunda ordem e a solução analítica. A análise é feita para a evolução transiente do mesmo ponto  $x^* = 0,5$  que o desenvolvido para o esquema de primeira ordem. Uma análise da comparação apresentada mostra que os modelos de Gear e de Dupont foram os que apresentaram os melhores resultados.

De maneira geral, não se percebe nos modelos de segunda ordem menores desvios em relação à solução analítica que os apresentados nos modelos de primeira ordem. Nota-se que a convergência se deu através de um processo oscilatório em torno da solução analítica, o que não foi verificado nos modelos de primeira ordem. Principalmente nos estágios iniciais as técnicas de segunda ordem apresentaram piores resultados que as de primeira ordem e tenderam a melhorar com o desenvolvimento do trabalho. Para esta aproximação, o critério adotado para estimar o valor da primeira derivada e dar partida da solução foi o de utilizar uma malha com  $\Delta x^*$  muito pequeno e tentar avaliar a derivada com maior precisão.

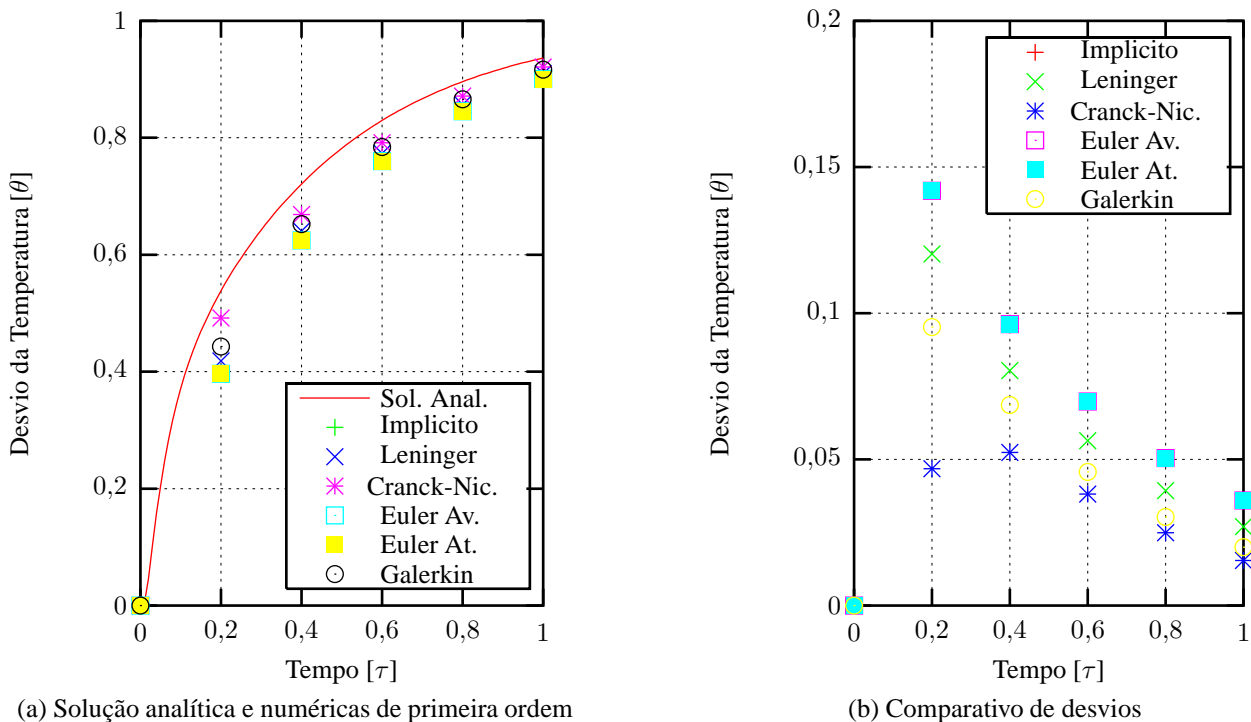


Figura 2: Comparativo entre a solução analítica e numéricas de primeira ordem e seus respectivos desvios.

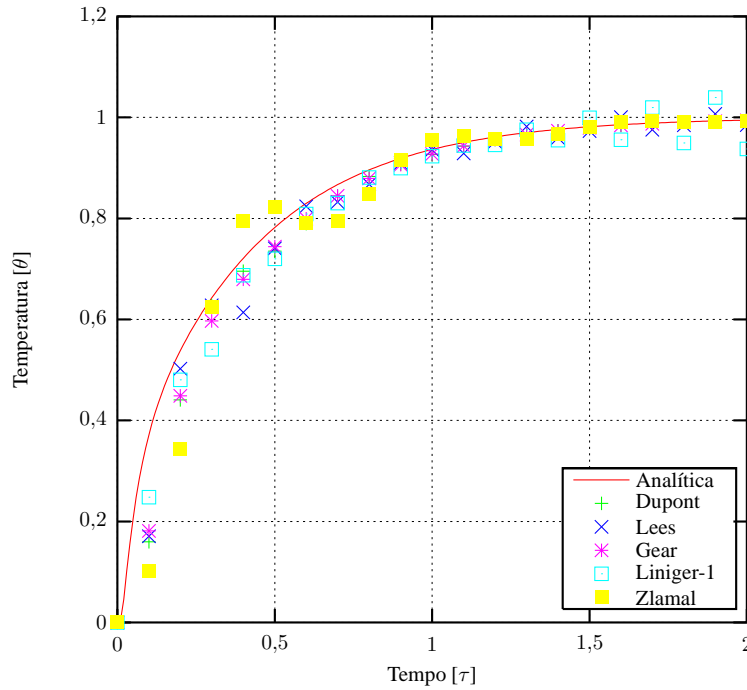


Figura 3: Comparativo entre a solução analítica e numéricas de segunda ordem.

Maiores detalhes sobre o comportamento dos esquemas mais importantes com a malha podem ser encontrados em Scalon *et al.*, 2005a.

O mesmo procedimento adotado para os esquemas de primeira e segundas ordem foi repetido para os de terceira ordem. As condições de análise foram as mesmas,  $\Delta x^* = 0,1$  e  $\Delta t = 0,1$ , assim como o ponto analisado:  $x^* = 0,5$ . Desta forma, a fig.(5) apresenta um comparativo entre os modelos de terceira ordem e a solução analítica e o seu respectivo desvio, fig.(6). Da análise deste gráfico é possível notar que os modelos de Wilson, de Houbolt e de Hilber-Hughes-Taylor apresentaram os melhores resultados. Cabe ressaltar ainda que o esquema de Bossak-Newmark para estas mesmas condições não conseguiu alcançar a convergência da solução. Por conta disto o modelo foi descartado e não aparece nas

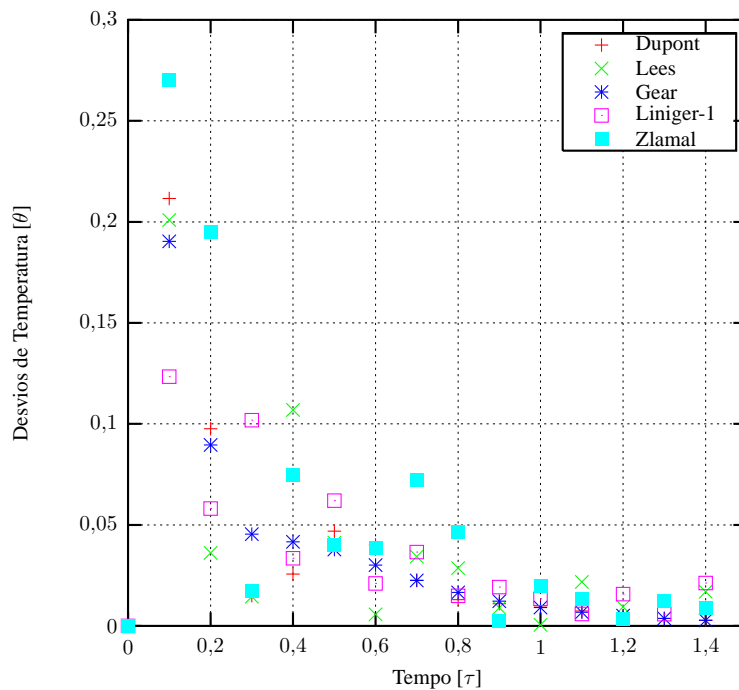


Figura 4: Desvios entre a solução analítica e numéricas de segunda ordem.



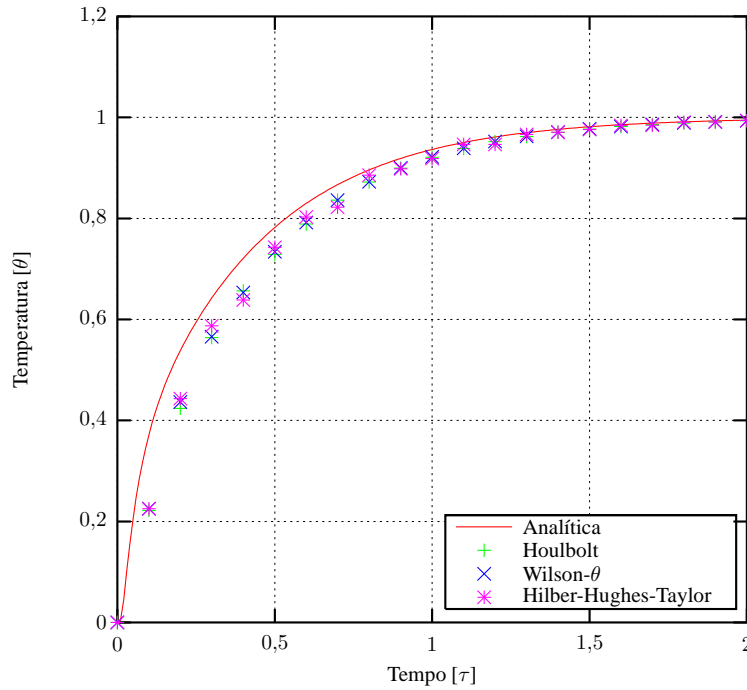


Figura 5: Comparativo entre a solução analítica e numéricas de terceira ordem.

referidas figuras. Outro fato que merece destaque é que, embora tenha convergido, o método de Hilber-Hughes-Taylor apresentou oscilações em torno da solução analítica como pode ser visto claramente na fig.(6).

As fig.(7), (8) e (9) mostram as as implementações de terceira ordem utilizadas e o comportamento dos desvios em função da da malha e incremento de tempo. Todos os esquemas: Houbolt, Wilson-θ e Hilber-Hughes-Taylor, apresentam comportamento similar e portanto a discussão será feita para todos os modelos globalmente. É possível notar, em todos os casos, que o refinamento no tempo, por se tratar de um método de alta ordem, não apresentou variação perceptível na solução obtida para a malha  $\Delta x^* = 0,1$ . Entretanto, em todos os esquemas foi possível notar uma melhora com o

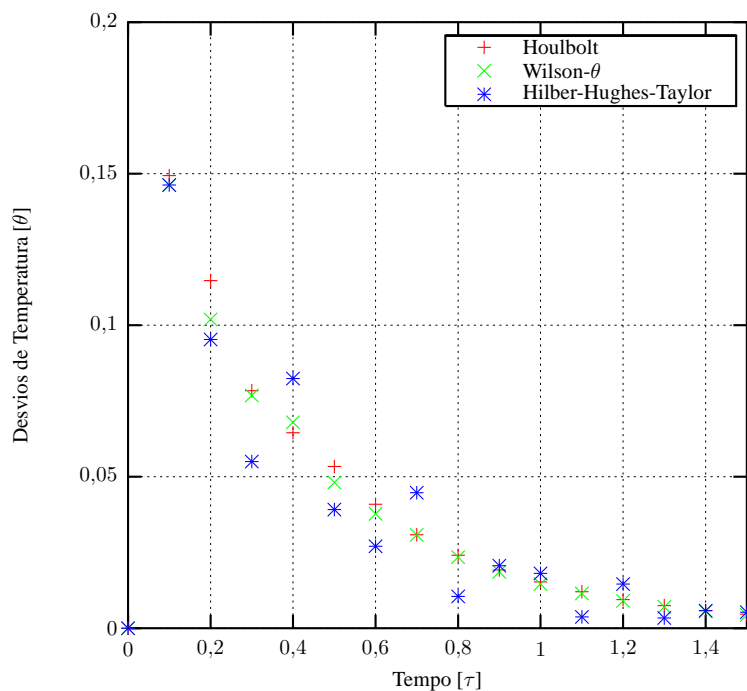


Figura 6: Desvios entre a solução analítica e numéricas de terceira ordem e seus respectivos desvios.

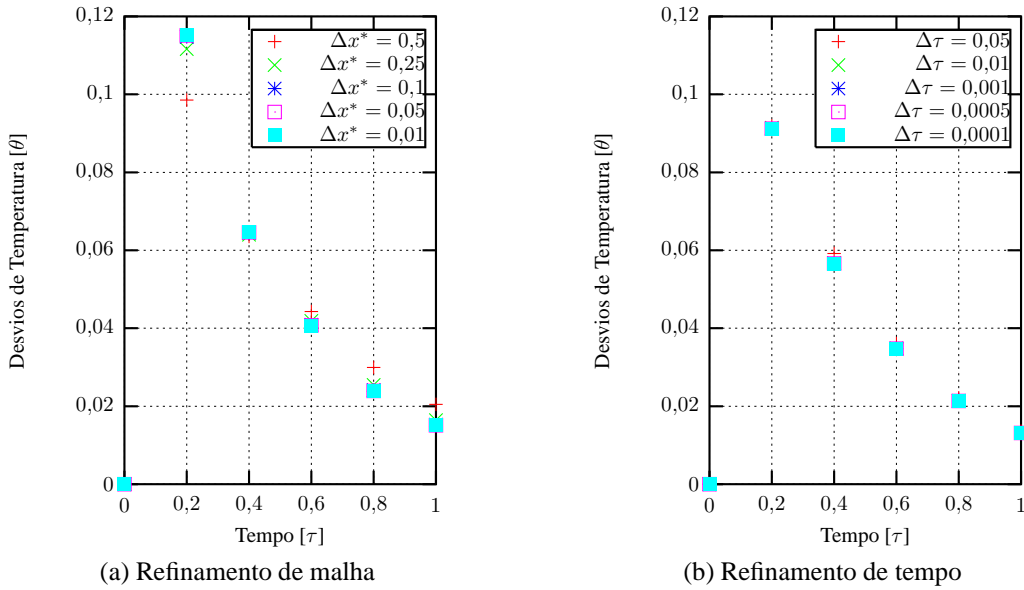


Figura 7: Desvios em relação à solução analítica para o esquema Houbolt.

refinamento da malha em tempos avançados mantendo-se  $\Delta\tau = 0,1$ . Desta forma, nota-se claramente que o refinamento da malha é a única forma de conseguir melhorar a precisão do modelo nas condições estabelecidas.

Com relação às diversas possíveis implementações e o seu comportamento na resolução do problema pode se observar que os métodos alternativos de Houbolt, Wilson- $\theta$  e Hilber-Hughes-Taylor de terceira ordem podem ser boas alternativas para a solução de problemas transitórios, destacando-se pela maior precisão, os dois últimos.

#### 4. Conclusão

Com base nos estudos desenvolvidos neste trabalho, pode-se afirmar que os esquemas baseados na família de funções peso para o tempo  $W_p$ , apresentaram boa concordância com a solução analítica mesmo trabalhando com valores relativamente altos de incremento de tempo.

