



DEVELOPMENT AND IMPLEMENTATION OF PROCESS CONTROL STRATEGIES FOR THE OPENFOAM CFD PACKAGE

Laissa Zanoni Duarte Medeiros*✉

laissa@peq.coppe.ufrj.br

Natalia Gomes Nogueira*✉

natalia@peq.coppe.ufrj.br

Jovani Luiz Favero†

jovani@peq.coppe.ufrj.br

Luiz Fernando Lopes Rodrigues Silva*

lflopes@eq.ufrj.br

UFRJ - Universidade Federal do Rio de Janeiro

Centro de Tecnologia - Ilha do Fundão

† Programa de Engenharia Química - COPPE, ✉ Actual Adress

* Departamento de Engenharia Química - Escola de Química

Abstract. *Industrial processes are highly dependent on the application of controllers in order to achieve the final products specifications. In addition, process simulation can bring informations on the dynamics of the variables and how it may affect the products such that the controllers may lead to better efficiency and profit. Usually, simulations using control strategies are based on spatially uniform models and only time variation is considered. The CFD (Computational Fluid Dynamics) simulation approach considers spatial variations such that control rules can be applied to a specific spatial location and a boundary condition used to modify the dynamics of the process. In this work, the development and implementation of different control strategies as a boundary condition was performed in the OpenFOAM open-source CFD package. In this case, the boundary condition is manipulated according to specific control rules in order to achieve the set-point for a variable of interest located in the spatial domain. The development is valid for any kind of scalar or vector variables also including the manipulation of the vector direction. The control strategies implementations can be purely proportional (P), proportional-integral (PI) or proportional-integral-derivative (PID). Simulations were accomplished in order to evaluate the dynamics and control performance for different test cases. The results showed that the implemented control techniques were able to efficiently reach the specified targets for the variable of interest through the manipulation of the request variables at the domain boundary. At last, tuning techniques were tested in order to obtain the proper controller parameters.*

Keywords: CFD, Control Systems, Vectorial Control, OpenFOAM

1. INTRODUCTION

Fluid flow is applied in every sense of daily life, as industrial processes or home appliances. In fact, the fluid in nature is not perfect mixed and there gradients involving mass and energy transfer. Several industrial processes suffer from bad property distribution, as plastic mold injection, ventilation safety, bread baking and more (Gerber *et al.*, 2006; Sun and Wang, 2010; Wong *et al.*, 2007). Therefore, it is important to control and manipulate the temperature or concentration distribution in order to achieve better performance and design (Lim *et al.*, 2010).

The Computational Fluid Dynamics (CFD) models have shown to be very useful to for analysis and design tool for problems involving fluid flow and related phenomena (Malalasekera and Versteeg, 1995; Patankar, 1980). In fact, the imperfectly mixed fluid dynamics can be characterized with good accuracy with CFD, providing the properties spatial and time profiles. In addition, this information can be used to propose geometric or operational modifications on the process in order to provide a better mixing. Still, the methodology is not appropriated to be used as on-line control due to the solution complexity and high computational effort. But the it could be used as virtual control systems to achieve the best operational condition.

Zerihun Desta *et al.* (2004) were the first to propose an approach to combine the CFD and process identification techniques in order to obtain a simplified model adequated to control purposes. The authors have simulated a non-isothermal flow using ANSYS CFX 4.3 (ANSYS Inc., 2012) and used the multivariable Transfer Function Matrix (TFM) to relate to Single-Input and Single-Output (SISO) obtained through the CFD simulation and achieve a simplified model. The results from the simplified model were very satisfactory and have shown that the research field combining CFD and

control techniques is promising.

Several works were done in the past few years improving the control application. Gerber *et al.* (2006) proposed a combination of the model predictive control (MPC) approach in a plastic mold injection process. The authors have used a Multiple-Input and Multiple-Output (MIMO) with three input temperatures to control the melting point in the geometry. Meng *et al.* (2009) used a Proportional-Integral rule to control the SISO temperature in a hot room process providing a good response. The success using the technique have encouraged other authors to apply CFD and control rules in different cases (Sun and Wang, 2010; Becker *et al.*, 2005; Lim *et al.*, 2010).

Nevertheless, all the past works have only dealt with scalar variables, as temperature or concentration. In this work, it is presented a SISO approach to manipulate vectorial variables using proportional, proportional-integral and proportional-integral-derivatives control rules implemented on an open source CFD package, OpenFOAM OpenCFD (2012). Test cases were performed in a non-isothermal flow in a chamber with heat sources. Both the temperature and velocity variables were controlled in specified points of the geometry presenting very good results. At last, different methods to obtain the controller parameters were successfully tested and evaluated.

2. METHODOLOGY

2.1 Transport phenomena

The governing equations for non-isothermal incompressible flows of Newtonian fluids are mass, momentum and energy conservation, respectively shown in Eqs. (1), (2) and (3).

$$\nabla \cdot (\mathbf{u}) = 0 \quad (1)$$

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \mathbf{u}) - \nabla \cdot \left\{ \nu_{eff} \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right] \right\} = -\nabla \bar{p} + \mathbf{g} \rho_i \quad (2)$$

$$\frac{\partial T}{\partial t} + \nabla \cdot (\mathbf{u} T) - \nabla \cdot (\kappa_{eff} \nabla T) = 0 \quad (3)$$

where \mathbf{u} is the velocity vector, \bar{p} is the kinematic pressure defined as p/ρ_o , where ρ_o is the density of the fluid, \mathbf{g} is the gravity acceleration vector and T is the temperature of the fluid (Bird *et al.*, 2006; Kundu and Cohen, 2004).

Using the Boussinesq approximation, the effective kinematic density is given by Ferziger and Peric (1997):

$$\rho_i = 1 - \beta(T - T_{ref}) \quad (4)$$

and the effective kinematic viscosity, ν_{eff} , and effective heat transfer coefficient, κ_{eff} , are given by:

$$\nu_{eff} = \nu + \nu_t, \quad \kappa_{eff} = \frac{\nu}{Pr} + \frac{\nu_t}{Pr_t} \quad (5)$$

where Pr and Pr_t are the Prandtl and turbulent Prandtl number, respectively, ν is the fluid laminar kinematic viscosity and ν_t is the turbulent kinematic viscosity obtained from an appropriate turbulence model. In this work, the well-known $k - \epsilon$ model was used (Launder and Spalding, 1974).

