Proceedings of the 11th Brazilian Congress of Thermal Sciences and Engineering -- ENCIT 2006
Braz. Soc. of Mechanical Sciences and Engineering -- ABCM, Curitiba, Brazil,- Dec. 5-8, 2006

Paper CIT06-0979

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 4th 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:

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

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:

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

where $f_{ap}^{(N)}$ represents the n^{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,

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

With the boundary conditions

$$f(0) = f_0$$
$$f'(0) = f_0'$$

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:

$$k_2 f^3 \cong k_2 f f_0^2 \tag{3}$$

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

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

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,

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

With the boundary conditions:

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

Equation (5) is valid for $x_N = N\Delta x < x < x_{N+1} = \left(N+1\right)\Delta x$. 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 4th order Runge-Kutta (RK4) method will be used and the results will be compared with the LSM approach. The 4th 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:

$$\frac{df(x)}{dx} = 1 - 4\,f(x)^2 \qquad f(0) = 0 \tag{6}$$

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:

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

The LSM furnishes the following linearized equation for each Δx step:

$$\frac{df_{N+1}(x)}{dx} = 1 - 4\,\frac{\left(f_N + f_{N+1}\right)}{2} f_{N+1}(x) \qquad f_{N+1}(N\Delta x) = f_N \tag{8}$$

The analytical solution of eq. (8) is:

$$f_{N+1}(x) = \frac{1 - e^{-4\frac{\left(f_N + f_{N+1}\right)}{2}(x+c)}}{4\frac{\left(f_N + f_{N+1}\right)}{2}} \quad \text{with } c = -N\Delta x - \ln\left(\left|\frac{1 - 4\frac{\left(f_N + f_{N+1}\right)}{2} f_N}{4\frac{\left(f_N + f_{N+1}\right)}{2}}\right|\right) \tag{9}$$

To calculate the function *f*(x) at $x = \left(N+1\right)\Delta x$, 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 estrange 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 Δx^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

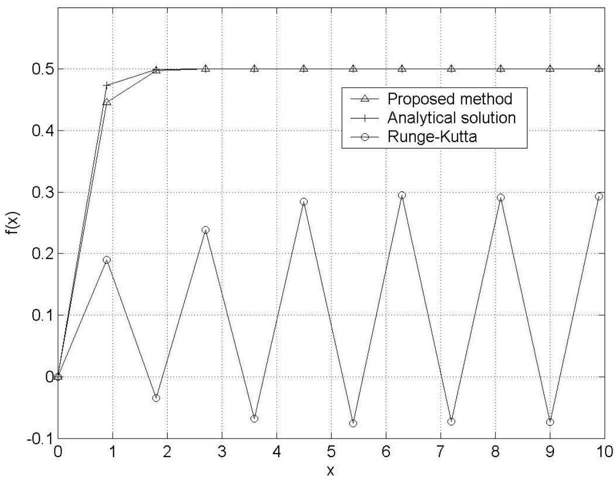

Figure 1 – Result obtained using a step $\Delta x = 0.9$. Observe that the Runge-Kutta method fails abruptly. The proposed method converges, even for this large step.

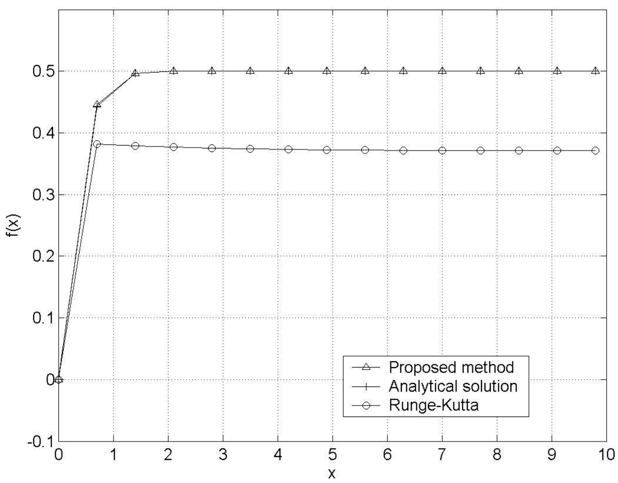

Figure 2 – Obtained result using a step $\Delta x = 0.7$. In this case the Runge-Kutta method fails.

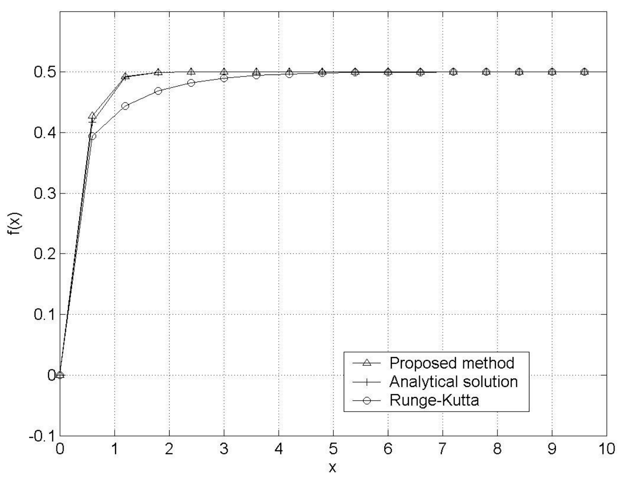

Figure 3 – Result obtained using a step $\Delta x = 0.6$. The Runge-Kutta method tends to get better.