Esta implementação é, em termos de resultados equivalente à implementação tradicional. Entretanto, pode-se dizer que a implementação  $W_p$  apresenta uma facilidade muito maior de troca do esquema utilizado: bastando para isto que se substitua os coeficientes do polinômio. Pode-se dizer que a grande vantagem deste tipo de formulação seja, justamente, o fato de deixar inalterada a discretização da geometria do problema. Desta forma, podem ser mantidas quaisquer tipos

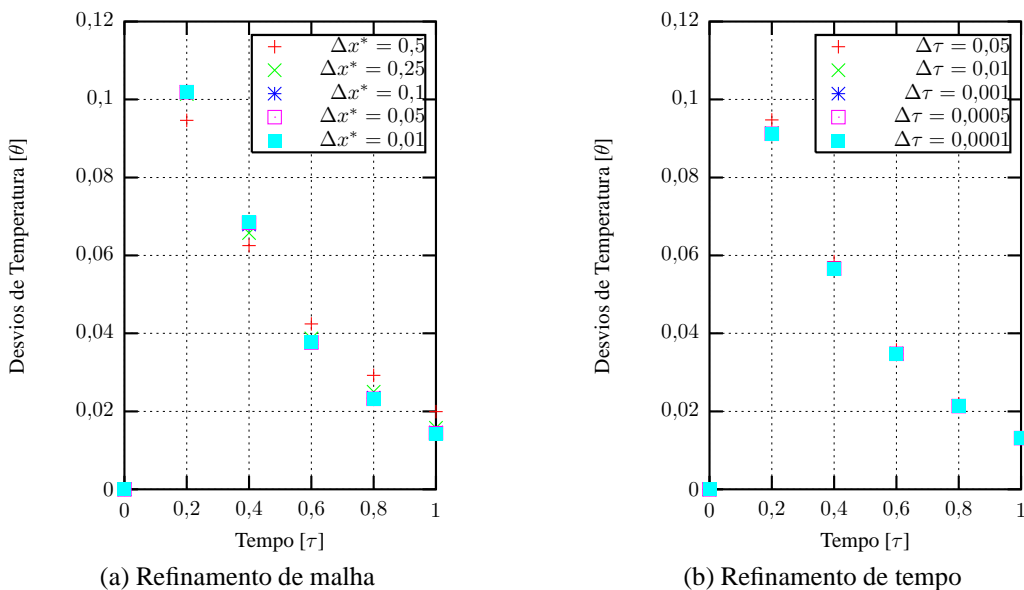


Figura 8: Desvios em relação à solução analítica para o esquema Wilson- $\theta$ .

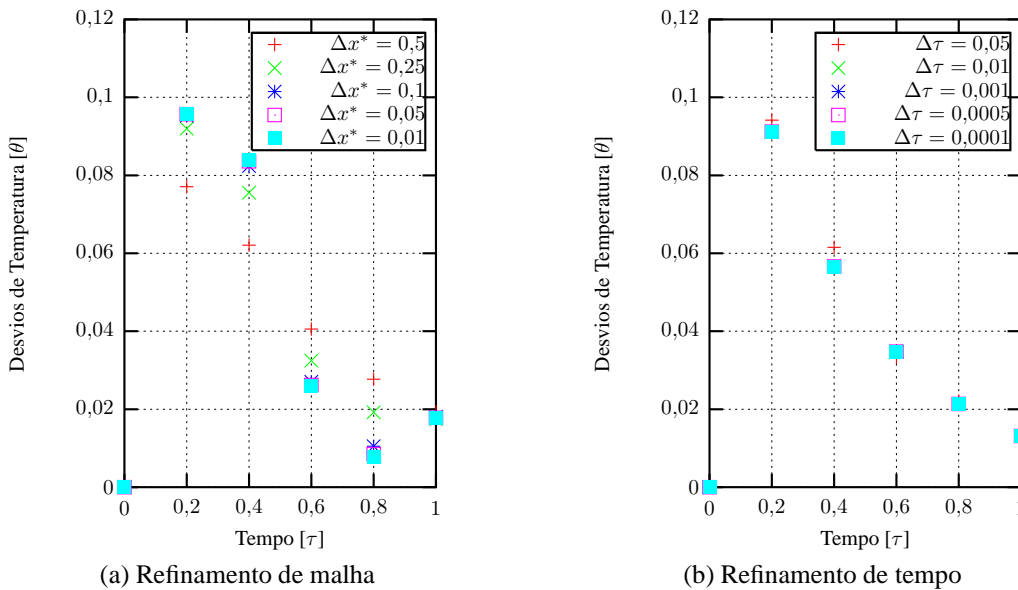


Figura 9: Desvios em relação à solução analítica para o esquema Hilber-Hughes-Taylor.

de características da discretização no espaço, seja no que tange a funções de aproximação, ou mesmo, esquemas adotados para função peso e, com pequenas alterações, mudar a discretização no tempo.

Entretanto, pode-se afirmar com base nos resultados apresentados que o aumento da ordem do esquema no tempo diminui o desvio em relação à solução analítica, principalmente para tempos afastados do instante inicial. Além disto, em função de sua própria natureza, os problemas de terceira ordem apresentaram comportamento praticamente independente dos valores utilizados como incremento de tempo.

Foi possível, ainda, verificar ainda que a utilização de malhas muito refinadas para a estimativa da primeira e segundas derivadas, quando isto é necessário, com esquemas de ordem mais baixa induziu oscilações na solução. Estimativas feitas da mesma forma mas com a malha utilizada na solução apresentaram melhores resultados.

Para finalizar, os resultados mostraram que para o problema proposto, os métodos de terceira ordem de Wilson- $\theta$  e Hilber-Hughes-Taylor apresentaram excelentes resultados na faixa considerada, embora este último tenha apresentado convergência oscilante.

## 5. Agradecimentos

Agradecimentos à Fundação de Amparo à Pesquisa do Estado de São Paulo - FAPESP, que financia a bolsa do aluno Fernando Massucheto Jorge (processo nº 04/13477-0), que muito auxiliou na complementação deste trabalho.

## 6. Referências

- ANSYS, 2002, ANSYS 5.7.1 Help, [http://www.nmt.edu/ansys/ansys5.71/ansyshelp/H1p\\_UI\\_ANSYS\\_Rev.html](http://www.nmt.edu/ansys/ansys5.71/ansyshelp/H1p_UI_ANSYS_Rev.html).
- Donea, J., Roig, B., and Huerta, A., 2000, High-Order Accurate Time-Stepping Schemes For Convection-Diffusion Problems, "Comput. Methods Appl. Mech. Engrg", Vol. 182, pp. 249–275.
- Incropera, F. P. and Dewit, D. P., 2003, "Fundamentos de Transferência de Calor e de Massa", Ed. LTC - Livros Técnicos e Científicos Editora S.A., Rio de Janeiro, Brasil, 5th edition.
- Scalon, V. L., Padilha, A., and Jorge, F., 2005a, Aplicação de Esquemas de Primeira e de Segunda Ordem em Análises Transientes pelo Método dos Elementos Finitos, "Proceedings of XXVI Iberian Latin American Congress of Computational Methods in Engineering - CILAMCE".
- Scalon, V. L., Padilha, A., and Jorge, F., 2005b, Aplicação de Esquemas de Primeira em Análises Transientes pelo Método dos Elementos Finitos, "Proceedings of VI Congresso Temático de Dinâmica, Controle e suas Aplicações - DINCON".
- Smolinshi, P. and Wu, Y., 1998, Stability of Explicit Subcycling Time Integration With Linear Interpolation For First-Order Finite Element Semidiscretizations, "Comput. Methods Appl. Mech. Engrg", Vol. 151, pp. 311–324.
- Tamma, K. K., D.Sha, and Zhou, X., 2003a, Time Discretized Operators. Part I: Towards the Theoretical Design of a New Generation of a Generalized Family of Unconditional Stable Implicit and Explicit Representation of Arbitrary Order, "Comput. Methods Appl. Mech. Engrg", Vol. 192, pp. 257–290.
- Tamma, K. K., D.Sha, and Zhou, X., 2003b, Time Discretized Operators. Part II: Towards the Theoretical Design of a New Generation of a Generalized Family of Unconditional Stable Implicit and Explicit Representation of Arbitrary

Order, “Comput. Methods Appl. Mech. Engrg”, Vol. 192, pp. 291–329.

Tamma, K. K., Zhou, X., and Kanapady, R., 2002, The Time Dimension and a Unified Mathematical Framework for First-Order Parabolic Systems, “Numerical Heat Transfer, Part B”, Vol. 41, pp. 239–262.

Tamma, K. K., Zhou, X., and Valasutean, R., 1997, Computational algorithms for transient analysis: the burden of weight and consequences towards formalizing discrete numerically assigned [DNA] algorithmic markers:  $W_p$ -family, “Comput. Methods Appl. Mech. Engrg.”, Vol. 149, pp. 153–188.

Zhong, W., Zhu, J., and Zhong, X. X., 1996, On a new time integration method for solving time dependant partial differential equations, “Comput. Methods Appl. Mech. Engrg.”, Vol. 130, pp. 163–178.

## **APPLICATION OF THIRD ORDER $W_p$ SCHEMES IN TRANSIENT ANALYSIS USING THE FINITE ELEMENT METHOD**

### **Vicente Luiz Scalon**

Dept. of Mechanical Engineering, FEB, UNESP,17033-360, Bauru, SP  
e-mail: scalon@feb.unesp.br

### **Alcides Padilha**

Dept. of Mechanical Engineering, FEB, UNESP,17033-360, Bauru, SP  
e-mail: padilha@feb.unesp.br

### **Fernando Massucheto Jorge**

Dept. of Mechanical Engineering, FEB, UNESP,17033-360, Bauru, SP  
e-mail: fjorge20@yahoo.com

### **Marcos Paulo de Carvalho Oliveira**

Dept. of Mechanical Engineering, FEB, UNESP,17033-360, Bauru, SP  
e-mail: mpcoliveira@feb.unesp.br

### **Abstract**

*The development of numerical methods associated to the evolution of computers processors leads to a new alternative for physical formulation. So, in a different way of 20 years ago, the numerical model for study physical phenomenon is much more cheap than realize an experimental study. Physical problems with transient analisys are usually complex and the use of lowers time increments leads to a long time to achieve the complete solution. Thus, in this paper a comparative analysis of first, second and third order  $W_p$  schemes for solution by finite elements methods will be done. This implementation for parabolic systems were presented as an integral operators series with the objective to achieve the time integration. In a different way of the tradidional methods, this family schemes uses also a weight function for the time in the transient analysis. In this scheme, it's possible obtain a equation system through integrations by parts. The advantage of this formulation is the maintenance of the problem geometry and any other characteristics of the discretization in space: approximation functions and schemes for the weighting functions. For the results validation, it was chosen a one-dimensional transient conduction problem in a plain wall, which the analytical solution is known. The results using the  $W_p$  scheme of first, second and third orders and the errors of these solutions were analyzed as function of time and mesh.*

**Keywords:** Transient Analysis, Finite Element Method,  $W_p$  Transient Schemes.

## NUMERICAL STUDY OF DISLOCATION MICROSTRUCTURES IN MATERIALS SUBJECTED TO CYCLIC LOADING USING THE WALGRAEF-AIFANTIS MODEL

**Daniel Walgraef**

Center for Nonlinear Phenomena and Complex Systems, CP-231,  
Université Libre de Bruxelles, B-1050, Brussels, Belgium  
dwaelgr@ulb.ac.be, dwalgraef@hotmail.com

**José Pontes**

Metallurgy and Materials Engineering Department, EP/COPPE  
Federal University of Rio de Janeiro, P.O. Box 68505, 21941-972, Rio de Janeiro, RJ, Brazil  
jopontes@metalmat.ufrj.br

**Abstract.** Strain localization and dislocation pattern formation are typical features of plastic deformation in metals and alloys. Glide and climb dislocation motion along with accompanying production/annihilation processes of dislocations lead to the occurrence of instabilities of initially uniform dislocation distributions. These instabilities result into the development of various types of dislocation microstructures, such as dislocation cells, slip and kink bands, persistent slip bands, labyrinth structures, etc., depending on the externally applied loading and the intrinsic lattice constraints. The Walgraef-Aifantis (WA) model is an example of a reaction-diffusion model of coupled nonlinear equations which describe microstructure formation of forest (immobile) and gliding (mobile) dislocation densities in the presence of cyclic loading. This paper briefly discusses two versions of the WA model and focus on a finite differences, second order in time Cranck-Nicholson semi-implicit scheme, with internal iterations at each time step, for solving the model evolution equations in two dimensions..

**keywords:** dislocation patterns, nonlinear dynamics, fatigue

### 1. The Walgraef-Aifantis (WA) model

In the spirit of earlier dislocation models derived for example by Ghoniem *et al.* (1990) for creep, or by Walgraef and Aifantis (1985, 1986, 1997), by Schiller and Walgraef (1988), and by Kratochvil (1979), for dislocation microstructure formation in fatigue, the dislocation population is divided into static dislocations, which may result from work hardening and consist in the nearly immobile dislocations of the “forest”, of subgrains walls or boundaries, etc., and the mobile dislocations which glide between these obstacles. One has thus to derive coupled rate equations for the static and mobile dislocation densities,  $\rho_s$  and  $\rho_m$ . For the sake of simplicity, we shall consider single crystals, oriented for single slip. These dislocations are related to the plastic strain rate via the Orowan relation:

$$\dot{\epsilon}_p = b\rho_m v_g \quad (1)$$

where  $b$  is the length of the Burgers vector,  $\rho_m$  is the total mobile dislocation density and  $v_g$  is the glide velocity in the primary slip band. Time is measured in cycles of loading. Although this relation is, in principle, valid for mean values we will consider that it is also valid locally.

The internal stress is defined by the relation:

$$\sigma_i = \frac{\mu b}{2\pi\lambda} + \xi\mu b\sqrt{\rho_s} \quad (2)$$

where the first contribution comes from obstacles such as precipitates or pre-existing walls separated by an effective spacing  $\lambda$  ( $\mu$  is the shear modulus). The second part is the contribution from the static dislocation population which also opposes dislocation motion. The internal stress reduces the effective stress acting on the dislocations and is defined as:

$$\sigma_e = \sigma_a - \sigma_i \quad (3)$$

where  $\sigma_a$  is the applied stress. Finally, the glide velocity is related to the effective stress via appropriate phenomenological relations expressing the fact that individual dislocation motion is initiated when the effective stress acting on a dislocation exceeds the yield stress. One may consider, for exemple:

$$v_g \propto \sigma_e - \sigma_0 \quad (\sigma_e > \sigma_0) \quad (4)$$

$$v_g = 0 \quad (\sigma_e < \sigma_0) \quad (5)$$

where  $\sigma_0$  is the yield strength.

The essential features of the dislocation dynamics in the plastic regime are, on the one side, their mobility, dominated by plastic flow, but which also includes thermal diffusion and climb, and, on the other side, their mutual interaction process, the more important being:

- Multiplication of static dislocations within the forest;
- Static recovery in the forest via static-static annihilation processes;
- Freeing of static dislocations: when the effective stress increases and exceeds some threshold, it disturbs the local structure of the forest and, in particular, destabilize dislocation clusters which decompose into mobile dislocations. The freeing of forest dislocations occurs with a rate  $\beta$ , which depends on the applied stresses and material parameters;
- Pinning of mobile dislocation by the forest. Effectively, mobile dislocations may be immobilized by the various dislocation clusters forming the forest. The dynamical contribution of such processes is of the form  $G(\rho_s)\rho_m$ , where  $G(\rho_s) = g_n\rho_s^n$  is the pinning rate of a mobile dislocation by a cluster of  $n$  static ones. The Walgraef-Aifantis (WA) model considers  $n = 2$ .

The resulting dynamical system may then be written as:

$$\frac{\partial \rho_s}{\partial t} = D_s \nabla^2 \rho_s + \sigma - v_s d_c \rho_s^2 - \beta \rho_s + \gamma \rho_s^2 \rho_m \quad (6)$$

$$\frac{\partial \rho_m}{\partial t} = D_m \nabla_x^2 \rho_m + \beta \rho_s - \gamma \rho_s^2 \rho_m \quad (7)$$

where  $D_s$  represents the effective diffusion within the forest resulting from the thermal mobility and climb and  $D_m$  represents the effective diffusion resulting from the glide of mobile dislocations between obstacles ( $D_m \gg D_s$ ). The diffusion coefficient  $D_m$  is positive in the present case, but it may become negative, when sweeping or correlation effects are taken into account (Kratochvil and Saxlova), leading to the instability of uniform forest dislocations from the very beginning. The coefficient  $d_c$  is the characteristic length of spontaneous dipole collapse. This length, or at least its order of magnitude, may be, in principle, evaluated from microscopic analysis (Mughrabi, 1979).  $\beta$  is the rate of dislocation freeing from the forest and is associated with the destabilization of dislocation dipoles or clusters under stress. Numerical dislocation dynamics simulations show that in BBC crystals, for exemple, there is a critical value of external applied stresses above which dislocation dipoles become unstable. This value is a decreasing function of the distance between dipole slip lines. If the forest may be considered as an ensemble of dipoles with a mean characteristic width, the threshold stress for destabilization, or freeing,  $\sigma_f$ , could be extracted from such simulatins. More extended numerical analysis could include higher order dislocation clusters and provide the dependence of the threshold stress on the forest dislocation density. The freeing rate is should thus be zero below the freeing threshold, and an increasing function of the applied stress above it. Hence,  $\beta \approx \beta_0(\sigma_a - \sigma_f)^n$  for  $\sigma_a > \sigma_f$ ,  $n$  being a phenomenological parameter.