2.2 Control rules

Control systems are classified in open and closed loops. In the open loop control system, the controller is connected in line with the controlled process. Thus, the process input must be such that its output will behave as desired such that . Its main characteristic is that the control action is independent of the output. On the other hand, the closed loop control system uses a feedback control strategy, as shown in Figure 1. It performs an additional measure of the output error compared to the desired system response, making control faster, accurate and less sensitive to disturbances and discrepancies between the model and the real process.

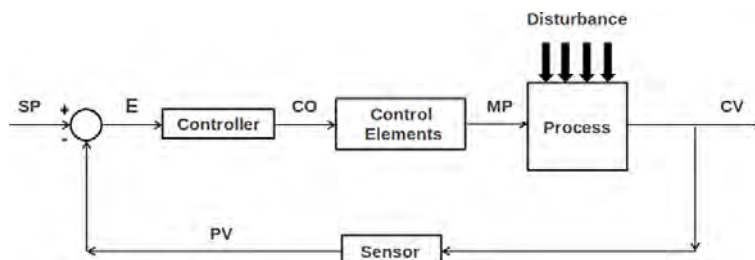


Figure 1. Feedback control closed loop system.

The controller purpose is receive the value of the process variable (PV) and compare it with a reference value (SP) to calculate an error measure (E), according to the Eq. (6). Considering to the calculated error, the controller must generate an output control signal (CO), correcting the manipulated variable (MV) value in order to reduce the error magnitude.

$$E(t) = SP(t) - PV(t) \quad (6)$$

The stability of the system is directly related to the choice of the controller type. Feedback controller types differ in the way which the controller output action is related with the detected error.

2.2.1 Proportional controller (P)

The proportional control is set as an action proportional to the measure of the error according to Eq. (7).

$$CO(t) = K_c E(t) + CO_0 \quad (7)$$

where $CO(t)$ is the control output, K_c is the proportionality constant, $E(t)$ is the error and CO_0 is the output control initially set.

The proportional gain control is the simplest control rule, with a single parameter to be adjusted. It has a relative disadvantage since the error can not be completely nullified, showing a permanent final error (off-set).

2.2.2 Proportional-integral controller (PI)

The proportional-integral control rule is defined as an action proportional to the error and its integral, as shown in Eq. (8), where τ_i is the integral time constant.

$$CO(t) = K_c \left(E(t) + \frac{1}{\tau_i} \int_0^t E(t)dt \right) + CO_0 \quad (8)$$

The great benefit of using this control action is to eliminate the error in permanent state. However, it may reduce the stability of the control loop.

For a constant value of K_c and high values of τ_i there is a predominance of the proportional action and, in this case, it shows the existence of an off-set. Reducing the τ_i value, the integral action should predominate over the proportional action and the error tends to decrease quickly. It is necessary to take care in decreasing excessively τ_i value since the response can be oscillatory in a tendency to instability as the controller tends to behave as a pure integrator.

2.2.3 Proportional-integral-derivative controller (PID)

The control action using the derivative control rule depends on the error rate of change, according to Eq. (9), where τ_d is the derivative time constant.

$$CO(t) = K_c \left(E(t) + \frac{1}{\tau_i} \int_0^t E(t)dt + \tau_d \frac{dE(t)}{dt} \right) + CO_0 \quad (9)$$

Combined with proportional-integral terms, the derivative action purpose is to anticipate the control action. Considering this, the process can react more quickly and, in this case, the control signal to be applied is proportional to a prediction of the process output.

The predictive action provides more stability to the system allowing to use higher values for the proportional constant K_c , providing a faster and more efficient response than the previous rules.

2.2.4 Controller tuning: reaction curve method

The controller tuning is performed changing the controller gain and evaluate the response of the output variables. In fact, it may lead to a tedious and non systematic procedure. Considering this, Ziegler and Nichols (1942) proposed two different methods of tuning, the sensitivity limit and the reaction curve methods, using experimental data to determine a set of initial parameters, K_c , τ_i and τ_d for PID controllers.

The tuning methodology is performed applying a step of known amplitude (A) in the process input operating in open-loop. After steady-state is reached, the reaction curve is obtained showing the response of the process along the time.

It is assumed that the process can be represented by a first order transfer function with time delay, as shown in Eq. (15), where K_p represents the proportionality constant of the process, τ_i the time constant and θ the time delay.

$$G_p(s) = \frac{K_p e^{-\theta s}}{\tau_p s + 1} \quad (10)$$

The response curve can be characterized by two parameters: the slope at the inflection point (α) and the time interval at which the tangent line intersects the t axis (Θ), as shown in Fig. 2.

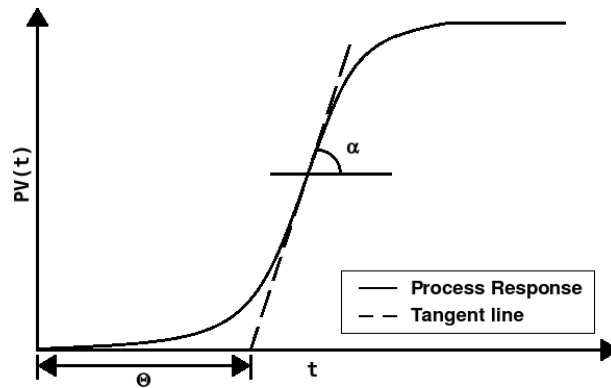


Figure 2. Response to first order process with time delay and identification of parameters α and Θ .

Ziegler and Nichols (1942) proposed empirical correlations to obtain the parameters that represent the response curve using Eq. (15). The tuning correlations are shown in Tab. 1 where A is the known amplitude.

Table 1. Tuning correlations defined by Ziegler and Nichols (1942) for the reaction curve method.