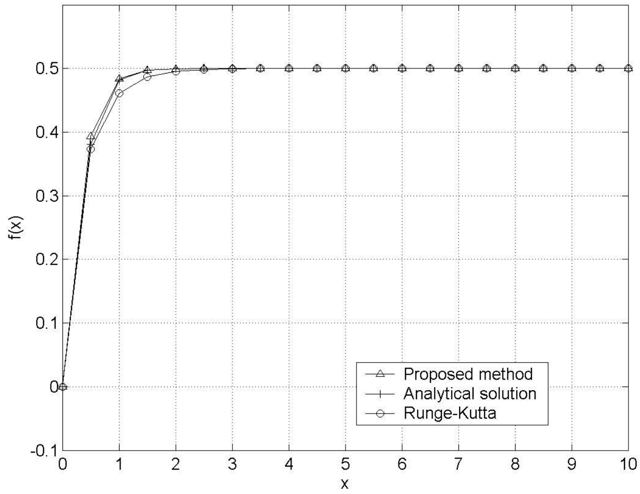

Figure 4 – Result obtained using a step $\Delta x = 0.5$. Here, the Runge-Kutta method presents good proximity of the analytical solution; however, the proposed method presents better convergence.

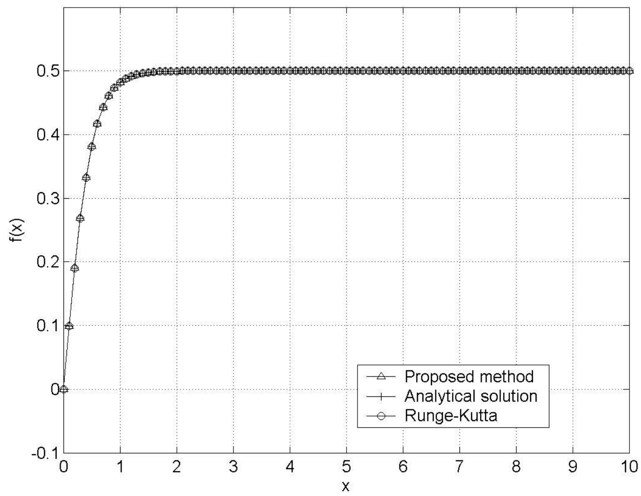

Figure 5 – Result obtained using a step $\Delta x = 0.1$. The solutions are practically coincident. The Runge-Kutta method has smaller error (see Table 1), but the proposed method presents a quite good approximation.

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 $\eta \to \infty$. 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:

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

Appling 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 $N\Delta\eta \le \eta \le (N+1)\Delta\eta$.

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:

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

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.

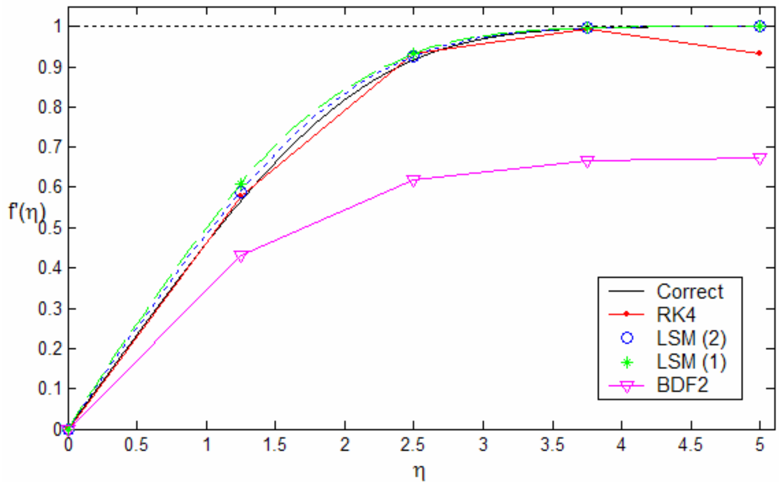

Figure 6 – Results obtained for a very coarse mesh

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^{nd} 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 $\eta \to \infty$;
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 $\eta \to \infty$. In fact, this behavior happens even when the mesh has only 2 points, as depicted in Fig. (7).

