IDENTIFICATION AND PRACTICAL CONTROL OF A FLUID PLANT

Paulo Ulisses Marques Liduário, paulo_liduario@hotmail.com [1]
Diego Colón, diego@lac.usp.br [1]
Átila Madureira Bueno, atila@sorocaba.unesp.br [2]
Eduardo Paciencia Godoy, epgodoy@sorocaba.unesp.br [2]
José Manoel Balthazar, jmbaltha@gmail.com [3]

[1]University of São Paulo (USP), Polytechnic School, Automation and Control Laboratory (LAC)
[2]São Paulo State University (UNESP), Sorocaba Campus, System Integration and Automation group (GASI)
[3]São Paulo State University (UNESP), Rio Claro Campus, Department Applied Mathematics and Computation (DEMAC)

Abstract: *In this paper, nonlinear models are obtained and closed loop controllers are designed for a didactical fluid plant in order to be used in process control undergraduate courses. This plant consists of a pvc piping network connecting two reservoir tanks. A proportional pump, which is the main actuator, forces (distilled) water through the piping system, which allows closed loop control of several physical variables, like water flow, temperature, water pressure and water level in one tank. There are four different (industrial standard) sensors for the variables above mentioned, that are: turbine sensor (water flow), pressure sensor, PT100 (temperature) and ultrasound (water level). Those sensors deliver voltage signal that are proportional to the measured variable, and an acquisition board discretize the signals in order to be used by the control algorithms. This board is also used to deliver control signals to the actuator. The controllers are implemented in a personal computer, by using the software MATLAB/Simulink. Industrial control techniques are tested, and the dynamics can be varied by opening and closing the manual valves in the piping network. The mathematical models of the system components (tank, pump, valve, to name a few) are experimentally determined (that is, identified) in order to help the controller design.Those controllers are used in the real plant, and the results are evaluated. Future improvement directions are also pointed out.*

Keywords: *Piping nerwork, Mathematical modeling, Controller implementation.*

1. INTRODUCTION

The practice of process control, despite the sophisticated control algorithms that has been developed in recent years, is heavily based in classical control techniques, like PID control (proportional + integral + derivative). The design methodologies used for such controllers are frequently *tuning* methods, like Cohen-Coon, Ziegler-Nichols and others (Seborg, 2004), that are simplified methodologies (not based in mathematical models). Also, several other devices/algorithms must be added to the PID controller in order to have a functional implementation, like saturations, filters and anti-windup mechanisms. Those devices (that are frequently software routines) transform the PID into a nonlinear controller, and frequently needs a fine tuning in the startup process. In many practical applications, mathematical models are very difficult to obtain, like in chemical processes involving pH control (Seborg, 2004). In some cases, on the other hand, it is possible to determine mathematical models via simple system identification techniques in order to aid in the controller design and validation processes, via simulation in appropriate software (Ljung, 1999). The controllers designed and validated in that way frequently need only minor *ad hoc* modifications (in the startup), which save precious time and resources.

In this work, the plant to be controlled is the *Process Control Compact Workstation* by FESTO (Helmich, 2004) which is in the Control Laboratory at the São Paulo State University (UNESP) – Sorocaba Campus, and has didactical purposes. In Figure 1, it is shown an image of the didactical plant (at left), and a schematic diagram (with sensors and actuators) at right, according to DIN 19227. There are four controlled variables: 1) water level in a tank, 2) water flow, 3) water pressure and 4) water temperature. Each variable is controlled separately (one at a time) by changing the piping configuration, and the control objective, for all the four variables, is to follow a reference signal, despite possible disturbances. The controllers used are PID controllers, but saturations, filters and anti-windup mechanisms are added. A centrifugal pump, represented as P101 in the schematic diagram, is the systems main actuator, and can be adjusted to function as an ON/OFF or as a proportional actuator (that is, the pump's power can be continuously varied). There is a set of manual valves, represented by the suffix 'V', which can be combined in order to change the fluid's route in the system (piping network). There are two water reservoir tanks (acrylic), and in tank B101 there are switch level sensors (that is, LS+/101.3 and LS-/101.2) to detect critical conditions (flooding or low level). A low level of fluid in the tank indicates that the pump will suck air, and must be turned off. In tank B102, there is also a contact level sensor for the same reason (LS-/102.2), but no high level sensor. The pump P101 can also force the fluid in a pressure reservoir B103 in order to control the pressure (depending on the valves' configuration) and this reservoir can contain water or air. The pressure is measured by a piezo resistive sensor. The water temperature is measured by a PT100 sensor (temperature-dependent resistance), represented by the symbol TIC104. The water heating element (secondary actuator) is represented by the symbol E104.