Control Type	K_c	τ_i	τ_d
P	$1.0 / (\Theta / \alpha^*)$	-	-
PI	$0.9 / (\Theta / \alpha^*)$	3.3Θ	-
PID	$1.2 / (\Theta / \alpha^*)$	2.0Θ	0.5Θ

⁽¹⁾ $\alpha^* = \alpha/A$

3. NUMERICAL PROCEDURE

3.1 Computational fluid dynamics

Computational Fluid Dynamics (CFD) consists in the use of numerical methods and algorithms to solve and analyze problems that involve fluid flows (Malalasekera and Versteeg, 1995). The properties prediction is accomplished through models based on conservation equations varying on space and time.

Several commercial and free CFD packages are available and the choice of use relies on the amplitude of physical problems it is capable to handle, its efficiency and robustness. Still, the commercial CFD packages has a closed software architecture approach and new implementations can be compromised.

Therefore, the OpenFOAM (Open Field Operation and Manipulation) CFD package (OpenCFD, 2012) emerges as a great option for solving a variety of flows problems, since this package is freely distributed and comes with its source code, which allows a deeper interaction with the user (Silva, 2008). It consists in a set of efficient and flexible modules written in C++ that can be used to build: (i) solvers for complex engineering problems involving tensorial fields, (ii) utilities for pre and post-processing data, (iii) libraries to be used by the solvers and utilities, such as libraries of physical models (Silva, 2008). At last, OpenFOAM numerical discretization is based on the finite volume method applied to polyhedral cells.

3.2 Control rule applied in CFD simulations

For each time step, the conservation equations are solved in each cell of the geometric mesh. From the variables obtained at the cell centroid, it is possible to calculate the surface values at the face of each cell using an interpolation scheme. Thus, the CFD solution provides the variable values for any discrete cell or cell face into the physical domain and all discrete time step.

The implementation of a feedback control strategy is based on the definition of a set-point, $SP(t)$, for a particular field located at a defined point or region inside the spatial domain. During the simulation, a process variable, $PV(t)$, is chosen and monitored in the controlled point or region in every time step. Using the defined set-point, the error $E(t)$ is calculated using Eq. (6) and the manipulated variable $MV(t)$ is modified according to the adopted control rule (P, PI or PID). This scheme is illustrated in Fig. 3 considering $PV(t)$ as a point into the domain and $MV(t)$ as a boundary condition.

The modification of the manipulated variable in each time step depends on the variable tensor order. In this work, the $PV(t)$ and $MV(t)$ variables can be scalars, such as temperature, concentration, etc. or vectors, as velocity, for instance.

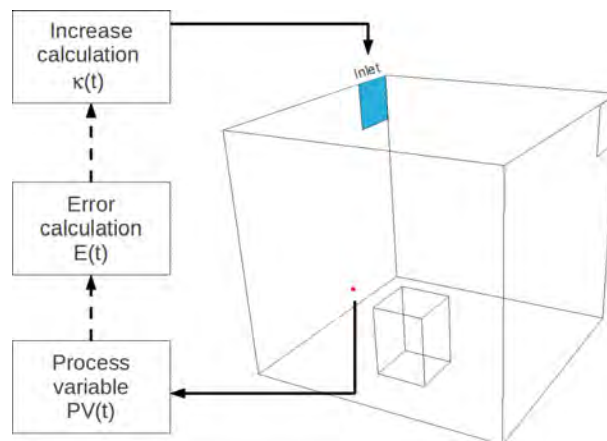


Figure 3. Feedback control strategy showing a sample monitoring point and an inlet region that contains the boundary condition to be manipulated.

3.2.1 Control rule for scalar variables

In this case, the error is calculated with Eq. (6) and used to provide a correction on a scalar variable value manipulating another scalar variable. The scalar correction should be based on control parameters tuned according to Eqs. (7), (8) and (9). Thus, the control action is applied for each time step or determined time intervals in order to correct the manipulated variables in the CFD simulation (Medeiros and Nogueira, 2012).

3.2.2 Control rule for vector variables

When the manipulated variable is a vector, it is possible to change the magnitude of the vector, keeping constant its direction, or change its components to vary the vector direction. For the latter case, considering the velocity vector for example, the flow through a surface would be kept constant changing only its direction.

- **Controlling vector magnitude:** The magnitude of a vector is a scalar variable. Thus, the control strategy is the same as for scalar variables, shown in Sec. 3.2.1 In this case, the control action is applied directly to the vector magnitude.
- **Controlling vector direction:** In order to control the vector direction, it is possible to define and manipulate the vector angles in relation to their origin considering a cartesian coordinate system used by the OpenFOAM CFD package. Since each vector angle is a scalar variable, the control methodology is the similar to those applied to scalar variables.

This work used two angles to obtain the vector direction. The first is defined as θ angle, corresponding to the vector projection at xy plane (vector $(x, y, 0)$). The second angle is defined as ψ which is the vector projection in the yz plane (vector $(0, y, z)$). The controlled angles located in Cartesian coordinate axis are shown in Fig. 4(a).

The scalar control strategy is applied for each vector angle and the result is converted to the Cartesian coordinate system used by the CFD simulation. Before the coordinate conversion, the corrected angles must be compared with maximum and minimum angles that they can assume. These angles limit the vector direction to a physical state defined by the user. So, the maximum and minimum angles are input parameters are obtained from the contoured surface as shown in Fig. 4(b).

In the next time step or time interval, the manipulated vector changes its direction considering the controlled angles corrections obtained according to the tuning controller. The manipulated vector must be located in a specified boundary condition. The control strategy for vector direction is shown in Fig. 5.

4. SIMULATION TESTS AND RESULTS

In order to evaluate the control strategy and its efficiency, a hot room simulation with an air inlet and outlet and two heat sources was performed. The geometry consisted in a room of 5.0 m width, 3.0 m depth and 3.0 m height. The air inlet with 1.5 m width and 0.2 m height and outlet with 2.0 m width and 0.2 m height. The inlet air direction was initially set with $\theta = 90^\circ$ and $\psi = 45^\circ$. At last, the two heat sources have each dimensions of 1.5 m width, 2.5 m depth and 2.0 m height. The geometry and boundaries are shown in Fig. 6.