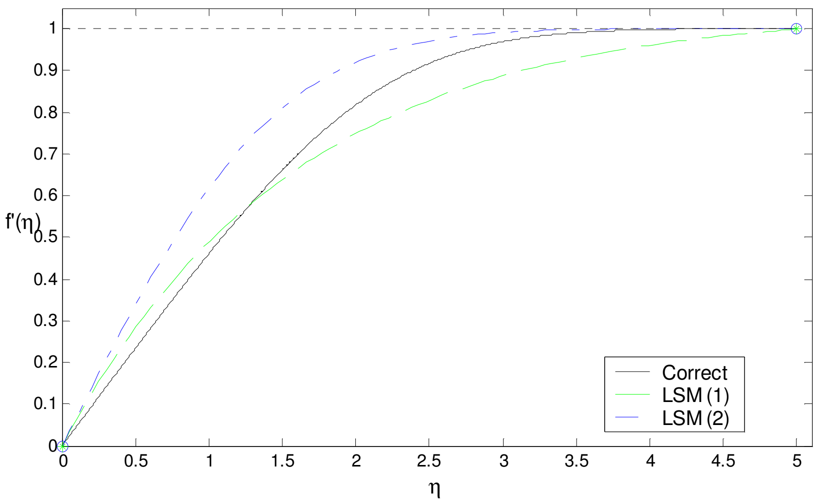

Figure 7 – Approximation using LSM with 1 step. The Runge Kutta does not give reasonable results.

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.

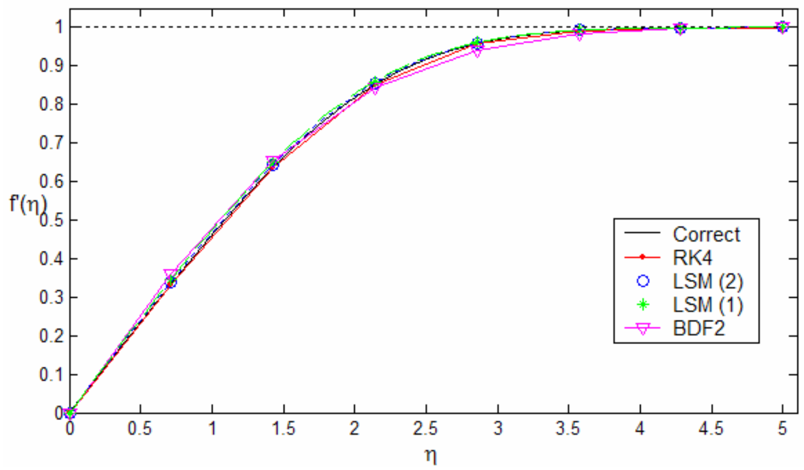

Figure 8 – Results obtained for a refined mesh

Figure (9) shows the relative error of LSM and RK4. It can be noted that the RK4 error presents a faster convergence. For coarses 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 $\Delta x \rightarrow 0$.

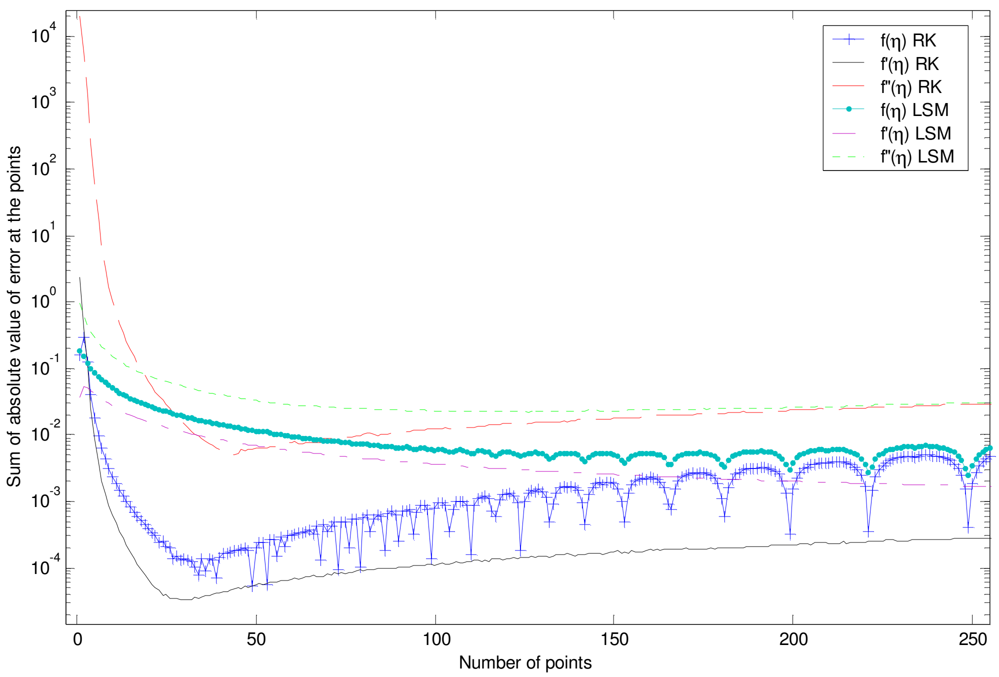

Figure 9 – Relative error of the methods

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^{th} 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