## 2. The modified WA model: effect of gradient terms

The approximation of mobile dislocation diffusion is controversial and may be lifted. To do so, the mobile dislocation density,  $\rho_m$  is divided into two subfamilies representing the dislocation gliding in the direction of the Burgers vector ( $\rho_m^+$ ) and in the opposite one ( $\rho_m^-$ ), with  $\rho_m = \rho_m^+ + \rho_m^-$ .

For crystals with well-developed forest density, and oriented for single slip, we now write (with  $v_g$  oriented along the  $x$  direction):

$$\frac{\partial \rho_s}{\partial t} = D_s \nabla^2 \rho_s + \sigma - v_s d_c \rho_s^2 - \beta \rho_s + \gamma \rho_s^2 \rho_m \quad (8)$$

$$\frac{\partial \rho_m^+}{\partial t} = -\nabla_x v_g \rho_m^+ + \frac{\beta}{2} \rho_s - \gamma \rho_s^2 \rho_m^+ \quad (9)$$

$$\frac{\partial \rho_m^-}{\partial t} = \nabla_x v_g \rho_m^- + \frac{\beta}{2} \rho_s - \gamma \rho_s^2 \rho_m^- \quad (10)$$

or:

$$\frac{\partial \rho_s}{\partial t} = D_s \nabla^2 \rho_s + \sigma - v_s d_c \rho_s^2 - \beta \rho_s + \gamma \rho_s^2 \rho_m \quad (11)$$

$$\frac{\partial \rho_m}{\partial t} = -\nabla_x v_g \rho_m + \frac{\beta}{2} \rho_s - \gamma \rho_s^2 \rho_m \quad (12)$$

$$\frac{\partial \sigma_m^-}{\partial t} = -\nabla_x v_g \rho_m - \gamma \rho_s^2 \sigma_m \quad (13)$$

where  $\sigma_m = \rho_m^+ - \rho_m^-$  is the density of geometrically necessary dislocations. This variables evolves faster than the other two and may be adiabatically eliminated, leading to the following system, which includes a nonlinear diffusion term in the equation of  $\rho_m$ :

$$\frac{\partial \rho_s}{\partial t} = D_s \nabla^2 \rho_s + \sigma - v_s d_c \rho_s^2 - \beta \rho_s + \gamma \rho_s^2 \rho_m \quad (14)$$

$$\frac{\partial \rho_m}{\partial t} = \nabla_x \frac{v_g}{\gamma \rho_s^2} \nabla_x v_g \rho_m + \beta \rho_s - \gamma \rho_s^2 \rho_m \quad (15)$$

### 3. The numerical scheme for solving the WA model

In order to solve the modified WA model, we use a numerical scheme based on a one proposed by Christov and Pontes (2002). Equations (14) and (15) are solved numerically in two-dimensional rectangular domains, through the finite difference method, using a grid of uniformly spaced points, a second order in time Crank-Nicholson semi-implicit method with internal iterations at each time step, due to the nonlinear nature of the implicit terms. The proposed scheme is splitted in two equations using the Stabilizing Correction scheme (Christov and Pontes, 2002, Yanenko, 1971). The first half-step comprises implicit derivatives with respect to  $x$  and explicit derivatives with respect to  $y$ . In the second half-step, the derivatives with respect to  $y$  are kept implicit and those with respect to  $x$  are explicit. The splitting scheme is shown to be equivalent to the original one.

#### 3.1. The target scheme

The target second order in time, Crank-Nicholson semi-implicit scheme is:

$$\frac{\rho_s^{n+1} - \rho_s^n}{\Delta t} = \Lambda_x^{n+1/2} \frac{\rho_s^{n+1} + \rho_s^n}{2} + \Lambda_y^{n+1/2} \frac{\rho_s^{n+1} + \rho_s^n}{2} + f_1^{n+1/2} \quad (16)$$

$$\frac{\rho_m^{n+1} - \rho_m^n}{\Delta t} = \Lambda_2^{n+1/2} \frac{\rho_m^{n+1} + \rho_m^n}{2} + f_2^{n+1/2} \quad (17)$$

where  $n$  is the number of the time step. Upon including the  $1/2$  factor in the operators  $\Lambda_x^{n+1/2}$ ,  $\Lambda_y^{n+1/2}$  and  $\Lambda_2^{n+1/2}$ , we obtain:

$$\frac{\rho_s^{n+1} - \rho_s^n}{\Delta t} = \Lambda_x^{n+1/2} (\rho_s^{n+1} + \rho_s^n) + \Lambda_y^{n+1/2} (\rho_s^{n+1} + \rho_s^n) + f_1^{n+1/2} \quad (18)$$

$$\frac{\rho_m^{n+1} - \rho_m^n}{\Delta t} = \Lambda_2^{n+1/2} (\rho_m^{n+1} + \rho_m^n) + f_2^{n+1/2} \quad (19)$$

The operators  $\Lambda_x^{n+1/2}$ ,  $\Lambda_y^{n+1/2}$  and  $\Lambda_2^{n+1/2}$  and the functions  $f_1^{n+1/2}$  and  $f_2^{n+1/2}$  are defined as:

$$\Lambda_x^{n+1/2} = \frac{D_s}{2} \frac{\partial^2}{\partial x^2} - \frac{1}{4} v_s d_c \left( \frac{\rho_s^{n+1} + \rho_s^n}{2} \right) - \frac{\beta}{4} \quad (20)$$

$$\Lambda_y^{n+1/2} = \frac{D_s}{2} \frac{\partial^2}{\partial x^2} - \frac{1}{4} v_s d_c \left( \frac{\rho_s^{n+1} + \rho_s^n}{2} \right) - \frac{\beta}{4} \quad (21)$$

$$f_1^{n+1/2} = \sigma + \frac{\gamma}{2} \left( \frac{\rho_s^{n+1} + \rho_s^n}{2} \right)^2 (\rho_m^{n+1} + \rho_m^n) \quad (22)$$

$$\Lambda_2^{n+1/2} = \frac{1}{2} \frac{\partial}{\partial x} \left[ \frac{v_g}{\gamma \left[ (\rho_s^{n+1} + \rho_s^n) / 2 \right]^2} \frac{\partial}{\partial x} v_g \right] - \gamma \left( \frac{\rho_s^{n+1} + \rho_s^n}{2} \right)^2 \quad (23)$$

$$f_2^{n+1/2} = \beta \left( \frac{\rho_s^{n+1} + \rho_s^n}{2} \right) \quad (24)$$

### 3.2. Internal iterations

Since the operators  $\Lambda_x^{n+1/2}$ ,  $\Lambda_y^{n+1/2}$  and  $\Lambda_2^{n+1/2}$ , as well as the functions  $f_1^{n+1/2}$  and  $f_2^{n+1/2}$  contain terms in the new stage, we do internal iterations at each time step, according to:

$$\frac{\rho_s^{n,k+1} - \rho_s^n}{\Delta t} = \Lambda_x^{n+1/2} (\rho_s^{n,k+1} - \rho_s^n) + \Lambda_y^{n+1/2} (\rho_s^{n,k+1} - \rho_s^n) + f_1^{n+1/2} \quad (25)$$

$$\frac{\rho_m^{n,k+1} - \rho_m^n}{\Delta t} = \Lambda_2^{n+1/2} (\rho_m^{n,k+1} \rho_m^n) + f_2^{n+1/2} \quad (26)$$

where the superscript  $(n, k + 1)$  identifies the “new” iteration,  $(n, k)$  and  $n$  stand for the values obtained in the previous iteration and in the previous time step, respectively. The operators  $\Lambda_x^{n+1/2}$ ,  $\Lambda_y^{n+1/2}$ ,  $\Lambda_2^{n+1/2}$  and the functions  $f_1^{n+1/2}$  and  $f_2^{n+1/2}$  are redefined as:

$$\Lambda_x^{n+1/2} = \frac{D_s}{2} \frac{\partial^2}{\partial x^2} - \frac{1}{4} v_s d_c S^{n+1/2} - \frac{\beta}{4} \quad (27)$$

$$\Lambda_y^{n+1/2} = \frac{D_s}{2} \frac{\partial^2}{\partial y^2} - \frac{1}{4} v_s d_c S^{n+1/2} - \frac{\beta}{4} \quad (28)$$

$$f_1^{n+1/2} = \sigma + \frac{\gamma}{2} (S^{n+1/2})^2 (\rho_m^{n,k} + \rho_m^n) \quad (29)$$

$$\Lambda_2^{n+1/2} = \frac{\partial}{\partial x} \left( \frac{v_g}{2\gamma (S^{n+1/2})^2} \frac{\partial}{\partial x} v_g \right) - \gamma (S^{n+1/2})^2 \quad (30)$$

$$f_2^{n+1/2} = \beta S^{n+1/2}, \quad \text{where: } S^{n+1/2} = \frac{\rho_s^{n,k} + \rho_s^n}{2} \quad (31)$$

The iterations proceed until the following criteria is satisfied:

$$\frac{\max\|\rho_s^{n,K+1} - \rho_s^{n,K}\|}{\max\|\rho_s^{n,K}\|} < \delta \quad \text{and} \quad \frac{\max\|\rho_m^{n,K+1} - \rho_m^{n,K}\|}{\max\|\rho_m^{n,K}\|} < \delta$$

in all grid points, for a certain  $K$ . Then the last iteration gives the value of the sought functions in the “new” time step,  $\rho_s^{n+1} \stackrel{\text{def}}{=} \rho_s^{n,K+1}$  et  $\rho_m^{n+1} \stackrel{\text{def}}{=} \rho_m^{n,K+1}$ .

### 3.3. The splitting of the $\rho_s$ equation

The splitting of Eq. (25) is made according to:

$$\frac{\tilde{\rho}_s - \rho_s^n}{\Delta t} = \Lambda_x^{n+1/2} \tilde{\rho}_s + \Lambda_y^{n+1/2} \rho_s^n + f_1^{n+1/2} + (\Lambda_x^{n+1/2} + \Lambda_y^{n+1/2}) \rho_s^n \quad (32)$$

$$\frac{\rho_s^{n,k+1} - \tilde{\rho}_s}{\Delta t} = \Lambda_y^{n+1/2} (\rho_s^{n,k+1} - \rho_s^n) \quad (33)$$

In order to show that the splitting represents the original scheme, we rewrite Eqs. (32) and (33) in the form:

$$(E - \Delta t \Lambda_x^{n+1/2}) \tilde{\rho}_s = (E + \Delta t \Lambda_y^{n+1/2}) \rho_s^n + \Delta t f_1^{n+1/2} + (\Delta t \Lambda_x^{n+1/2} + \Delta t \Lambda_y^{n+1/2}) \rho_s^n \quad (34)$$

$$(E - \Delta t \Lambda_y^{n+1/2}) \rho_s^{n,k+1} = \tilde{\rho}_s - \Delta t \Lambda_y^{n+1/2} \rho_s^n \quad (35)$$

where  $E$  is the unity operator. The intermediate variable  $\tilde{\rho}_s$  is eliminated by applying the operator  $(E - \Delta t \Lambda_x^{n+1/2})$  to the second equation and by summing the result to the first one:

$$(E - \Delta t \Lambda_x^{n+1/2}) (E - \Delta t \Lambda_y^{n+1/2}) \rho_s^{n,k+1} = (E + \Delta t \Lambda_y^{n+1/2}) \rho_s^n - \quad (36)$$

$$(E - \Delta t \Lambda_x^{n+1/2}) \Delta t \Lambda_y^{n+1/2} \rho_s^n + \Delta t f_1^{n+1/2} + (\Delta t \Lambda_x^{n+1/2} + \Delta t \Lambda_y^{n+1/2}) \rho_s^n \quad (37)$$

this result may be rewritten as:

$$(E + \Delta t^2 \Lambda_x^{n+1/2} \Lambda_y^{n+1/2}) = \Delta t (\Lambda_x^{n+1/2} + \Lambda_y^{n+1/2}) (\rho_s^{n,k+1} + \rho_s^n) + \Delta t f_1^{n+1/2}$$

or either:

$$(E + \Delta t^2 \Lambda_x^{n+1/2} \Lambda_y^{n+1/2}) \frac{\rho_s^{n,k+1} - \rho_s^n}{\Delta t} = (\Lambda_x^{n+1/2} + \Lambda_y^{n+1/2}) \frac{\rho_s^{n,k+1} - \rho_s^n}{2} + f_1^{n+1/2} \quad (38)$$



A comparison with Eq. (25) shows that Eq. (38) is actually equivalent to the first one except by the defined positive operator having a norm greater than one,

$$B \equiv E + \Delta t^2 \Lambda_x^{n+1/2} \Lambda_y^{n+1/2} = E + \mathcal{O}(\Delta t^2)$$

which acts on the term  $(\rho_s^{n,k+1} - \rho_s^n) / \Delta t$ . This means that this operator does not change the steady state solution. Furthermore, since  $\|B\| > 1$  the splitting scheme is more stable than the original scheme.

### 3.4. Spatial discretization

The grid is “staggered” and the discretization of the diffusive term of Eq. (15) is made according to the following formula, which preserves the conservation law implicit in the divergence:

$$\Delta t \frac{\partial}{\partial x} \left[ \frac{v_g}{\gamma \rho_s^2} \frac{\partial}{\partial x} v_g \rho_m \right] \approx \Delta t \frac{\partial}{\partial x} \left[ \frac{v_g}{\gamma (S^{n+1/2})^2} \frac{\partial}{\partial x} v_g \frac{\rho_m^{n,k+1} + \rho_m^n}{2} \right]$$

Upon writting:

$$Q_{i,j} = \frac{\Delta t v_g}{4\gamma (S_{i,j}^{n+1/2})^2}$$

we replace the diffusive term of  $\rho_m$  by:

$$\frac{\Delta t}{2} \frac{\partial}{\partial x} \left[ \frac{v_g}{\gamma \rho_s^2} \frac{\partial}{\partial x} v_g \rho_m \right] \approx \tag{39}$$

$$(Q_{i,j} + Q_{i,j+1}) \frac{v_g}{\Delta x} ((\rho_m)_{i,j+1} - (\rho_m)_{i,j}) - (Q_{i,j-1} + Q_{i,j}) \frac{v_g}{\Delta x} ((\rho_m)_{i,j} - (\rho_m)_{i,j-1}) = \tag{40}$$

$$(Q_{i,j-1} + Q_{i,j}) \frac{v_g}{\Delta x} (\rho_m)_{i,j-1} - (Q_{i,j-1} - 2Q_{i,j} + Q_{i,j+1}) (\rho_m)_{i,j} + (Q_{i,j} + Q_{i,j+1}) (\rho_m)_{i,j+1} \tag{41}$$

The diffusive terms of Eq. (14) are written in discrete form by using the usual three points centred formula, of second order. Neumann boundary conditions are used in the integration of the WA model, with derivatives in the direction perpendicular to the walls equal to zero. The algebraic linear systems were solved using a routine with gaussian elimination and pivoting, written by C. I. Christov, from the University of Louisiana at Lafayette.

## 4. Results

We present the result of one simulation with the nonlinear diffusion of  $\rho_m$ , run in a square box with  $20\mu m \times 5\mu m$ , using a grid containing  $3000 \times 750$  points. The initial condition consists of the time independent uniform base state plus a random with zero mean.

The model parameters adopted in the simulation are  $v_s = 1\mu m\ cm^{-1}$ ,  $d_c = 2.5^{-2}\mu m$ ,  $D_s = 3 \times 10^{-3}\mu m^2\ cy^{-1}$ ,  $v_g = 10^2\mu m\ cy^{-1}$ ,  $\gamma = 2 \times 10^{-2}$ ,  $\sigma = 250\mu m^{-2}\ cy^{-1}$ . The simulations were run with a time step of  $2.5 \times 10^{-3}\ cy$ .

During the simulations, we monitore the velocity of evolution of the patterns, by measuring:

$$L_1 = \frac{1}{\Delta t} \frac{\sum_{ij} |\rho_s^{n+1} - \rho_s^n| + \sum_{ij} |\rho_m^{n+1} - \rho_m^n|}{\sum_{ij} |\rho_s^{n+1}| + \sum_{ij} |\rho_m^{n+1}|} \tag{42}$$

where the summations are made over all points  $(i, j)$  of the grid. The simulations proceeded while the velocity of evolution of the patterns is above a specified value.