In order to manipulate the inlet boundary, 6 monitor points were defined to be used as set-point localizations. The monitoring points were distributed in a strategic manner on the geometry and are detached in black in Fig. 6 and its

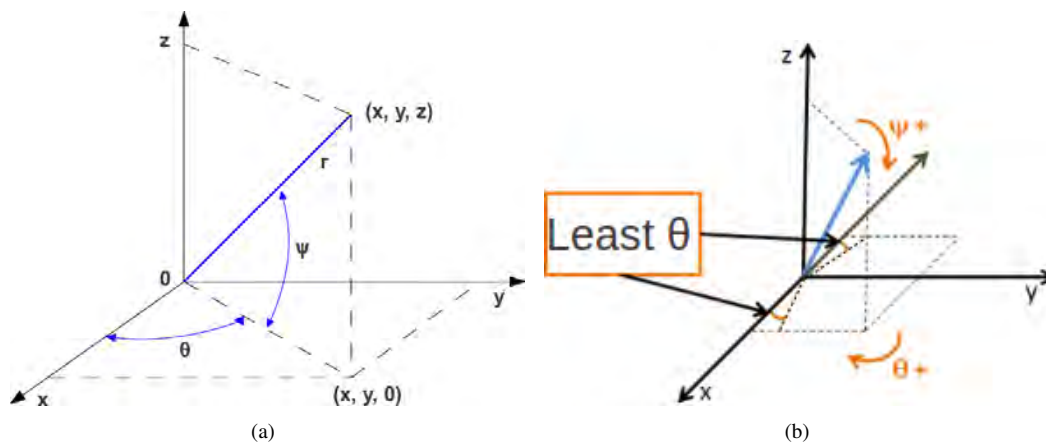


Figure 4. (a) Definition of θ and ψ angles and (b) illustration of angles θ and ψ for positive increments and a minimum θ angle from the contoured surface corresponding to xz plane.

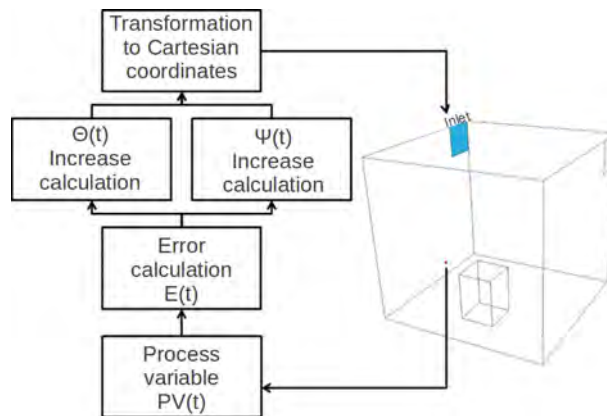


Figure 5. Feedback control strategy with a sample monitoring point and an inlet region that contains the boundary condition with the vector to be manipulated in θ and ψ angles.

positions are shown in Tab. 2.

Table 2. Position of monitoring points.

Monitoring Point	x	y	z
1	0.25	0.2	1.5
2	0.45	2.8	2.8
3	0.45	2.8	0.2
4	2.5	2.8	0.5
5	2.5	2.8	2.5
6	4.0	2.0	1.5

The non-isothermal flow was solved using Eqs. (1) to (3) considering the Boussinesq approximation. The adopted turbulence model was the $k-\epsilon$ (Launder and Spalding, 1974) and the gravity acceleration was $-9.81 \text{ m}^2/\text{s}$ in y direction. The simulation parameters, the schemes used for the discretization of temporal, advective and diffusive terms, as well as the physical properties of the air are shown in Tabs. 3 and 4, respectively.

In this work, two test cases were simulated. The first case aimed to control the temperature at a point near to a heat source manipulating the velocity vector direction at air inlet and maintaining a constant flow rate. The second case aimed at obtaining the parameters for controller tuning through the reaction curve method. For this, it was given a step perturbation in the inlet temperature, maintaining constants both velocity vector direction and magnitude. Then, the system response curve was constructed to obtain the parameters and, finally, the functionality of controllers tuned was validated with the parameters obtained.

The evaluation of the controlled variable at a monitor point was performed using a time average approach as

$$X_t = \frac{t - \Delta t}{t} X_{t-1} + \frac{\Delta t}{t} X_t \tag{11}$$

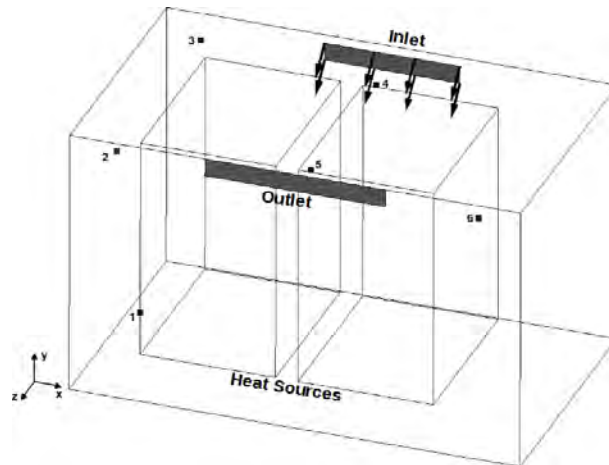


Figure 6. Geometry and boundaries considering initial velocity vector inlet with $\theta = 90^\circ$ and $\psi = 45^\circ$.

Table 3. Interpolation schemes used (OpenCFD, 2012).

Temporal Term	Implicit Euler
Advective Term (U,T)	High Resolution (linear Upwind)
Advective Term (k, ϵ)	First Order Upwind
Diffusive Term	Central Difference Scheme (CDS/linear)

where X_t and X_{t-1} corresponds to the variable value at the current and previous time step, respectively, and Δt is the time interval.

4.1 Mesh convergence study

The mesh convergence was performed to evaluate the velocity and temperature independence on the used mesh. Three hexahedral meshes were used and their information are described in Tab. 5. Mesh refinements near walls and heat sources were accomplished since it might occur high gradients for the variables.