In order to configure the plant to control the level of tank B102 or the fluid flow through the pipe, one has to open V101 and close V103, and the pump P101 must be used as the actuator. The water level in tank B102 is measured by the ultra-sonic sensor LIC(102.1) and the flow is measured by turbine sensor FIC(101.1). For pressure and temperature control, V101, V104 and V107 must be closed and V103 and V108 must be open. In pressure control, the pump must also be used as actuator, but there is no fluid circulation in this case. In temperature control, the heating element E104 and the sensor TIC(104) are used, and the pump circulates water in the tank B101.

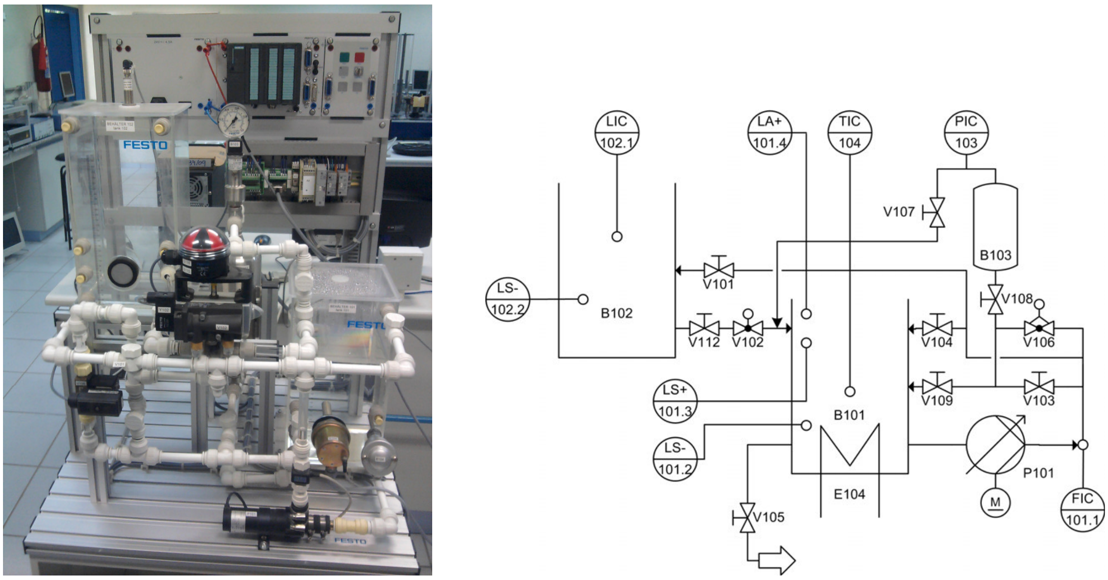

Figure 1. Photography and Schematic Diagram of the Plant (Helmich, 2004) .

It is also presented the mathematical modeling and closed-loop controllers designed for the plant described above. The mathematical models obtained, as well as the designed controllers, will be used for didactical purposes in process control courses at this university. Part of the experimental apparatus (more specifically, the cables and amplifier circuits), developed for this work, will also be used to teach technological disciplines related to process control. Other works have determined the mathematical model of this system, like (Keyser *et al.*, 2010), but the methods applied here are mainly experimental, and PID control techniques are effectively implemented and verified here. The paper is divided as follow: in section 2, the mathematical model for the system, for the several controlled variables, are presented. In section 3, the controllers' designs are presented, as well as the simulations for the controllers in linear region of operation. In section 4, some details of the controllers' implementation, as well as evaluation of the performance in nonlinear operation are presented, and in section 5, conclusions and suggestions for future work are presented. This paper was presented in the VIII Congresso de Engenharia Mecânica (Liduário, 2014).