The results are presented in the followig figures. Figs. (1) to (4) show selected frames of the time evolution of  $\rho_s$ . The simulation evolves to a set of walls parallel to the slip direction. Fig. (6) shows the time evolution of  $L_1$ , the number of iterations per time step and the evolution of the maximum of  $\rho_s$ .

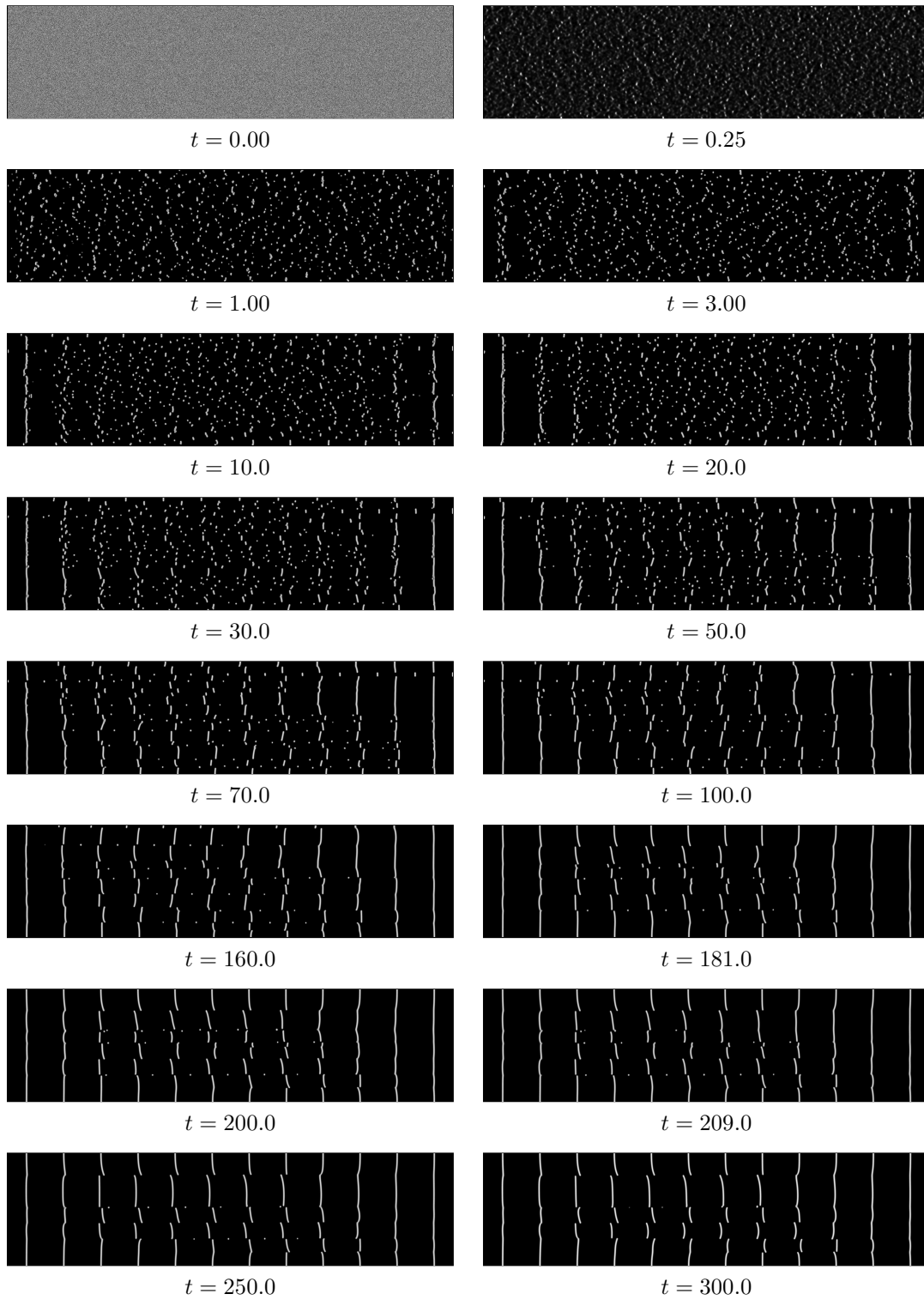


Figure 1: Time evolution of  $\rho_s$  (part 1).

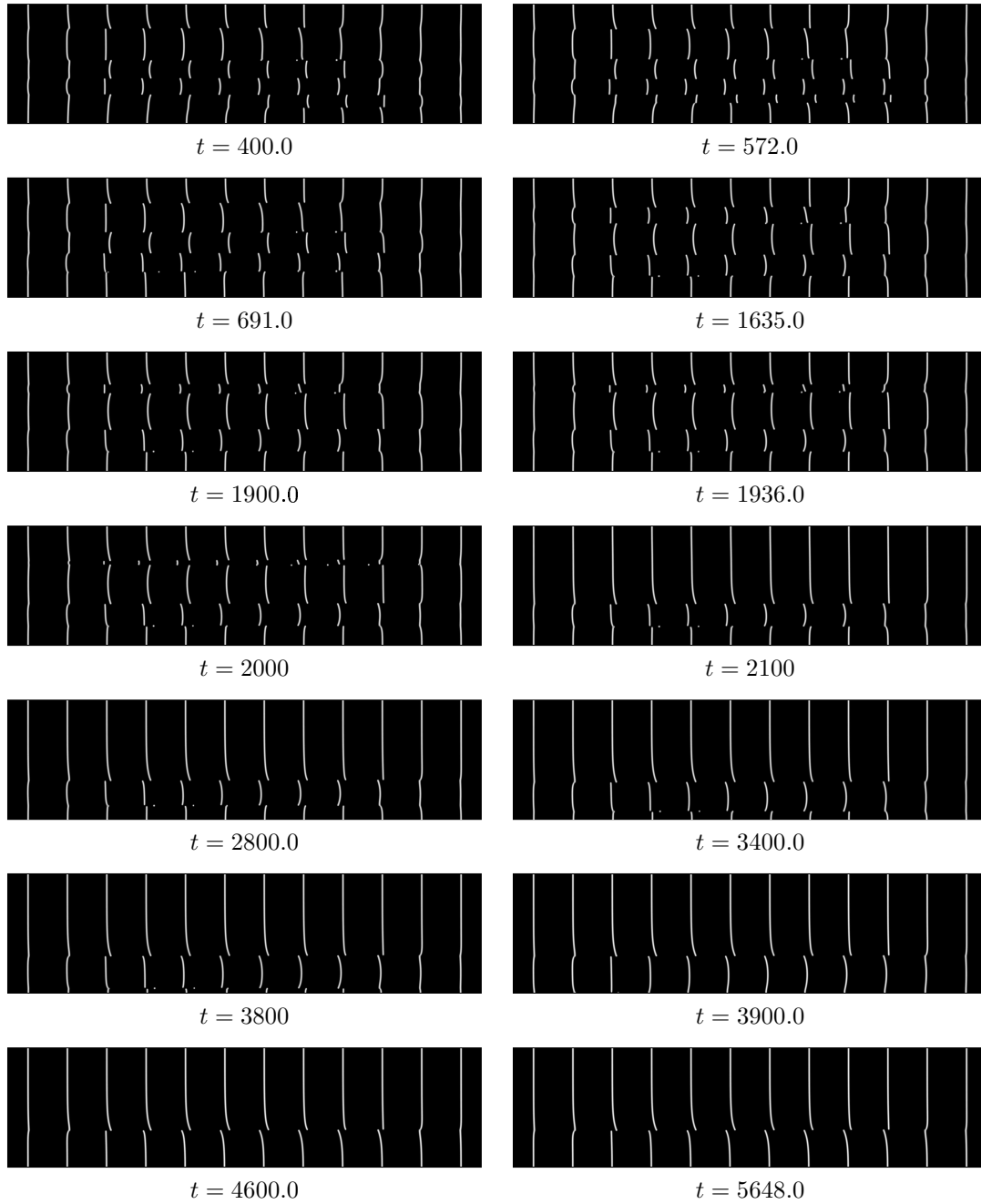


Figure 2: Time evolution of  $\rho_s$  (part 2).

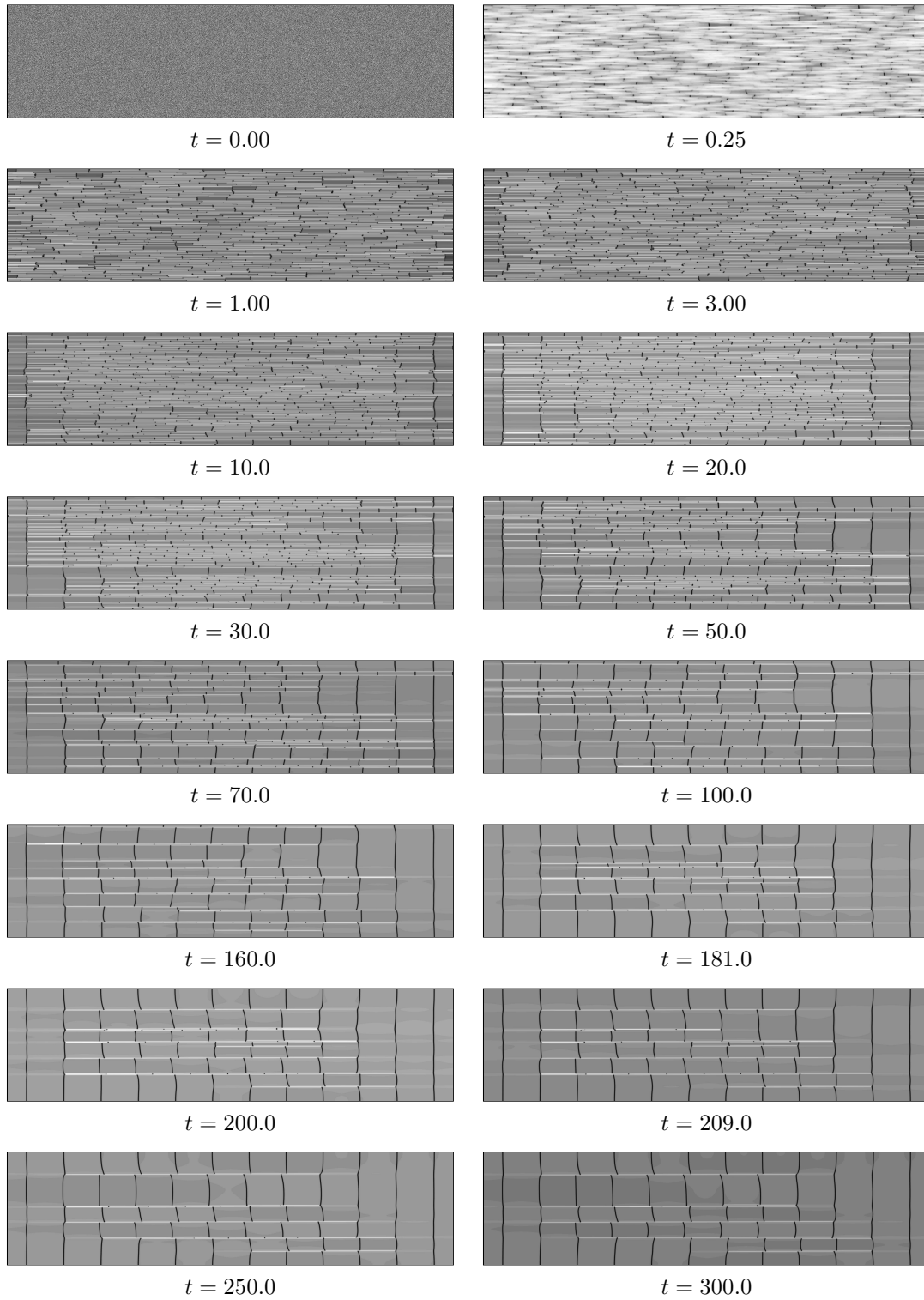


Figure 3: Time evolution of  $\rho_m$  (part 1).

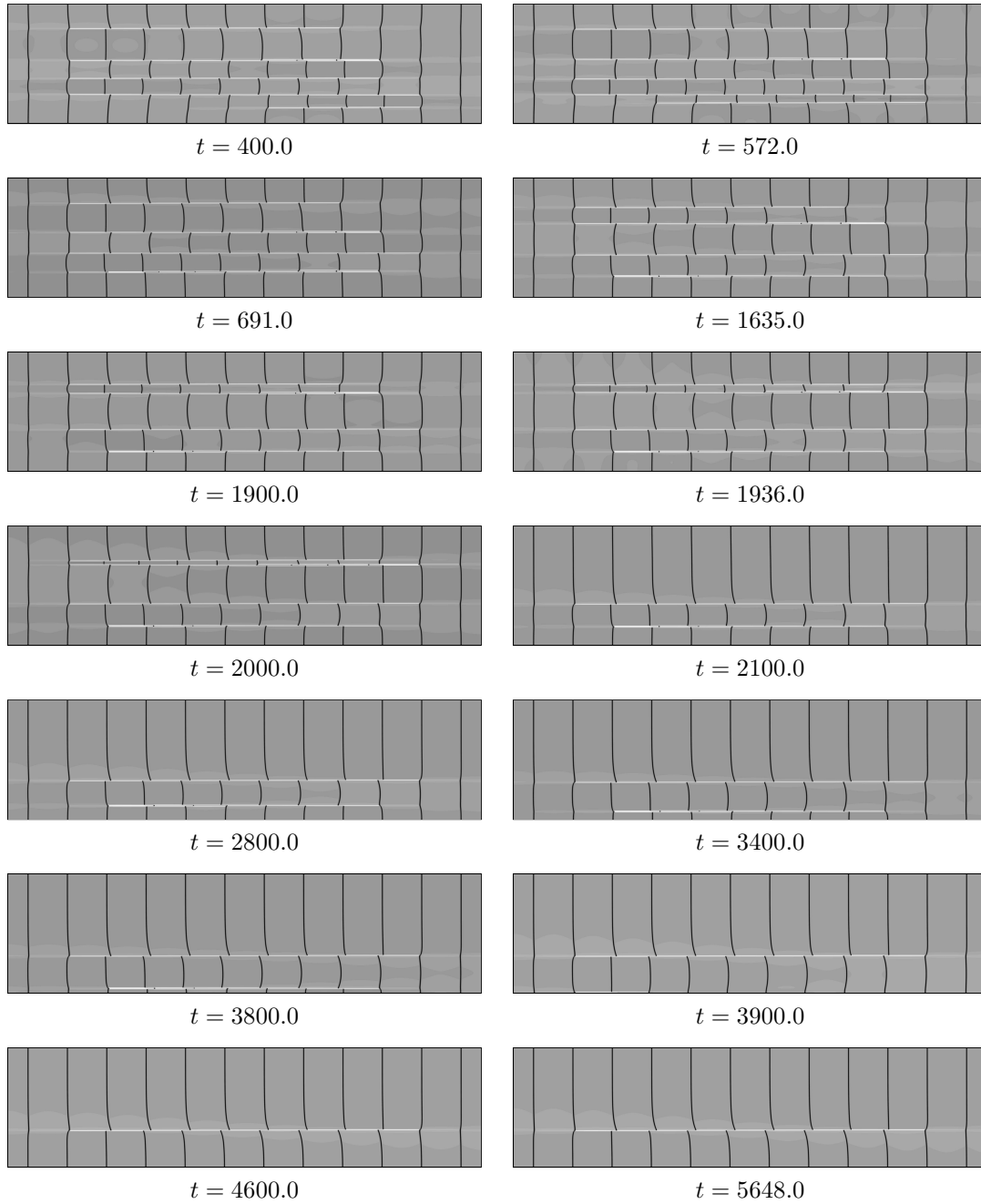


Figure 4: Time evolution of  $\rho_s$  (part 2).