The simulation was performed with air ingoing at a constant velocity of 3.24 m/s with initial angles $\theta = 90^\circ$ and $\psi = 45^\circ$ at the inlet patch. The time average variation of temperature and velocity in each one of the monitored points were evaluated according to Eq. (11).

The maximum relative error between meshes, in percentage, was calculated as:

$$\%ERM_{max} = \max_{j=1}^N \left(\frac{X_j^i - X_j^{ref}}{\max(|X^{ref}|)} \right) 100 \quad (12)$$

where X_j^i corresponds to the variable value, the superscript i corresponds to meshes 1 or 2 and the subscript j corresponds to each one of the monitoring points that range from 1 to N . X_j^{ref} corresponds to the reference value, that is, the value of the finest mesh.

The monitoring points with the largest velocity magnitude and temperature differences between meshes using the finest as reference are shown Figs. 7(a) and 7(b) respectively. The largest difference occurred at monitoring point 2 for the velocity and at point 5 for the temperature.

As observed in Fig. 7, it is clear that the largest differences occurred between the coarsest and finest meshes for both variables. The maximum relative error in time of meshes 1 and 2 related to 3 are shown in Tab. 6.

It was noticed that the velocity presented the mostly dependency on the mesh resolution as the temperature presented much smaller relative errors. It is also important to notice that the error is much smaller for both variables in mesh 2 than

Table 4. Air physical properties (Kreith and Bohn, 2011).

Laminar viscosity (ν) [m^2/s]	1.511×10^{-5}
Thermal expansion coefficient (β) [K^{-1}]	3.41×10^{-3}
Reference temperature (T_{ref}) [K]	293.15
Laminar Prandtl number (Pr)	0.7
Turbulent Prandtl number (Pr_t)	0.9

Table 5. Meshes refinement

Mesh	Number of Elements
Mesh 1	90 000
Mesh 2	150 000
Mesh 3	290 000

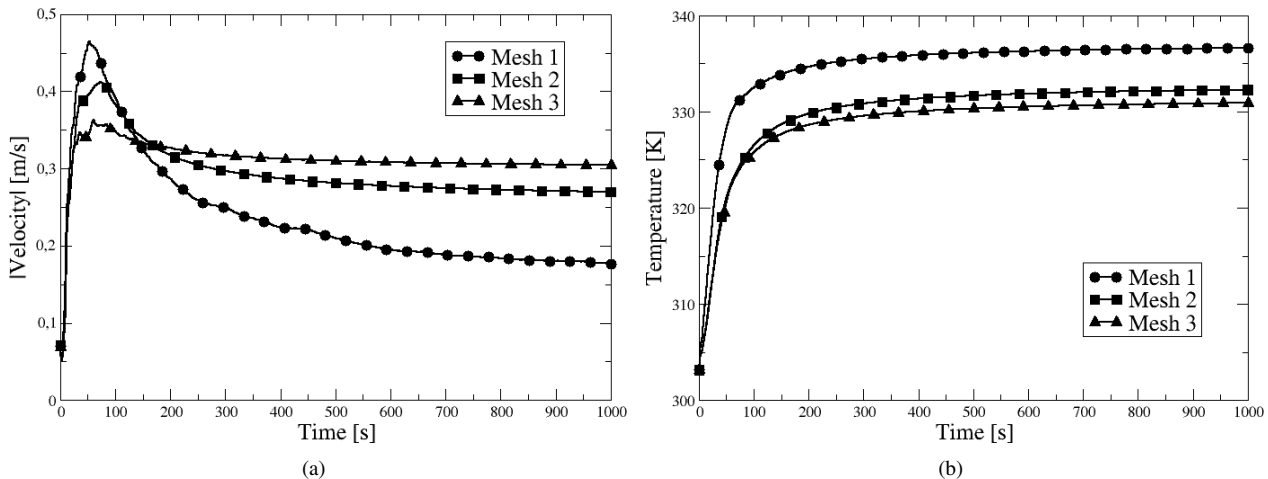


Figure 7. (a) Velocity magnitude profile in the monitoring point 2 and (b) temperature profile in the monitoring point 5.

1. Comparing the intermediate mesh with the finest mesh, the differences are not significant for the analysis performed in this work and, therefore, mesh 2 was used in the next simulations in order to attend the accuracy and computational cost goals.

4.2 Temperature Control through an inlet direction manipulation

This case analyzed a simulation with a proportional-integral-derivative controller applied to a monitored point located at (4.5 1.0 1.0) and acting every 1 second of simulation. The set-point was set to 291.15 K and the initial angles θ and ψ were both set to 30°.

The boundary condition for the temperature at the outlet was set as inlet-outlet, which means zero gradient in case of outflow or a fixed value of 303.15 K in case of inflow. It implies that, in case of inflow, the temperature inside the room will increase to values higher than the set-point. The PID parameters used to control θ and ψ angles and the settings of boundary conditions are shown at Tab. 7 and 8, respectively.

It is to be noticed that the controller parameters were obtained in a trial and error approach. So, many adjustments on the PID parameters were made during the simulation until the response was satisfactory and values shown in Tab. 7 were obtained. After that, the simulation was performed from the beginning using the obtained parameters.

The temperature profile at the monitor point is shown in Fig. 8. As can be seen, the PID controller was able to stabilize the system and reach the set-point value. Therefore, the implemented control obtained a satisfactory performance since the temperature was controlled at the selected point.

Fig. 9 shows the velocity vectors distribution at the instant which controller stability is reached, which occurs at the end of the simulation ($t = 2000$ s). As can be noticed, one of the heat sources is blocking the passage of air between the inlet and the monitoring point (detached in red in the figure). Instead of pointing directly to the monitoring point, the control action moved the velocity vectors to point toward the wall and provide a recirculating flow pattern and still reach the monitoring point.

Table 6. Relative error, in percentage, of meshes 1 and 2 in relation to mesh 3.