2. MATHEMATICAL MODELING

2.1. Liquid Level Plant Modeling

The plant's configuration used in the water level control was achieved by closing valve V101and leaving V112 and V102 open. The tank B102 was filled up to a certain level, and level measurements in function of time was sensed by the ultra-sonic sensor LIC, and registered by MATLAB via acquisition board. By using Bernoulli equation (Garcia, 1992), (Ogata, 2010), (Dorf and Bishop, 2010), it can be shown that the relation between flow and pressure drop in a valve involves square roots, and after some substitutions, the differential equation for h (fluid level) is obtained, as shown in Eq. (1):

$$S\frac{dh}{dt}+q_e+q_i=S\frac{dh}{dt}+C\sqrt{h}=0 \quad (1)$$

where C is a constant, q_e is the output flow, q_i is the input flow and S is the tank's base area. It was determined, by direct measurement, that S = 332.5 cm³. The initial water level was h_{s0} = 20.337cm. As the input flow q_i is zero, an analytic/explicit solution can be found for Eq. (1) by direct integration, and results in:

$$\int \frac{1}{\sqrt{h}} \times dh = -\frac{C}{S} \times dt \rightarrow \sqrt{h} + \sqrt{h_0} = \frac{C}{2S}(t+t_0) \quad (2)$$

This curve was adjusted to the level measurements by using MATLAB's fit library, and resulted in C = 6.032 $cm^{5/2}/s$.

2.2. Water Flow Plant Modeling

The water flow plant model was determined experimentally by applying different voltage steps in the pump and registering the resulting flow (the measurements of sensor FIC 101.1) in function of time. The flow in this sensor is exactly the flow q_i in the input of the tank. In order to determine the static nonlinear water flow model, the final values of the (step response) flows were measured and they are presented in Tab. 1. A polynomial function was interpolated to the data presented in Tab.1, and its formula is shown in Eq. (3). This static model must then be associated in series with the dynamic part of the model.

Table 1. Different flows for different step voltages in the pump.

Voltage (V)	4.5	5.5	6.5	7.5	8.5	9.5
Flow (L/min)	2.308407	2.82016	3.350387	3.796177	4.390914	5.329801

$$q_i = 0.0257V^3 - 0.4995V^2 + 3.660V - 6.427 \quad (3)$$

The above mentioned dynamic part was approximated by a first order transfer function, with time constant equals to one second (value determined experimentally). This time constant corresponds to the time necessary to the flow reach 63,2% of the final value (Ogata, 2010), (Dorf and Bishop, 2010),. As the tank's level influences the result, this experiment was done for different levels for sensor LIC 102.1 and a mean valued was determined.

2.3. Water Pressure Plant Model

After a plant reconfiguration, as described above, it was applied a voltage step of 10V in the pump, which produced the pressure response shown in Fig. 2 (in fact, this is the voltage signal delivered by the sensor, which is proportional to the pressure). It is clear that this signal can be approximated by a first order system, and the time constant obtained (time to reach 63.2 % of final value) was approximately 5.0 seconds.

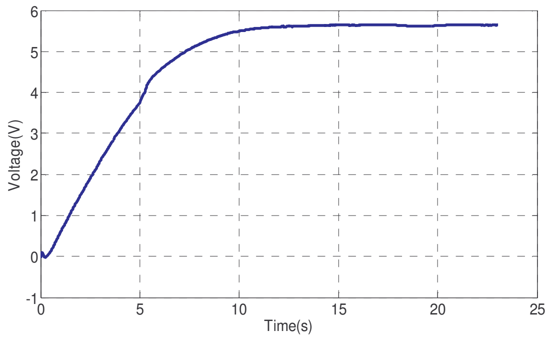

Figure 2. Pressure sensor response (in Volts) for a step of 10 Volts in the pump.

Different step voltages were applied in the pump, and the final value of the corresponding pressure were measured and presented in Tab. 2.

Table 2. Pressure in the reservoir for different voltages in the pump.