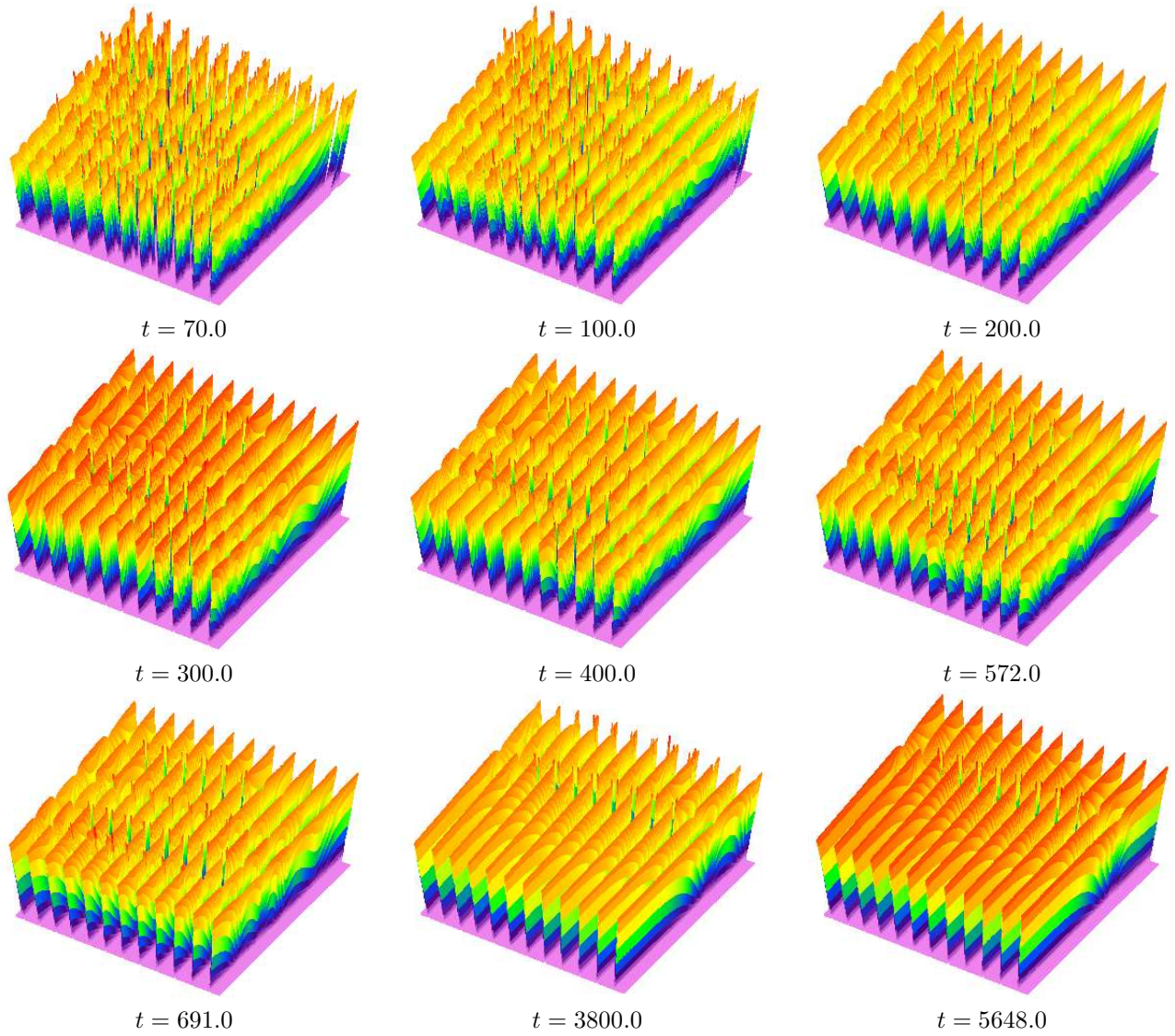


Figure 5: 3D figures of the time evolution of  $\rho_s$ .

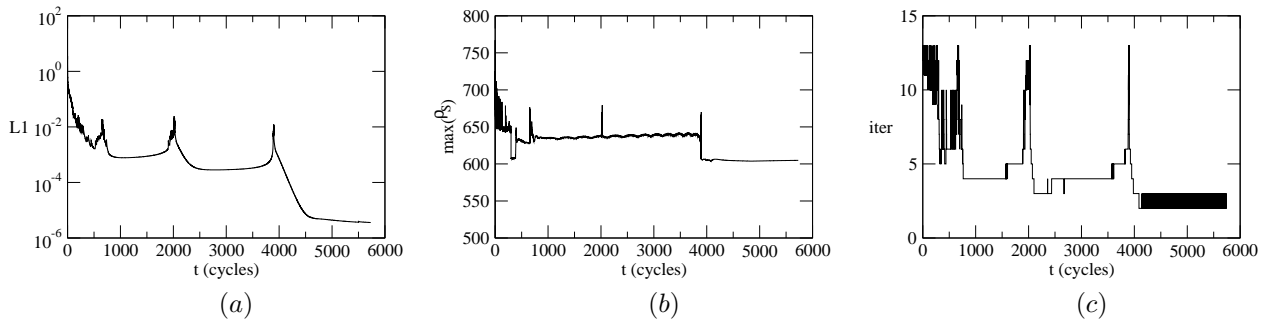


Figure 6: Curves of  $L_1 \times t$ ,  $\max(\rho_s) \times t$  and number of iterations  $\times t$ .

## 5. Acknowledgements

The authors warmly acknowledge the Center of Parallel Computing of the Federal University of Rio de Janeiro (NACAD/COPPE/UFRJ) and to prof. Alvaro L.G.A. Coutinho for the use of a cluster where the simulations presented were made. JP also acknowledges prof. C. I. Christov, from the University of Louisiana at Lafayette. The numerical scheme here proposed is based on his ideas.

## 6. References

- Christov, C. I. and Pontes, J., 2002, Numerical Scheme for Swift-Hohenberg Equation with Strict Implementation of Lyapunov Functional, “Mathematical and Computer Modelling”, Vol. 35, pp. 87–99.
- Ghoniem, N. M., Matthews, J. R., and Amoedo, R. J., 1990, “Res Mechanica”, Vol. 29, pp. 197.
- Kratochvil, J., 1988, On stability of dislocation structures in cyclically deformed metal crystals and formation of persistent slip bands, “Rev. Phys. Appliquée”, Vol. 23, pp. 419.
- Mughrabi, H., Ackermann, F., and Herz, K., 1979, Fong, E. T., editor, “Fatigue Mechanisms, ASTM-NBS-NSF Symposium”, Kansas City. ASTM, paper No. STP-675.
- Pontes, J., Walgraef, D., and Aifantis, E. C., 2006, On dislocation patterning: multiple slip effects in the rate approach, “Int. J. Plasticity”, Vol. 22, No. 8, pp. 1486–1505.
- Schiller, C. and Walgraef, D., 1988, Numerical simulation of persistent slip band formation, “Acta Metall.”, Vol. 36, No. 2, pp. 563–574.
- Walgraef, D., 1997, “Spatio-Temporal Pattern Formation”, Springer, New York.
- Walgraef, D. and Aifantis, E. C., 1985, Dislocation Patterning as the Result of Dynamical Instabilities, “J. Appl. Phys.”, Vol. 58, pp. 668.
- Walgraef, D. and Aifantis, E. C., 1986, On the formation and stability of dislocation patterns, “Int. J. Eng. Sci.”, Vol. 23, pp. 1351, 1359 and 1364.
- Walgraef, D. and Pontes, J., 2005, Numerical simulation of dislocation dynamics in materials subjected to cyclic loading, with the Walgraef-Aifantis model, “Anais do VII Encontro Modelagem Computacional”, Nova Friburgo, RJ, Brasil. Instituto Politécnico/UERJ, (in CD - ISBN ).
- Yanenko, N. N., 1971, “The Method of Fractional Steps”, Springer, New York.

## A PROCEDURE FOR SIMULATING THE NONLINEAR CONDUCTION HEAT TRANSFER IN A BODY WITH TEMPERATURE DEPENDENT THERMAL CONDUCTIVITY.

### ABRÃO MACANDI DONGALA

Universidade do Estado do Rio de Janeiro, UERJ, Faculdade de Engenharia.  
Rua São Francisco Xavier, 524 – 20550-013 – Rio de Janeiro, RJ, Brasil.  
[abramado1@hotmail.com](mailto:abramado1@hotmail.com)

### LEON MATOS RIBEIRO DE LIMA

Universidade do Estado do Rio de Janeiro, UERJ, Faculdade de Engenharia.  
Rua São Francisco Xavier, 524 – 20550-013 – Rio de Janeiro, RJ, Brasil.  
[leonmatos@hotmail.com](mailto:leonmatos@hotmail.com)

### ROGÉRIO MARTINS SALDANHA DA GAMA

Departamento de Engenharia Mecânica, Universidade do Estado do Rio de Janeiro, UERJ.  
Rua São Francisco Xavier, 524 – 20550-013 – Rio de Janeiro, RJ, Brasil.  
[rsgama@terra.com.br](mailto:rsgama@terra.com.br)

**Abstract.** *In this work a systematic procedure is proposed for simulating the conduction heat transfer process in a solid with a strong dependence of the thermal conductivity on the temperature. Such kind of inherently nonlinear partial differential equation, subjected to classical linear boundary conditions, will be solved with the aid of a Kirchoff Transform by means of a sequence of very simple linear problems. The proposed procedure provides the exact solution of the problem and induces finite dimensional approximations that can be used for computational simulations. Some typical cases are simulated by means of a finite difference scheme.*

**Keywords:** *Temperature dependent thermal conductivity, Kirchoff transform, heat transfer, nonlinear equations system.*

### 1. Introduction

Most of the conduction heat transfer phenomena are described under the assumption of temperature independent thermal conductivity. Such hypothesis is mathematically convenient because, in general, gives rise to linear partial differential equations. Nevertheless, the thermal conductivity is always a temperature dependent function. Many times, neglecting such dependence (assuming constant thermal conductivity), we have an inadequate mathematical description of the conduction heat transfer process.

The main objective of this work is to provide a reliable and systematic procedure for describing the conduction heat transfer in a rigid solid, with temperature dependent thermal conductivity, subjected to linear boundary conditions (Newton's law of cooling). This procedure is exact and employs only the tools utilized in problems in which the thermal conductivity is assumed to be a constant.

The first step consists of employing the Kirchoff transform in order to change the original problem into another one consisting of a linear partial differential equation subjected to a nonlinear boundary condition. The second step consists of regarding the new problem as the limit of a sequence of linear problems.

So, the problems with temperature dependent thermal conductivity will be regarded as the limit of a sequence whose elements are solution of linear problems.

### 2. Governing Equations

Let us consider a rigid, opaque and isotropic body at rest with domain represented by bounded open set  $\Omega$  with boundary  $\partial\Omega$ . The steady-state heat transfer process inside this body is mathematically described by the following elliptic partial differential equation



$$\operatorname{div}(k \operatorname{grad} T) + \dot{q} = 0 \quad \text{in } \Omega \quad (1)$$

where  $T$ ,  $\dot{q}$  and  $k$  denote, respectively, the temperature field, the internal heat generation rate (per unit volume) and the thermal conductivity. In this work the thermal conductivity is assumed to be a function of the local temperature. In other words,

$$k = \hat{k}(T) \quad (2)$$

Assuming that body boundary and the environment exchanges energy according to Newton's law of cooling, the boundary conditions associated with equation (1) are given by

$$-(k \operatorname{grad} T) \mathbf{n} = h(T - T_\infty) \quad \text{on } \partial\Omega \quad (3)$$

in which  $\mathbf{n}$  is the unit outward normal (defined on  $\partial\Omega$ ),  $h$  is the convection heat transfer coefficient and  $T_\infty$  is a temperature of reference

The resulting problem (nonlinear) is given by

$$\begin{aligned} \operatorname{div}(k \operatorname{grad} T) + \dot{q} &= 0 \quad \text{in } \Omega \\ -(k \operatorname{grad} T) \mathbf{n} &= h(T - T_\infty) \quad \text{on } \partial\Omega \end{aligned} \quad (4)$$

### 3. The Kirchoff Transform

Since the thermal conductivity is always positive-valued, the new variable  $\omega$  (Kirchoff transform) defined by

$$\omega = \int_{T_0}^T \hat{k}(\xi) d\xi = \hat{f}(T) \quad (5)$$

being an invertible function of  $T$ . This definition allows us to write

$$\operatorname{grad} \omega = k \operatorname{grad} T \quad (6)$$

So, the original problem can be rewritten as follows

$$\begin{aligned} \operatorname{div}(\operatorname{grad} \omega) + \dot{q} &= 0 \quad \text{in } \Omega \\ -(\operatorname{grad} \omega) \mathbf{n} &= h(\hat{f}^{-1}(\omega) - T_\infty) \quad \text{on } \partial\Omega \end{aligned} \quad (7)$$

where  $T \equiv \hat{f}^{-1}(\omega)$ . Although above problem remains nonlinear, this nonlinearity takes place only on the boundary, not in the partial differential equation. It is to be noticed that, since the thermal conductivity is everywhere positive,

$$k > 0 \Rightarrow \frac{d\omega}{dT} > 0 \quad \text{and} \quad \frac{dT}{d\omega} > 0 \quad \text{everywhere} \quad (8)$$

so, the temperature is a strictly increasing function of  $\omega$ .

For instance, if we have

$$k = \begin{cases} k_1 = \text{constant} & \text{if } T < T_0 \\ k_2 = \text{constant} & \text{if } T \geq T_0 \end{cases} \quad (9)$$

it is easy to show that

$$T \equiv \hat{f}^{-1}(\omega) = \omega \left[ \frac{1}{2k_2} - \frac{1}{2k_1} \right] + \omega \left[ \frac{1}{2k_2} + \frac{1}{2k_1} \right] + T_0 \quad (10)$$

#### 4. Constructing the Solution from a Sequence of Linear Problems

The solution of

$$\begin{aligned} \operatorname{div}(\operatorname{grad}\omega) + \dot{q} &= 0 \quad \text{in } \Omega \\ -(\operatorname{grad}\omega)\mathbf{n} &= h(\hat{f}^{-1}(\omega) - T_\infty) \quad \text{on } \partial\Omega \end{aligned} \quad (11)$$

can be represented by the limit of the nondecreasing sequence  $[\Phi_0, \Phi_1, \Phi_2, \dots]$  whose elements are obtained from the solution of the linear problems below

$$\begin{aligned} \operatorname{div}(\operatorname{grad}\Phi_{i+1}) + \dot{q} &= 0 \quad \text{in } \Omega \\ -(\operatorname{grad}\Phi_{i+1})\mathbf{n} &= \alpha\Phi_{i+1} + \beta_i \quad \text{on } \partial\Omega \\ \beta_i &= h(\hat{f}^{-1}(\Phi_i) - T_\infty) - \alpha\Phi_i \end{aligned} \quad (12)$$

in which  $\alpha$  is a sufficiently large, constant and  $\Phi_0 \equiv 0$ .

It is remarkable that, for each  $i$ , the function  $\Phi_{i+1}$  is the unknown and the function  $\Phi_i$  is known.

So,  $\hat{f}^{-1}(\Phi_i)$  is always known in (12), being evaluated from the following equation

$$\Phi_i = \int_{T_0}^{\hat{f}^{-1}(\Phi_i)} k(\zeta) d\zeta \quad (13)$$

For each spatial position, the root of the above equation is unique. This uniqueness is supported by equation (8).

The constant  $\alpha$  must be large enough for ensuring that, at any point of  $\Omega$ ,  $\Phi_{i+1} \geq \Phi_i$ . In reference [2] such result is proven as well as it is provided an upper bound for the constant  $\alpha$  [3]. For the problem considered in this work it is sufficient to choose  $\alpha$  such that

$$\alpha \geq \frac{h}{k_{MIN}} \quad (14)$$

where  $k_{MIN}$  is the minimum value of the thermal conductivity.

#### 5. Convergence

The limit of the sequence  $[\Phi_0, \Phi_1, \Phi_2, \dots]$ , denoted here by  $\Phi_\infty$  exists and is, in fact, a solution of the problem. To prove this assertion, let us begin showing that  $\Phi_\infty$  is a solution of (7). In other words

$$\begin{aligned} \operatorname{div}(\operatorname{grad}\Phi_\infty) + \dot{q} &= 0 \quad \text{in } \Omega \\ -(\operatorname{grad}\Phi_\infty)\mathbf{n} &= h(\hat{f}^{-1}(\Phi_\infty) - T_\infty) \quad \text{on } \partial\Omega \end{aligned} \quad (15)$$

Since  $\beta_\infty$  is given by

$$\beta_\infty = h \left( \hat{f}^{-1}(\Phi_\infty) - T_\infty \right) - \alpha \Phi_\infty \quad (16)$$

we have that (12) and (15) coincide. So,  $\Phi_\infty$  is a solution. Now, taking into account that the sequence is nondecreasing and has an upper bound, we ensure the convergence, once that the solution of (15) belongs to the same space of the solutions of (12) for each  $i$  [4] e [5].