Mesh	Maximum relative error (%)		Minimum relative error (%)	
	T	U	T	U
1	5.788	54.512	0.886	0.501
2	0.975	11.469	0.095	2.239

Table 7. θ and ψ angles gains for the tuned PID controller.

Proportional Gain	Integral Gain	Derivative Gain
θ angle		
0.001	0.001	0.001
ψ angle		
0.00005	0.00005	0.00005

Table 8. Settings of boundary conditions.

Variables	Boundary Conditions		
	INLET	OUTLET	HEAT SOURCES 1 AND 2
U	Fixed Value ($0,972 \text{ m}^3/\text{s}$)	Zero Gradient	Fixed Value (0 m/s)
P	Zero Gradient	Fixed Value (0 Pa)	Buoyant Pressure (0 Pa)
T	Fixed Value (281.15 K)	Inlet-Outlet (303.15 K)	Fixed Value (6000 W/m^2)

4.3 Temperature control through an inlet temperature manipulation

4.3.1 Determination of controllers parameters

The inlet boundary condition for the air flowing with a constant velocity of 3.24 m/s with direction given according to Sec. 4.1 The temperature at the inlet was fixed at 288.15 K and a constant heat flux of 6000 W/m^2 in both heat sources were set.

The simulation was performed until the process reached steady state which occurred at the 1000 seconds. After that, a step of 10 K at the inlet temperature was applied and simulated until another steady state was reached. The system response curve at the monitoring point 6, shown in Fig. 6, was constructed and the reaction curve method for controller tuning was applied. From the response curve, the first order model with time delay was used to determine graphically the control parameters. The response curve at monitor point is shown in Fig. 10.

The proportional gain, K_p , was obtained using Eq. (13),

$$K_p = \frac{\Delta_{output}}{\Delta_{input}} = \frac{318 - 308}{10} = 1.0 \quad (13)$$

where Δ_{output} is the temperature difference after and before the step and Δ_{input} is the temperature step value.

For the response curve shown in Fig. 10, it can be noticed that the process did not change during the first 9 seconds after the step. so, this is the time delay value (Θ). The τ_p parameter corresponds to the time that response takes to reach 63.2 % of total variation. The corresponding temperature can, in this way, be calculated by Eq. (14).

$$T_{63.2\%} = T_{inicial} + 0.632 \Delta T = 308 + 0.632 \cdot (318 - 308) = 314.32 \text{ K} \quad (14)$$

The instant that the system takes to reach the temperature of 314.32 K can be determined graphically, which is approximately 73 seconds. Subtracting it from the time delay, τ_p is obtained as 64 seconds.

So far, the transfer function for the process at the monitoring point assessed can be approximated to first order model with time delay as shown by Eq. (15).

$$G_p(s) = \frac{1.0 e^{-9s}}{64s + 1} \quad (15)$$

For the application of controllers tuning method it was traced a tangent line to curve inflection point corresponding to the time average of a first order system. Thus, altogether with the time domain response at the monitored point for the temperature, Fig. 10 also shows the approximate first order model (Eq. 15), the time average and the tangent line. It can be seen that the approximate model provides a good representation of the time averaged temperature after the time delay. The tangent line was traced from the beginning of first order model approximation with a time interval of 0.01 seconds after the considered time delay. It can be considered a good approximation, given that the time interval between obtained points through simulation is considered very small and can represent the tangent. So, the angular coefficient and the inclination to inflection point (α) were determined according to Eqs. (16) and (17), respectively.

$$\alpha = \frac{\Delta T}{\Delta t} = \frac{308.03074 - 308.01537}{0.01} = 0.1537 \quad (16)$$

$$\arctan(0.1537) = 8.7387^\circ \quad (17)$$

Tab. 9 shows the controllers tuning parameters calculated according to Ziegler and Nichols (1942) correlations in Tab. 1.

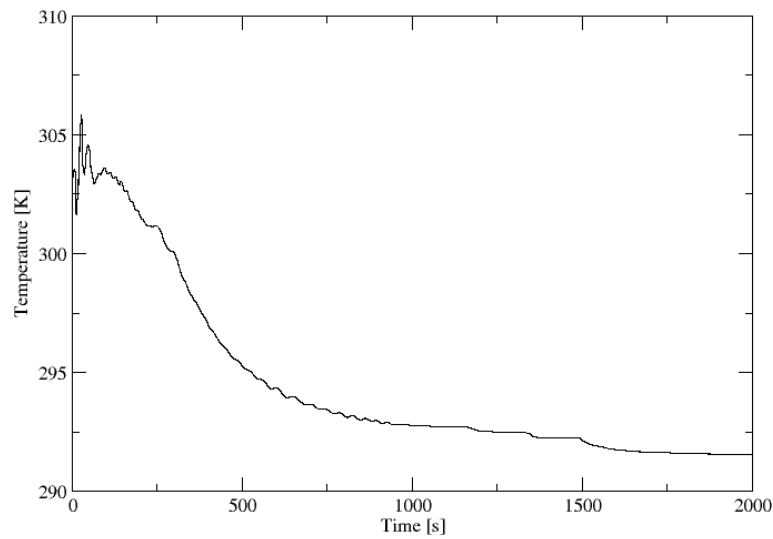


Figure 8. Temperature profile obtained by PID controller at the monitor point.

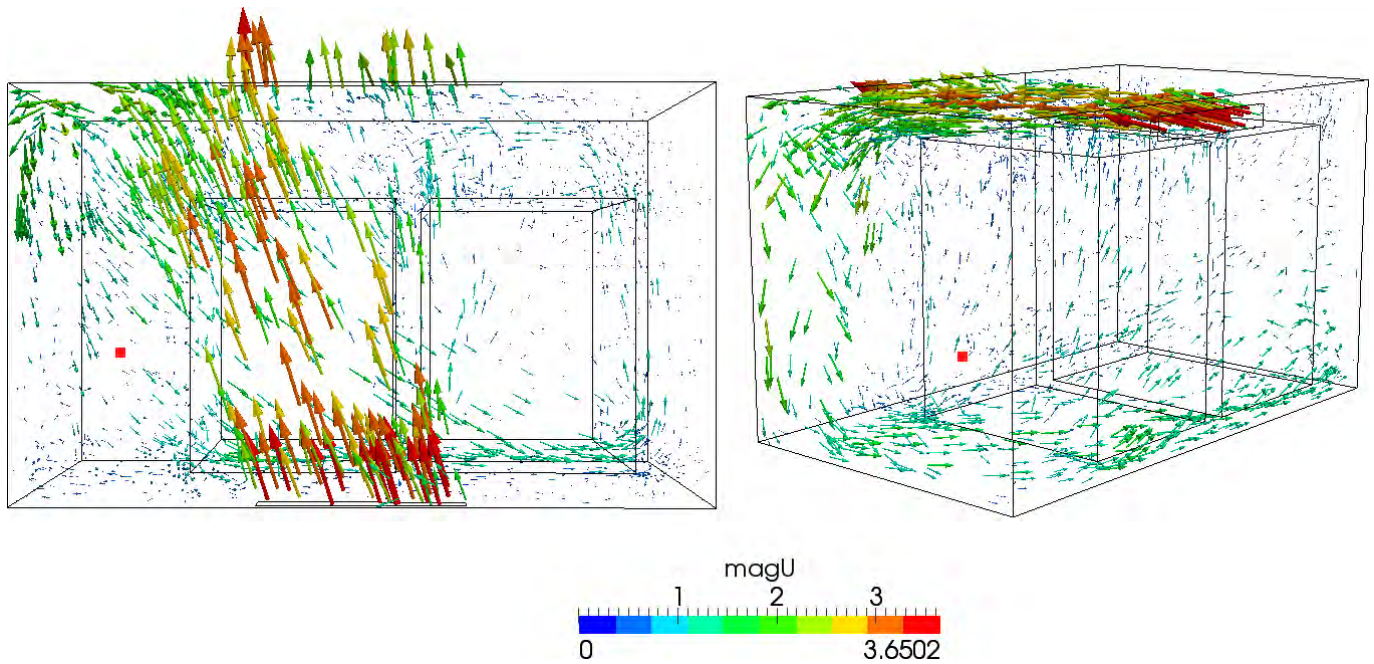


Figure 9. Velocity vectors and monitoring point in xz plane and in 3D view at a simulation time of 2000 seconds.

Table 9. Tuning correlations defined by Ziegler and Nichols (1942) for the reaction curve method.

Control Type	K_c	τ_i	τ_d
P	0.0971	-	-
PI	0.08739	0.002942	-
PID	0.1165	0.006472	0.5243

4.3.2 Evaluation of the tuned controllers

The parameters found by tuning method were then used in three different simulations, for proportional, proportional-integral and proportional-integral-derivative controllers. It was applied a step of 10 K in the temperature at simulation time equal to 1000 seconds. After that, the controllers were used to control the temperature set point by manipulating the temperature value in the inlet.

Fig. 11(a) shows the time average value of the temperature along the time for the uncontrolled and P, PI and PID controllers. The set-point value is represented by SP and it was fixed at 308 K. It can be seen that in case of using a proportional controller the temperature establishes in a value with a high offset. This is a typical result for proportional

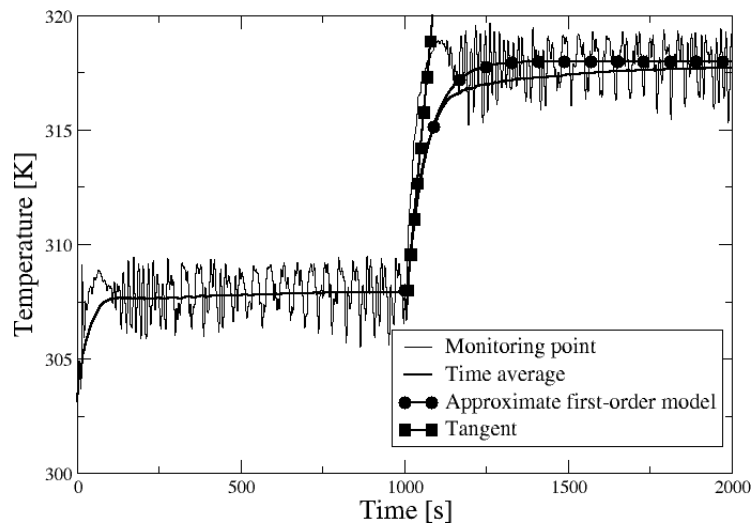


Figure 10. System response curve at monitoring point, approach to first order model with time delay and tangent line to inflection point.

controllers. On the other hand, the results of the temperature time average for both PI as PID are achieving the set point value, showing that the instantaneous temperature values must be around the set point. The zoom region around 3000 and 4000 seconds is seen in Fig. 11(b) and shows that for both PI and PID instantaneous temperature values are oscillating around the set point. Another important observation is that the PID controller is more robust and efficient for this case, showing a smaller overshoot and achieving the set point faster.