## 6. An Example

Let us consider the following problem (spherical body with uniform heat generation, surrounded by the same medium, with  $h = 1$  and  $T_\infty = 1$  - in some system of units)

$$\left[ \frac{1}{r^2} \frac{d}{dr} \left( r^2 k \frac{dT}{dr} \right) \right] + 1 = 0 \quad \text{for } 0 \leq r < 1 \quad (17)$$

$$-k \frac{dT}{dr} = T \quad \text{at } r = 1$$

in which it is assumed that  $T$  represents an absolute temperature and  $k = 3T + 2$ . The solution is easily reached and given by

$$T = -\frac{2}{3} + \left[ \frac{(1-r^2)}{9} + 1 \right]^{1/2} \quad (18)$$

Now, let us employ the proposed procedure. With the Kirchoff transform, the problem yields

$$\left[ \frac{1}{r^2} \frac{d}{dr} \left( r^2 \frac{d\omega}{dr} \right) \right] + 1 = 0 \quad \text{for } 0 \leq r < 1 \quad (19)$$

$$-\frac{d\omega}{dr} = \hat{f}^{-1}(\omega) = -\frac{2}{3} + \left[ \frac{4}{9} + \frac{2\omega}{3} \right]^{1/2} \quad \text{at } r = 1$$

The linear procedure for reaching the elements of the sequence is represented as follows

$$\left[ \frac{1}{r^2} \frac{d}{dr} \left( r^2 \frac{d\Phi_{i+1}}{dr} \right) \right] + 1 = 0 \quad \text{for } 0 \leq r < 1$$

$$-\frac{d\Phi_{i+1}}{dr} = \alpha \Phi_{i+1} + \beta_i \quad \text{at } r = 1 \quad (20)$$

$$\beta_i = -\frac{2}{3} + \left[ \frac{4}{9} + \frac{2\Phi_i}{3} \right]^{1/2} - \alpha \Phi_i$$

or, simply as

$$\left[ \frac{1}{r^2} \frac{d}{dr} \left( r^2 \frac{d\Phi_{i+1}}{dr} \right) \right] + 1 = 0 \quad \text{for } 0 \leq r < 1 \quad (21)$$

$$-\frac{d\Phi_{i+1}}{dr} = \alpha \Phi_{i+1} - \alpha \Phi_i - \frac{2}{3} + \left[ \frac{4}{9} + \frac{2\Phi_i}{3} \right]^{1/2} \quad \text{at } r = 1$$

Since the problem makes sense only for  $T > 0$ , we have that  $k_{MIN} > 2$ . So, we can work with any  $\alpha \geq 1/2$ . We shall use  $\alpha = 3$ !

The general solution of equation

$$\left[ \frac{1}{r^2} \frac{d}{dr} \left( r^2 \frac{d\Phi_{i+1}}{dr} \right) \right] + 1 = 0 \quad \text{for } 0 \leq r < 1 \quad (22)$$

is

$$\Phi_{i+1} = -\frac{r^2}{6} + C_{i+1} \quad \text{for } 0 \leq r < 1 \quad (23)$$

where the constant  $C_{i+1}$ , for  $i > 0$ , is obtained from the boundary condition. In other words,

$$\frac{1}{3} = 3 \left[ -\frac{1}{6} + C_{i+1} \right] - 3 \left[ -\frac{1}{6} + C_i \right] - \frac{2}{3} + \left[ \frac{4}{9} + \frac{2}{3} \left( -\frac{1}{6} + C_i \right) \right]^{1/2} \quad (24)$$

Then,

$$C_{i+1} = C_i + \frac{1}{3} \left[ 1 - \sqrt{\frac{1}{3} + \frac{2}{3} C_i} \right] \quad (25)$$

The constant  $C_1$  is obtained from

$$\frac{1}{3} = 3 \left[ -\frac{1}{6} + C_1 \right] \Rightarrow C_1 = \frac{5}{18} \quad (26)$$

If we employ  $\alpha = 20$ , then

$$C_{i+1} = C_i + \frac{1}{20} \left[ 1 - \sqrt{\frac{1}{3} + \frac{2}{3} C_i} \right] \quad (27)$$

and the constant  $C_1$  is obtained from

$$\frac{1}{3} = 20 \left[ -\frac{1}{6} + C_1 \right] \Rightarrow C_1 = \frac{11}{60} \quad (28)$$

Table 1 presents a comparison between the values obtained for  $C_i$  with two distinct values of  $\alpha$

	$\alpha = 3$	$\alpha = 20$
$i =$	$C_i =$	$C_i =$
1	0.371083678318	0.199585905322
2	0.450399853146	0.215439507012
3	0.518403223696	0.230908373418
4	0.577078085519	0.246005928926
5	0.627951577215	0.260744856563
10	0.800503112959	0.329459272507
20	0.940194171974	0.445715284244
50	0.998272514569	0.679342555325
100	0.999995202718	0.866108252119
500	0.99999985099	0.999842551009
1000	0.99999985099	0.99999949824

Table 1 – A comparison between results obtained with  $\alpha = 3$  and  $\alpha = 20$ .

It can be proven that  $C_\infty = 1$ . Hence, the limit of the sequence is given by

$$\Phi_\infty = -\frac{r^2}{6} + 1 \quad \text{for } 0 \leq r < 1 \quad (29)$$

The solution  $\omega$  is exactly the limit of the sequence. Taking into account that

$$\hat{f}^{-1}(\omega) = T = -\frac{2}{3} + \left[ \frac{4}{9} + \frac{2\omega}{3} \right]^{1/2} \quad (30)$$

we have the following result (coincident with the exact solution, previously obtained)

$$T = -\frac{2}{3} + \left[ \frac{4}{9} + \frac{2}{3} \left( -\frac{r^2}{6} + 1 \right) \right]^{1/2} \quad (31)$$

### 7. A computational numerical simulation example

A computational support has been developed parallel to the mathematical modeling in order to provide numerical results of the proposed method. All the programs were made in MATLAB. Now, a computational simulation for a typical problem will be shown. The physical characteristics of the problem are described by the following parameters: convection heat transfer coefficient  $h = 5$ , heat generation  $\dot{q} = 10$ , reference temperature  $T_\infty = 20$ . The conduction is processed in a rectangular flat plate. The spatial domain is described in a rectangular cartesian coordinates system. The plate is represented by a rectangular elements mesh for Finite Differences Method application, and the linear equations system is solved by the Gauss-Seidel iterative method. The temperature dependence of the thermal conductivity is given by:

$$k = \begin{cases} k_1 = 45 \rightarrow T < 21 \\ k_2 = 10 \rightarrow T \geq 21 \end{cases}$$

In this simulation we assume  $\alpha = 8$ . The following figures show the iterative evolution of the simulation.

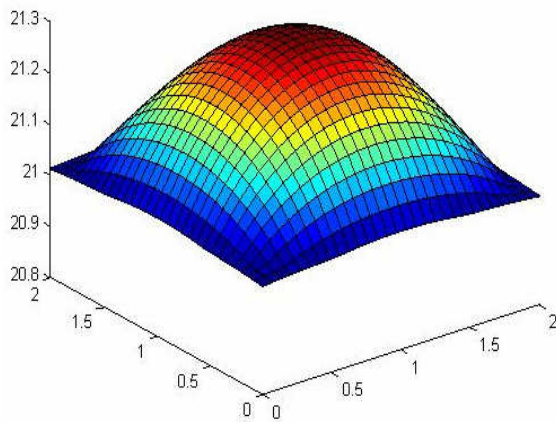


Figure 1. first iteration result

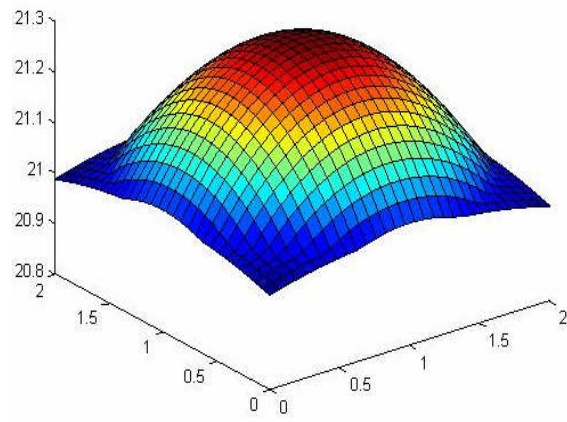


Figure 2. seventh iteration result

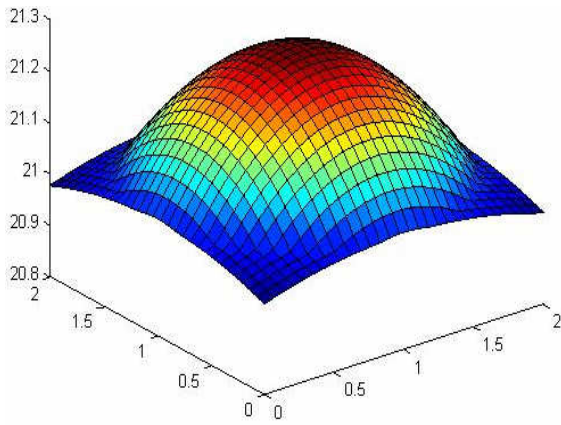


Figure 3. thirteenth iteration result

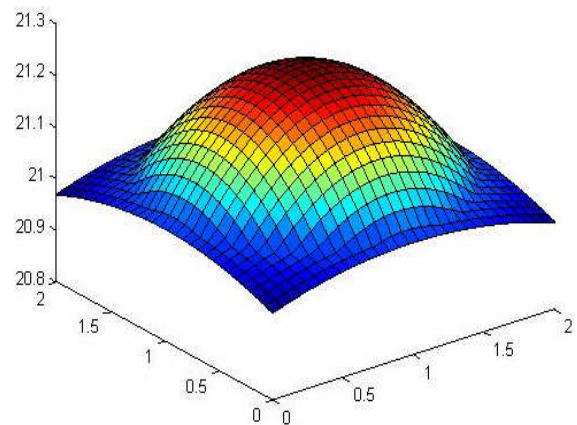


Figure 4. seventeenth iteration result

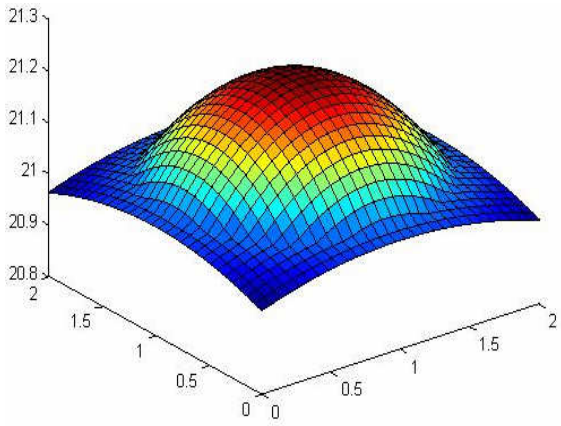


Figure 5. twenty third iteration result

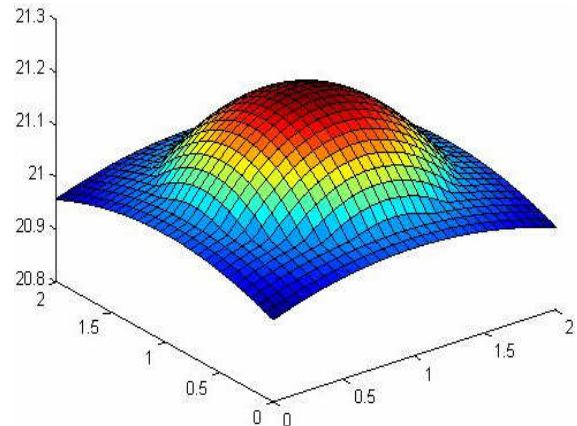


Figure 6. thirtieth and final result

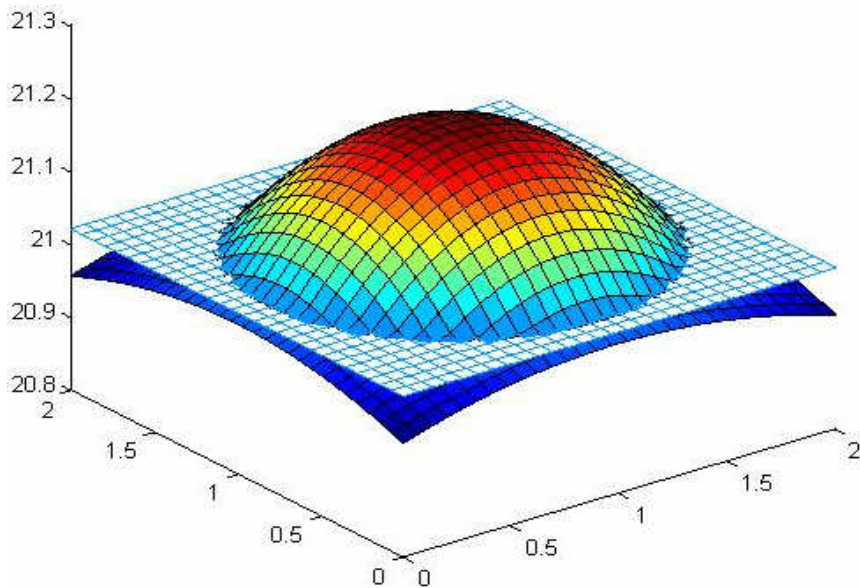


Figure 7. plane dividing the temperature field in two regions of constant thermal conductivity

## **8. Final remarks**

The presented procedure proved to be a simple, but efficient subsidy for solving problems of conduction heat transfer with temperature dependent thermal conductivity, by means of classic tool employed on heat transfer linear problems, such as finite differences.

## **9. References**

- Slattery, J. C., 1972, "Momentum, energy and mass transfer in continua", McGraw-Hill, Tokyo.
- Gama, R. M. S., "An a priori upper bound estimate for conduction heat transfer problems with temperature-dependent thermal conductivity (to appear)" *Int. J. Non-Linear Mech.*
- Gama, R. M. S., 2000, "An upper bound estimate for a class of conduction heat transfer problems with nonlinear boundary conditions", *Int. Comm. Heat Mass Transfer*, vol. 27, pp.955-964.
- Helmberg, G., 1974, "Introduction to spectral theory in Hilbert space", North-Holland.
- John, F., 1982, "Partial differential equations", Springer-Verlag.
- Wylie, C. R., 1960, "Advanced Engineering Mathematics", McGraw-Hill.

## ESTUDO NUMÉRICO DO ESCOAMENTO TURBULENTO SOBRE TERRENOS COMPLEXOS

### Gilberto Augusto Amado Moreira

UFMG – Universidade Federal de Minas Gerais, Departamento de Engenharia Mecânica, Av. Antônio Carlos, 6627 – Pampulha, 31270-901 – Belo Horizonte, MG, Brasil.  
gilbertomoreira@ufmg.br

### Ramon Molina Valle

UFMG – Universidade Federal de Minas Gerais, Departamento de Engenharia Mecânica, Av. Antônio Carlos, 6627 – Pampulha, 31270-901 – Belo Horizonte, MG, Brasil.  
ramon@demec.ufmg.br

### Marcio Arêdes Martins

UFV – Universidade Federal de Viçosa, Avenida P. H. Rolfs s/n - Campus - UFV CEP 36570-000 - Viçosa - MG  
aredes@ufv.br

### Carlos Alexandre M. Nascimento

CEMIG – Cia. Energética de Minas Gerais, CEMIG - Cia. Energética de Minas Gerais, Avenida Barbacena 1200, Santo Agostinho, 30123-970 - Belo Horizonte, MG, Brasil.  
caxandre@cemig.com.br

**Resumo.** Este trabalho tem como finalidade modelar a camada limite atmosférica visando a obtenção de campos de velocidade que possam ser utilizados na otimização de linhas de transmissão (LTs) aéreas. Os resultados obtidos do modelo numérico são validados com dados experimentais encontrados na literatura. As equações que regem estes escoamentos geofísicos são basicamente as equações de continuidade e de conservação de quantidade de movimento, além das equações de conservação de energia e de conservação da espécie química. No modelo, as flutuações nas equações de conservação são definidas por meio da viscosidade turbulenta, em conjunto com as equações do modelo de turbulência RNG  $k-\epsilon$ . No modelo são consideradas também as irregularidades do terreno e a vegetação através de uma lei logarítmica. O contorno superior do domínio é prescrito acima da camada limite atmosférica, ou seja, acima de 500 m da superfície da CLA. Os resultados apresentados neste trabalho envolvem a validação do vento para a topologia mundialmente conhecida de Askervein, cujos resultados experimentais são usados para a validação do modelo numérico. Também são apresentados resultados para uma topografia de Belo Horizonte, no estado de Minas Gerais, Brasil.

*Palavras chave:* Camada limite Atmosférica 1, Modelo Numérico 2, Terrenos Complexos 3.

### 1. Introdução

O conhecimento dos campos de velocidade fornecidos por um modelo computacional previamente validado, torna possível comprovar se os projetos de LTs aéreas estão adequados para as condições ambientais para as quais foram concebidos e ainda determinar uma nova realidade de campo, se for o caso desses dados indicarem grandes alterações na velocidade de vento.

Uma ferramenta importante para a obtenção dos mapas de vento nas regiões desejadas é a simulação da camada limite atmosférica em computador. O modelo, uma vez validado com dados experimentais, permitirá fazer uma correlação entre a topografia e rugosidade do terreno com o comportamento do vento na região. Perfis de velocidade e de temperatura do ar próximos ao solo podem ser obtidos para qualquer região onde a topologia do terreno for conhecida. De uma forma geral, a necessidade de realização de estudos sobre a alteração dos dados climatológicos na terra torna o modelo computacional da camada limite atmosférica de grande importância na reavaliação dos critérios e procedimentos adotados em projeto para uma nova realidade de campo.

Neste trabalho é apresentado um estudo numérico do fluxo de ar atmosférico sobre um terreno, apresentando variações abruptas e/ou suaves na superfície, incluindo matas e relevos diferenciados. As soluções das equações do modelo matemático permitem a obtenção dos perfis de velocidade acima do terreno.

De acordo com Albertson et al. (1999), um domínio típico para a simulação de escoamentos atmosféricos deve ser maior que 6 km nas direções  $x$  e  $y$ , respectivamente e o contorno superior do domínio deve ser prescrito acima da camada limite atmosférica, ou seja, acima de 500 m da superfície da CLA. Nesta altitude o escoamento é considerado estável o suficiente para que a condição de Neumann possa ser imposta a todas as variáveis do escoamento. O contorno inferior apresenta maior influência sobre a estrutura da turbulência do que os demais contornos devido ao fato de que a produção mecânica da turbulência é decorrente da ação de forças de arraste sobre a superfície e, ainda, pelos fluxos de calor, responsáveis pelos movimentos convectivos, serem originados da superfície. As transferências entre campos



escalares e quantidade de movimento entre o escoamento e a superfície são descritas por equações de transferências instantâneas.

Os modelos para simulação dos campos de velocidade de vento são classificados em três classes: modelos de circulação global, modelos de previsão de climas e modelos de mesoescalas Camilla et al. (1999). Os modelos de circulação global utilizam domínios entre 200 e 500 km e são utilizados para analisar os campos de vento sobre a superfície terrestre. Os modelos de previsão de climas utilizam domínios entre 50 e 100 km e são utilizados para resolver estruturas de frentes climáticas. Os modelos de mesoescalas utilizam domínios típicos entre 2 a 50 km e são empregados no estudo dos campos de velocidade de vento sobre topologias (topografias).

Nesse trabalho, são apresentados os resultados de simulação do estudo da CLA, que permitem fazer uma análise teórica da distribuição da velocidade dos ventos em regiões vizinhas a uma linha de transmissão aérea em estudo. Os resultados foram validados com os dados experimentais da literatura e dados de monitoramento em campo, de forma a buscar o aprimoramento do modelo numérico proposto. O modelo proposto pode ser aplicado a uma diversidade de problemas geofísicos e de engenharia. O modelo é capaz de resolver variações de densidade do ar, resultantes dos efeitos térmicos e de altitude, por meio da aproximação de Boussinesq. O modelo de turbulência utilizado neste trabalho para o estudo da CLA é o RNG k-ε, amplamente utilizado em escoamentos atmosféricos Zhang et al. (1996). O modelo proposto para a CLA, pertence à classe de modelos de mesoescalas e não considera o chamado efeito Coriolis, que pode ser descrito como o efeito da rotação da terra sobre as estruturas do escoamento.

## 2. Metodologia

### 2.1. Modelo Matemático

$$\frac{\partial u_i}{\partial x_i} = 0 \quad (1)$$

$$\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{\partial}{\partial x_i} \left( \frac{p}{\rho_0} + \frac{2}{3} k \right) + \frac{\partial}{\partial x_j} \left[ \nu_t \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] - \underbrace{\frac{\delta \rho}{\rho_0} g \delta_{i3}}_I \quad (2)$$

$$\frac{\delta \rho}{\rho_0} = -\frac{\delta T}{T_0} \quad (3)$$

$$c_p \left( \frac{\partial T}{\partial t} + u_j \frac{\partial T}{\partial x_j} \right) = \frac{\partial}{\partial x_j} \left( \frac{\nu_t}{Pr_T} \frac{\partial T}{\partial x_j} \right) + Q \quad (4)$$

As Eqs. (1, 2 e 4) representam a conservação da massa, quantidade de movimento e energia, respectivamente, onde,  $u_i$  são as componentes de velocidade,  $\rho_0$  é densidade de referência,  $p$  é a pressão,  $k$  é a energia cinética,  $g$  é a aceleração da gravidade,  $t$  é o tempo,  $\nu_t$  é a viscosidade efetiva,  $c_p$  é a capacidade térmica do ar a pressão constante,  $Pr_T$  é o número de Prandtl turbulento,  $T$  é a temperatura e  $Q$  é a taxa de geração de energia. A Eq. (3), conhecida como equação de estado, permite acoplar as equações de quantidade de movimento, Eq. (2), com a equação da energia, Eq. (4). O termo  $I$ , definido na Eq. (2), é conhecido como termo de flutuação. A definição do termo de flutuação caracteriza diversos modelos disponíveis na literatura. Usualmente, os modelos apresentados na literatura (Andreas, 1996; Huser, 1997; Kim, 1998 e Uchida, 1999) utilizam a aproximação de Boussinesq e os modelos de turbulência derivados do modelo k-ε, onde são feitas as seguintes considerações:

- a viscosidade dinâmica ( $\mu$ ) e a condutividade térmica ( $k$ ) são consideradas constantes em todo o domínio fluido;
- as velocidades do escoamento apresentam magnitude na qual o fluido pode ser considerado incompressível;
- as flutuações de turbulência são muito inferiores quando comparada com as respectivas grandezas médias;
- o efeito térmico associado à viscosidade do fluido pode ser desconsiderado; e,
- as flutuações de densidade são significativas apenas quando multiplicadas pelo vetor da gravidade ( $g$ ).

### 2.2. Modelos para o termo de flutuação

O modelo de Uchida et al. (1999), utilizado neste trabalho, define o termo de flutuação considerando o transporte da perturbação de densidade ( $\rho'$ ), incluindo a variação hidrostática da densidade  $\rho_{B(z)}$ . A equação de transporte para a perturbação de densidade é definida pelos autores por meio da equação:

$$\frac{\partial \rho'}{\partial t} + u_j \frac{\partial \rho'}{\partial x_j} = \alpha \frac{\partial^2 \rho'}{\partial x_j \partial x_j} - w \frac{d\rho_B}{dz} \quad (5)$$

onde  $\alpha$  é difusividade molecular para a densidade e a perturbação da densidade e o termo de flutuação são definidos como:

$$\rho' = \rho - \rho_B(z) \quad (6)$$

$$S_i = -\rho' \rho_0 g \quad (7)$$

onde  $\rho_0$  é a densidade de referência e  $w$  é a componente vertical da velocidade.

As Eqs. (1 a 5) são resolvidas numericamente pelo software CFX o qual permite a implementação de equações diferenciais parciais e algébricas adicionais, como por exemplo, a equação de estado. A equação da energia pode ser, ainda, acoplada a este modelo, pois as variações de temperatura em função da altitude promovem alterações na densidade do fluido.

A equação de transporte escalar é uma equação de convecção-difusão, com termo fonte, definida como:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (u_j \phi) - \nabla \cdot \left( \left( \rho D_\phi + \frac{\mu_t}{Sc_t} \right) \frac{1}{\rho} \nabla \phi \right) = S_\phi \quad (8)$$

onde  $\phi$  é uma variável adicional volumétrica (e.g. kg/m<sup>3</sup>),  $D$  é a difusividade cinemática (m<sup>2</sup>/s),  $\mu_t$  é a viscosidade turbulenta,  $Sc_t$  é o número de Schmidt turbulento e  $S$  é o termo fonte.

O modelo de turbulência usual em escoamentos geofísicos é o modelo RNG k- $\epsilon$ , no qual é utilizada a viscosidade turbulenta ( $\mu_t$ ), em conjunto com as equações da energia cinética turbulenta,  $k$ , e da dissipação desta energia,  $\epsilon$ . Estes termos são definidos como:

$$\frac{D(\rho k)}{Dt} = \frac{\partial}{\partial x_j} \left( \frac{\mu_t}{\sigma_k} \frac{\partial k}{\partial x_j} \right) + \Pi + G - \rho \epsilon \quad (9)$$

$$\frac{D(\rho \epsilon)}{Dt} = \frac{\partial}{\partial x_j} \left( \frac{\mu_t}{\sigma_\epsilon} \frac{\partial \epsilon}{\partial x_j} \right) + C_1 \frac{\epsilon}{k} [\Pi + G(1 - C_3)] - C_2 \rho \frac{\epsilon^2}{k} \quad (10)$$

onde  $\Pi$  é o termo de produção de energia cinética ( $k$ ) e  $G$  é o termo de flutuação, sendo definidos como:

$$\Pi = \mu_t \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \frac{\partial U_j}{\partial x_i} \quad (11)$$

$$G = g_i \frac{\mu_t}{\sigma_{t,\Theta} T_0} \frac{\partial \Theta}{\partial x_i} \quad (12)$$

Os parâmetros do modelo ( $C$  and  $\sigma$ ) são definidos na Tabela 1.

Tabela 1 – Parâmetros do Modelo RNG k- $\epsilon$

$C_\mu$	$C_1$	$C_2$	$\sigma_k$	$\sigma_\epsilon$	$\sigma_{t,\Theta}$	$\sigma_{t,C}$
0,085	1,44 - $\mathbb{Y}$	1,68	0,72	0,72	0,9	0,9

Onde

$$\mathbb{Y} = \frac{\eta(1 - \eta/4,38)}{1 + 0,012\eta^3} \quad (13)$$

$$\eta = \frac{k\sqrt{2D(U) : D(U)}}{\epsilon} \quad (14)$$

O parâmetro  $C_3$  varia entre 0 e 1, sendo este valor dependente da estabilidade da camada limite atmosférica.

### 2.3. Condições de Contorno

#### 2.3.1. Condições de Contorno da Superfície do solo

$$u = u_{ref} \cdot \frac{\ln\left(\frac{z}{z_0}\right)}{\ln\left(\frac{z_{ref}}{z_0}\right)} [m/s] \quad (15)$$

onde  $u_{ref}$  é a velocidade de referência, medida a uma altura de 10m,  $u_{ref} = 5,1 m/s$  (valor médio obtido por dados experimentais coletados nas posições V1 e V3),  $z_{ref}$  é a altura de referência (10 m),  $z_0$  é a rugosidade. A literatura recomenda usar para a rugosidade, os seguintes valores:  $z_0 = 1,00m$  (cidade),  $z_0 = 0,30m$  (floresta),  $z_0 = 0,03m$  (grama baixa) e  $z_0 = 0,0001m$  (água). No presente trabalho, o valor de  $z_0$  utilizado foi de 0,03

Os perfis de entrada para os campos de energia cinética turbulenta ( $k$ ) e taxa de dissipação da energia cinética turbulenta ( $\epsilon$ ) também são prescritos. Prescrevendo estes perfis no plano de entrada do domínio computacional, a velocidade do vento e a classe de estabilidade atmosférica podem ser impostas. No modelo adotado, a rugosidade da superfície, o fluxo de calor no solo e a classe de estabilidade são os parâmetros que definem o estado da atmosfera, sendo estes parâmetros definidos como condições de contorno.

#### 2.3.2. Condições de Contorno de Parede

A condição de parede (wall boundary condition) é aplicada automaticamente pelo CFX na superfície do domínio (terreno). O terreno é considerado como uma superfície não deslizante ( $u = 0$ ) e com rugosidade média  $z_0$ . Em relação ao transporte de energia, a superfície em condição isotérmica foi adotada.

#### 2.3.3. Condições de Contorno de Simetria

A condição de contorno de simetria foi adotada para o topo e para as laterais do domínio. Esta condição de contorno garante fluxo nulo de todas as grandezas nas respectivas superfícies.

#### 2.3.4. Condições de Contorno de Saída

A condição de contorno na saída do domínio foi imposta por meio da especificação da média da pressão estática nula e gradiente nulo na direção do escoamento para as demais grandezas.

#### 2.3.5. Obtenção do Termo Fonte em Função do Número de Froude

Após o estudo das condições de contorno, é necessário verificar a influência do termo fonte, o qual é dado por:

$$S = g \cdot \Delta\rho = g \cdot (\rho - \rho_0) \quad (16)$$

onde  $S$  é termo fonte,  $g$  é a aceleração da gravidade,  $\rho$  é a densidade do meio e  $\rho_0$  é a densidade de referência. O termo fonte pode ser relacionado com o número adimensional de Froude ( $Fr$ ), da seguinte forma:

$$Fr = \frac{V}{\sqrt{gL}} \quad (17)$$

onde  $V$  é a velocidade e  $L$  é o comprimento característico (altura da maior elevação)

A Eq.(17) pode ser transformada em:

$$Fr = \frac{U_\infty}{(g \cdot L \cdot \Delta\rho / \rho_0)^{1/2}} \quad (18)$$

onde  $U_\infty$  representa a velocidade desenvolvida acima de 500 metros e o número de Froude expressa a relação entre as forças de inércia e as forças de gravidade. Desta forma, o termo fonte em relação à direção  $w$  (altura) pode ser escrito da seguinte forma

$$S_w = -\frac{U_\infty^2 \cdot \rho_0}{L \cdot Fr^2} \quad (19)$$

onde  $U_\infty$  é a velocidade a uma altura de 500m,  $\rho_0$  é a densidade de referência ( $\rho_0 = 1,185 \text{ kg/m}^3$ ) e  $L$  é o comprimento característico, usualmente é a altura da maior elevação.

Desta forma, para poder validar o modelo com mapas topográficos da literatura torna-se necessário definir a faixa de número de Froude na qual a literatura apresenta dados experimentais. A tabela 2 mostra a relação entre o estado da atmosfera e o número de Froude

Tabela 2 – Correlação entre o número de Froude e estado da atmosfera

Valor de Froude	Estado da Atmosfera
Acima de 1000	Neutra
10 - 1000	Estratificadas e Estáveis
-100 - 10	Instáveis

### 3. Tratamento numérico das equações

A modelagem e a simulação do escoamento são efetuadas por meio do emprego do pacote comercial CFX-5.6. Este software possui, resumidamente, as seguintes características: discretiza as equações de conservação pelo método de volumes finitos centrado no vértice; resolve problemas laminares e turbulentos tridimensionais; utiliza malhas não-estruturadas e híbridas; resolve problemas conjugados de calor e escoamento de fluidos, entre outras. A utilização de malhas não-estruturadas permite que refinamentos de malha sejam aplicados próximos às superfícies, onde grandes variações de velocidade e temperatura estão presentes.