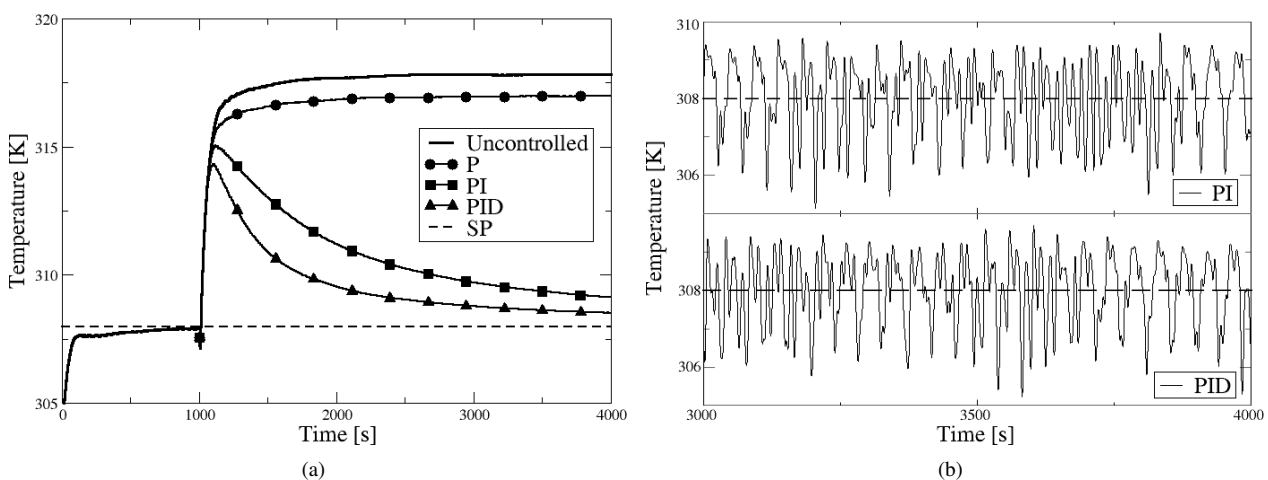


Figure 11. (a) Temperature time average for uncontrolled, proportional (P), proportional-integral (PI) and proportional-integral-derivative (PID) controllers and (b) instantaneous temperature values for the velocity vector direction manipulation.

5. Conclusion

This work presents the formulation and development of a new boundary condition, named “*manipulatedValue*”, implemented in the OpenFOAM CFD package in order to control any scalar or vectorial variable affected by spatial and temporal flow characteristics. This boundary condition consists of a feedback control model that can be configured, according to the user needs, to work as a proportional, proportional-integral and proportional-integral-derivative control types.

The boundary condition shows itself flexible by enabling adjustment of manipulated and monitoring variables, in accordance with the user purposes. The simulated cases showed that it is possible to control and to stabilize the temperature at strategic points in systems involving indoors air flows.

The technique presents very good results controlling the temperature manipulating the inlet temperature and velocity direction. In fact, it is shown that the control technique is able to manipulate the velocity direction providing a recirculation flow pattern in order to indirectly control the temperature in a specific location.

L Z D Medeiros, N G Nogueira, J L Favero and L F L R Silva
Development and implementation of process control strategies for the OpenFOAM CFD package

The control parameters were successfully obtained using the reaction curve method. The tuned controllers presented a good performance to control the temperature, as the PI and PID have shown the best results.

As a result, the numerical tool developed in this work proved useful to simulate and manipulate the process variables according to control rules and defined set points. In addition, the tool can be also used to determine functions relating system perturbations to its output variables and, through this, tuning the controller for complex flow systems.

6. REFERENCES

- Becker, R., Garwon, M., Gutknecht, C., Bärwolff, G. and King, R., 2005. “Robust control of separated shear flows in simulation and experiment”. *Journal of Process Control*, Vol. 15, No. 6, pp. 691–700.
- Bird, R.B., Stewart, W.E. and Lightfoot, E.N., 2006. *Transport Phenomena, Revised 2nd Edition*. John Wiley & Sons, Inc., 2nd edition.
- Ferziger, J.H. and Peric, M., 1997. *Computational Methods for Fluid Dynamics*. Springer.
- Gerber, A.G., Dubay, R. and Healy, A., 2006. “CFD-based predictive control of melt temperature in plastic injection molding”. *Applied Mathematical Modelling*, Vol. 30, No. 9, pp. 884–903.
- Kreith, F. and Bohn, M.S., 2011. *Principles of heat transfer*. Cengage Learning, Stamford, EUA, 7th edition.
- Kundu, P.K. and Cohen, I.M., 2004. *Fluid Mechanics*. Elsevier Academic Press, San Diego, CA, 3rd edition.
- Launder, B.E. and Spalding, D.B., 1974. “The numerical computation of turbulent flows”. *Computer Methods in Applied Mechanics and Engineering*, Vol. 3, pp. 269 – 289.
- Lim, S.G., Lee, S.H. and Han-Gon, K., 2010. “Benchmark and parametric study of a passive flow controller (fluidic device) for the development of optimal designs using a CFD code”. *Nuclear Engineering and Design*, Vol. 240, pp. 1139–1150.
- Malalasekera, W. and Versteeg, H.K., 1995. *An Introduction to Computational Fluid Dynamics (The Finite Volume Method)*. Thomson, NY, EUA.
- ANSYS Inc., 2012. “CFX and Fluent”. 13 Aug. 2012 <<http://www.ansys.com/>>.
- Medeiros, L.Z.D. and Nogueira, N.G., 2012. “Desenvolvimento e implementação de controle paralelo de variáveis vetoriais utilizando o pacote CFD OpenFOAM”. Technical report, Escola de Química/UFRJ, RJ, Brasil.
- Meng, Q., Wang, Y., Yan, X. and Li, Z., 2009. “CFD assisted modeling for control system design: A case study”. *Simulation Modelling Practice and Theory*, Vol. 17, No. 4, pp. 730–742.
- OpenCFD, 2012. *OpenFOAM: The Open Source CFD Toolkit User Guide*. OpenCFD Ltd.
- Patankar, S.V., 1980. *Numerical heat transfer and fluid flow*. Hemisphere Publishing Corporation.
- Silva, L.F.L.R., 2008. *Desenvolvimento de Metodologias para Simulação de Escoamentos Polidispersos Usando Código Livre*. Ph.D. thesis, COPPE/UFRJ, RJ, Brasil.
- Sun, Z. and Wang, S., 2010. “A CFD-based test method for control of indoor environment and space ventilation”. *Building and Environment*, Vol. 45, No. 6, pp. 1441–1447.
- Wong, S.Y., Zhou, W. and Hua, J., 2007. “Designing process controller for a continuous bread baking process based on CFD modelling”. *Journal of Food Engineering*, Vol. 81, No. 3, pp. 523–534.
- Zerihun Desta, T., Janssens, K., Brecht, V., Meyers, J., Baelmans, M. and Berckmans, D., 2004. “CFD for model-based controller development”. *Building and Environment*, Vol. 39, pp. 621 – 633.
- Ziegler, J.G. and Nichols, N.B., 1942. “Optimum settings for automatic controllers”. *Transactions of the A.S.M.E.*, Vol. 64.