### 4. Validação do modelo

A validação apresentada no presente trabalho compara os dados obtidos do modelo aqui apresentado com dados experimentais da colina de Askervein. Esta é uma colina que possui 116 m de altura na costa oeste da ilha do South Uist na Outer Hebrides na Escócia. Ela é relativamente isolada e suave, com seu ápice a 126 m do nível do mar e sua forma é aproximadamente elíptica, com o seu eixo maior aproximadamente de 2 km de comprimento, orientado na direção 133°-313° (linha B-B), ou seja, SE-NO (sudeste-noroeste). O seu eixo menor, de 1km de comprimento, está orientado na direção 43°-223° (linha A-A ou AA-AA), ou seja, SO-NE (sudoeste-nordeste). Sua vegetação é baixa e com rugosidade não uniforme. Os conjuntos completos de dados coletados em Askervein encontram-se disponíveis na literatura. As medidas de campo foram feitas utilizando vários dispositivos, medindo ao longo das linhas A-A, AA-AA e B-B, a uma altura de 10m acima da superfície da colina. A Figura 1 (a) mostra o mapa topográfico da colina de Askervein com as seções de medição. A fig 1 (b) mostra a topografia de Askervein digitalizada no CFX

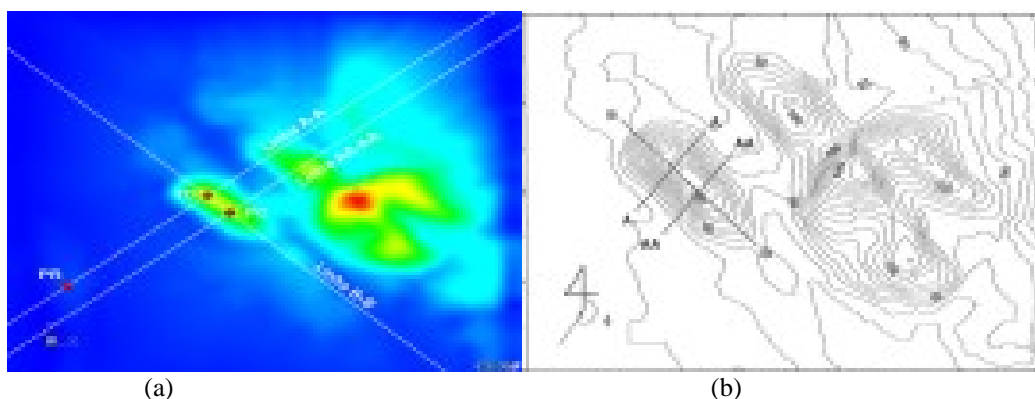


Figura 1 - Mapa Topográfico da colina Askervein. (a) seções das medições (b) mapa digitalizado no CFX

Nas Figs. 2 (a) e (b) são apresentadas as comparações dos resultados experimentais medidos por Taylor et al. (1983) com os resultados numéricos obtidos do presente modelo ao longo das linhas A-A e AA-AA, respectivamente. Estes resultados experimentais foram obtidos, segundo os autores, para uma atmosfera neutra, isto é, para números de Froude maiores que 1000. O aumento relativo da velocidade, mostrado nas figs. 1 (a) e (b) representa quantas vezes a velocidade local é maior que a velocidade de referência.

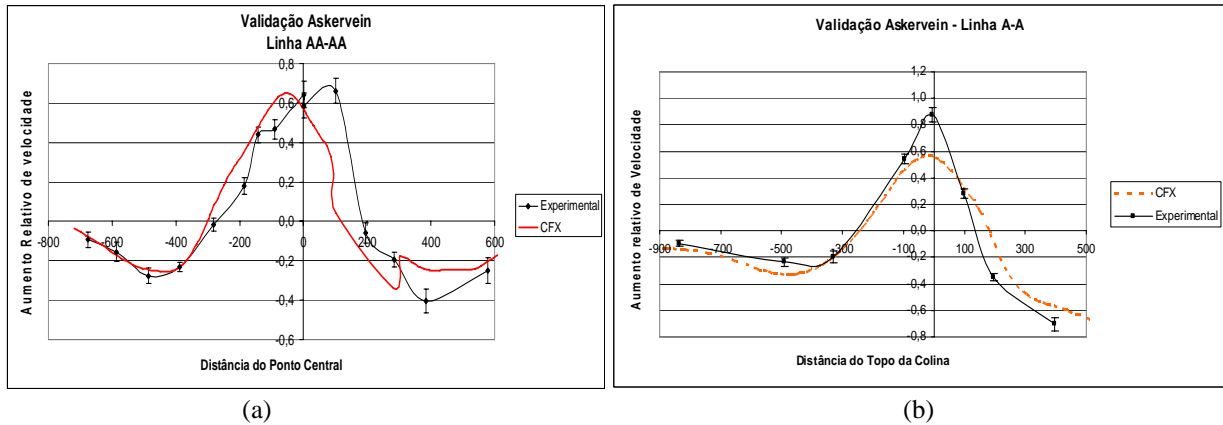


Figura 2 – Validação do modelo numérico. (a) ao longo da linha AA-AA (b) ao longo da linha A-A

Os resultados apresentados nas Figuras 2 (a) e (b) mostram que, mesmo com malha muito refinada, os resultados do modelo apresentam uma diferença considerável quando comparados com os dados experimentais. Por outro lado, o comportamento das duas curvas é semelhante, mostrando que o modelo comporta-se qualitativamente semelhante aos dados experimentais.

### 5. Resultados

Os resultados apresentados referem-se ao mapa topográfico das regiões de Belo Horizonte, Minas Gerais, Brasil. O escoamento de vento sobre a topologia de Belo Horizonte foi simulado utilizando as medias experimentais referentes ao ano de 2002. Foi utilizado um valor para a velocidade de referência igual a 5.1 m/s, que consiste no valor médio obtido por dados experimentais coletados pelos anemômetros localizadas nas posições V1 e V3, a 10 m do solo. A direção do vento foi definida como sendo 157,5°, que corresponde à direção média entre sul e sudoeste, predominante nos dados experimentais medidos. A Figura 3 apresenta a topologia no entorno dos pontos de medição. O valor de velocidade medido experimentalmente na posição V2 é comparado com o valor obtido numericamente para aquela coordenada.

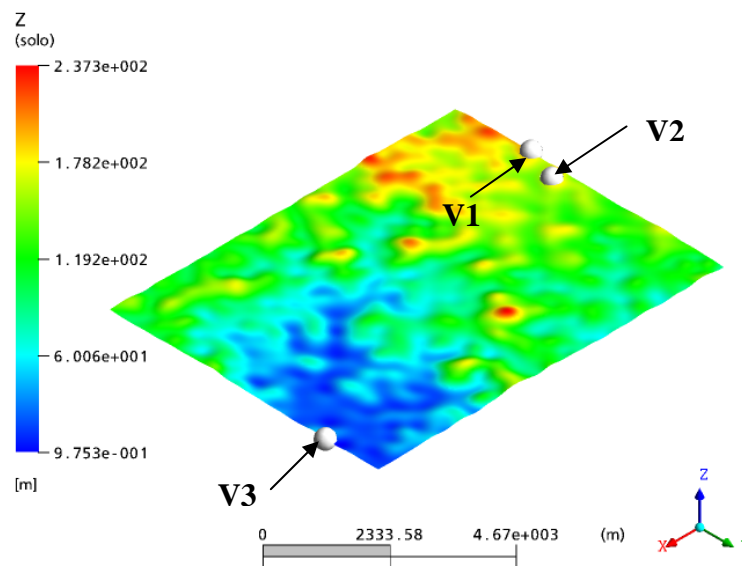


Figura 3 – Domínio de Belo Horizonte, pontos de medição experimentais

A Figura 4 mostra os perfis de velocidade para os pontos 1, 2, 3 da região de Belo Horizonte - MG obtidos pelo modelo matemático proposto. Nota-se que o ponto 3, por estar localizado em uma superfície mais baixa e plana, como mostrado na Figura 3, o perfil de velocidade não apresenta regiões de aceleração de velocidade, permanecendo um perfil mais parabólico. Já nos pontos 1 e 2, aparecem acelerações por se tratar de posições localizadas em regiões mais elevadas, como mostrado na figura 3 para alturas superiores. Estes perfis de velocidades foram obtidos para condições de contorno referentes aos valores médios de velocidade e direção predominante de ventos medidos nos pontos de fronteira a 10 m de altura do solo, nas posições mostradas no domínio da Figura 3.

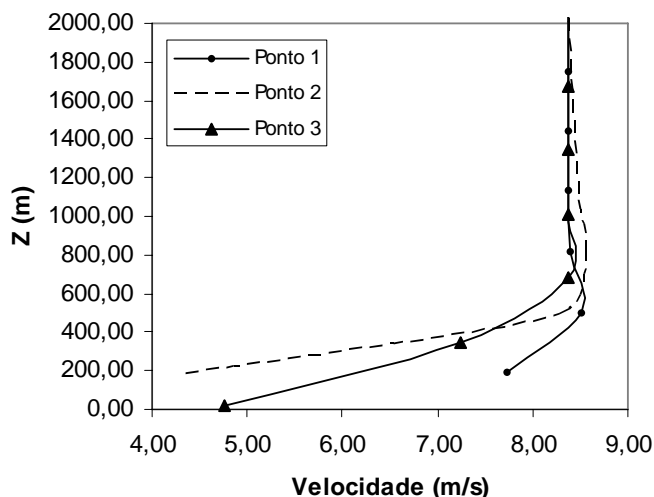


Figura 4 – Perfis de velocidade nos pontos V1, V2 e V3 da região de Belo Horizonte – MG.

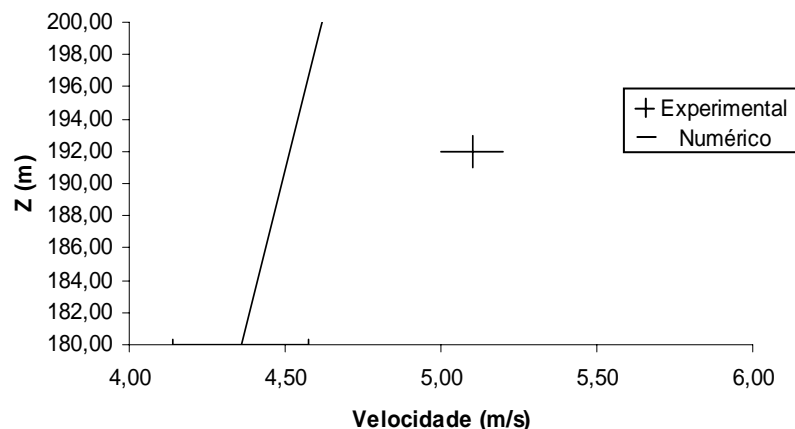


Figura 5 – Perfil de velocidade no ponto V2 e ponto da velocidade referencia de 5,1 (m/s) utilizada no modelo.

A Figura 5 mostra o perfil de velocidade obtido para o ponto 2, indicando o valor obtido numericamente a 10 m de altura do solo e o valor medido experimentalmente na mesma altura. Nesta figura, o menor valor da ordenada z representa a superfície do solo. O valor numérico obtido foi de 4,5 m/s, sendo que o valor medido experimentalmente foi de 5,1 m/s. Isto dá uma diferença de 11,76%, indicando que o modelo ainda necessita de maiores ajustes.

## 6. Conclusão

O modelo foi validado com dados de Askervein e dados medidos em campo na região de Belo Horizonte, mostrando que ainda necessita de maiores ajustes. No modelo proposto não foram considerados o efeitos térmicos e transientes. No entanto o modelo apresenta uma análise qualitativa satisfatória, visto que os resultados obtidos já representam o comportamento obtidos em estudos realizados em (Kim, 1998; Camilla, 1999; Taylor, 1985 e Uchida, 1999).

O modelo validado permite obter resultados mais confiáveis para qualquer topografia de terreno, sempre que se tomem os cuidados necessários com o tratamento das condições de contorno, tratamento dos termos fonte e cuidados com a malha utilizada.

A velocidade obtida do modelo no ponto 2 apresentou um erro de 11,76 % quando comparada com o valor medido, o que é bem aceitável, uma vez que o regime simulado é para uma situação de regime permanente e os dados medidos experimentalmente refletem uma situação de regime transiente.

Com base nestes dados a utilização de mapas de velocidades obtidas pelo modelo para diversas superfícies pode ser muito importante para implementação de novas linhas de transmissão e assim melhorar eficiências de projetos das linhas de transmissão

## 7. Referências

- Albertson, J.D. and Parlange, M.B., Natural Integration of Scalar Fluxes from Complex Terrain, *Advances in Water Resources*, vol. 23, p. 239-252, 1999.
- Camilla, G., Griff, J., Michel, O., and Vincent, D., Transport and Mixing in the Atmospheric Boundary Layer, *Lecture Notes, Aosta Summer School*, 1999
- Zhang, Y.Q., Arya, S.P., and Snyder, W.H., A comparison of numerical and physical modeling of stable atmospheric flow and dispersion around a cubical building, *Atmospheric Environment*, vol. 30, n. 8, pp. 1327-1345, 1996.
- AEA Technology plc. CFX-5.5 Documentation. AEA Technology Engineering Software Ltd, United Kingdom, 2002.
- Andreas, E.L., The atmospheric boundary layer over polar marine surfaces, US Army Corps of Engineers, 1996.
- Huser, A., Nilsen, P.J., and Skatun, H., Application of k- model to the stable ABL: pollution in complex terrain, *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 67-68, pp. 425-436, 1997.
- Kim, H.G, Lee, C.M., and Noh, Y.J., Numerical prediction of the wind flow over hilly terrain, 1998.
- Martins, M.A., Estimativa de erro e adaptatividade para métodos de volumes finitos em malhas triangulares não estruturadas, Tese de Doutorado, Departamento de Engenharia Mecânica, UFMG, Belo Horizonte, 2002.
- Montavon, C., Validation of a non-hydrostatic numerical model to simulate stratified wind fields over complex topography, *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 74-76, 1998, pp. 273-282., 1998.
- Taylor P.A. and Teunissen, H.W., Askervein 82: an initial report on the September/October 1982 experiment to study boundary layer flow over Askervein, South Uist, Scotland. In: Internal Report MSRB-83-8. Downsview. Ontario, Canada. 1983.
- Taylor P.A. and Teunissen, H.W., The Askervein Hill Project: Report on the September/October 1983 main field experiment. In: Internal Report MSRB-84-6. Downsview. Ontario, Canada. 1985.
- Uchida, T. and Onya, Y., Numerical Simulation of Atmospheric Flow Over Complex Terrain, *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 81, p. 283-293, 1999.

### NUMERICAL STUDY OF THE TURBULENT FLOW OVER COMPLEX TERRAIN

Gilberto Augusto Amado Moreira

UFMG – Universidade Federal de Minas Gerais, Departamento de Engenharia Mecânica, Av. Antônio Carlos, 6627 – Pampulha, 31270-901 – Belo Horizonte, MG, Brasil.  
gilbertomoreira@ufmg.br

Ramon Molina Valle

UFMG – Universidade Federal de Minas Gerais, Departamento de Engenharia Mecânica, Av. Antônio Carlos, 6627 – Pampulha, 31270-901 – Belo Horizonte, MG, Brasil.  
ramon@demec.ufmg.br

Marcio Arêdes Martins

UFV – Universidade Federal de Viçosa, Avenida P. H. Rolfs s/n - Campus - UFV CEP 36570-000 - Viçosa - MG  
aredes@ufv.br

Carlos Alexandre M. Nascimento

CEMIG – Cia. Energética de Minas Gerais, CEMIG - Cia. Energética de Minas Gerais, Avenida Barbacena 1200, Santo Agostinho, 30123-970 - Belo Horizonte, MG, Brasil.  
caxandre@cemig.com.br

#### Abstract

A numerical model is proposed in this paper to simulate the atmospheric boundary layer (ABL). The results obtained by the numerical model were validated using experimental data presented in the literature. The governing equations of the geophysical flows are the continuity, momentum and energy conservation equations. The momentum equations are coupled to the energy equation by an equation of state. Turbulence is considered in the model using RNG k- $\epsilon$  model. The irregularities of the terrain due to the vegetation are treated as an average rugosity. The computational domain is defined as an area far enough the interest region in order to guarantee the development of the wind flow from the inflow boundary edges. The top boundary of the domain is prescribed at least 500 m above the atmospheric boundary layer. At this altitude the wind flow is considered stable enough so that the Newman boundary condition can be applied. The results obtained can be used for further purposes, such as the determination of the swinging of cables isolation strings and to determine the ampacity of transmission cables. The results presented in this paper consist on the reference data of wind speed from Askervein topology, whose experimental results are used to validate the numerical model. Hence, results are presented for a topography of Belo Horizonte, in the state of Minas Gerais, Brazil.

Keywords: Atmospheric Boundary Layer 1, Numerical Model 2, Complex Terrain 3