Voltage (V)	4.5	5.5	6.5	7.5	8.5	9.5
Pressure (bar)	0.033	0.0495	0.09	0.125	0.155	0.24

A polynomial function was interpolated to the data in Tab. 2 and the resulting function, which constitutes the static part of the model, is presented in Eq. (4).

$$p(V)=0.0054V^2-0.036V+0.085 \tag{4}$$

2.4. Water Temperature Plant Model

The temperature control is only possible by means of the water heating resistance E104, which can be only in two states: ON or OF. Despite the fact that a PWM switching could be used in this resistance (which would allow continuous control techniques), a simple ON/OFF controller was designed in order to avoid a considerable change in the plant's electrical wiring. No model was determined for temperature control.

3. CONTROLLERS' DESIGN

The control technique used in this work (except for the temperature control) was the PID control, which has the general transfer function shown in Eq. (5) below:

$$H(s)=K_p+\frac{K_i}{s}+\frac{K_d s}{T_f s+1} \tag{5}$$

This transfer function is called *realizable* PID, as it has the same number of poles and zeros (Ogata, 2010). It can be said that the derivative part $K_d s$ has a filter in series (with a pole in $-1/T_f$) that filters the noise amplified by this derivative part. In general, this filter's pole is put far in the left of the complex plane (fast pole), that does not change radically the traditional (non-realizable) configuration of the PID (with $T_f=0$). On the other hand, care must be taken in order to avoid closed loop instability. The other constants are: K_p: proportional gain, K_i: integral gain and K_d: derivative gain. The tool used to design those gains' values was the MATLAB/Sisotool, which needed the plant's mathematical model. The closed loop poles were chosen with the aid of the classical visual tools, like root locus, Bode diagrams, Nichols charts and others (Ogata, 2010) (Dorf and Bishop, 2010), that are part of the MATLAB/Sisotool resources.

3.1. Water Level PID Design

The models in Eq. (1) and in section 2.2 were combined, resulting in the nonlinear model presented in Eq. (6) below:

$$\begin{cases} \dot{q} = \dfrac{f(v)+q}{\tau} = \beta \\ \dot{h} = -\dfrac{C}{S}\sqrt{h} + \dfrac{q}{S} = \alpha \end{cases} \tag{6}$$

In order to apply the PID design methodology, an equilibrium point must be chosen, and Eq. (6) must be linearized (by the jacobian linearization process) around this point (Ogata, 2010). The chosen value for the equilibrium level was h* = 12.0 cm, which corresponds to a pump's equilibrium flow equals to q*=1.2 L/min and a pump's equilibrium voltage equals to v*=3.7 V. The function f(v) is the one in Eq. (3) and τ is the time constant determined in the same section, describing the pump's dynamics of first order. The linearization process was conducted in state space form. The linear model has the general form presented in Eq. (7), and the variables are all variations around the equilibrium values (represented with a Δ at left).

$$\begin{cases} \begin{bmatrix} \dot{\Delta q} \\ \dot{\Delta h} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial\beta}{\partial q} & \dfrac{\partial\beta}{\partial h} \\ \dfrac{\partial\alpha}{\partial q} & \dfrac{\partial\alpha}{\partial h} \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta h \end{bmatrix} + \begin{bmatrix} \dfrac{\partial\beta}{\partial v} \\ \dfrac{\partial\beta}{\partial v} \end{bmatrix} \Delta v \\ \Delta h = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta h \end{bmatrix} \end{cases} \tag{7}$$

The transfer function relating the variation of pump's voltage (around the equilibrium voltage) and the level variation (around h*) can be calculated from the system in Eq. (7) by the formula in Eq. (8) below (Ogata, 2005):

$$\frac{\Delta H}{\Delta V}=C(sI-A)^{-1}B+D \tag{8}$$

where A is the square jacobian matrix in Eq. (7), B is the 2x1 jacobian matrix, and C is the 1x2 zero/one matrix (with D equals to zero), The values of the partial derivatives in the matrices must then be calculated around the equilibrium point, and the resulting transfer function is:

$$\frac{\Delta H}{\Delta V}=\frac{0.05787}{s^2+1.003s+0.002618} \tag{9}$$

After a test of the implementation in MATLAB, it was verified that a high level of noise would be present (due to the sensor noise). In order to reduce this noise, a first order low pass filter (LPF) was designed with a cutoff frequency of about 2.0 Hz. This filter was put in series with the input (sensor) signal block in the Simulink diagram, which means that the filter is in series with the plant. The PID controller should then be designed by including the filter in the plant's transfer function in Eq. (9), as its pole can affect significantly the closed-loop stability. So the plant's transfer function used in MATLAB/Sisotool was the one shown in Eq. (10) below:

$$G(s)=\frac{0.1157}{s^3+3.002s^2+2.008s+0.005236} \tag{10}$$

The following values for the PID controller were found by using MATLAB/Sisotool (after several unsuccessful combinations): K_p = 10.5; K_i = 5; K_d = 8.07; T_f = 0.05. After that, simulations with the model and the controller (in closed loop) were realized, and in Fig. 3, it is shown the level response (in red), the reference input (in blue) and the voltage applied to the pump (control signal, in green). The reference signal is not a perfect step, as in practical situations, the derivative control cause a surge voltage at the beginning.

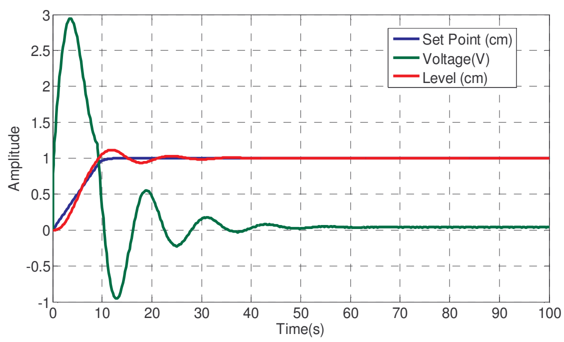

Figure 3. Level response and control signal for a step of 1 cm as reference.

The closed-loop overshoot is about 15%, and the settling time is about 35s. The voltage in the pump was maintained less that 3 Volts (the maximum is 10 V).

3.2. Flow PID Design

In order to design the flow control system, the linearization of the model was done around the equilibrium flux q*=1.2 L/min, that corresponds to the pump's voltage v*=3.7 V (the same values of the preceding section). The corresponding transfer function can be found from Eq. (7), by selecting a different output, like in Eq. (11), and the resulting transfer function is presented in Eq. (12). With an additional LPF (to filter the noise), the transfer function used in the design process is shown in Eq. (13).

$$[\Delta q]=[1\ 0]\begin{bmatrix}\Delta q\\ \Delta h\end{bmatrix} \tag{11}$$

$$\frac{\Delta Q}{\Delta V}=\frac{0.05787}{s+0.002618} \tag{12}$$

$$G(s)=\frac{0.05787}{0.5s^2+1.001s+0.002618} \tag{13}$$

The PID controller determined by using MATLAB/Sisotool (which gave the best results) has the following parameters values: K_p = 4.155, K_i = 3.172, K_d = 1.3253, T_f = 0.05. The closed-loop system response, obtained by simulation, is shown in Fig. 4 (with the same colors as in the preceding figure).

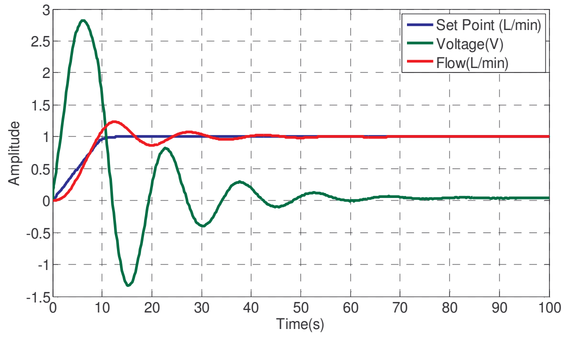

Figure 4. Flow response and control signal for a step of 1 L/min as reference.

3.3. Water pressure PID design

The pressure mathematical model (with static and dynamic part), determined in section 2.3, is reproduced in Eq. (14) below, in form of a differential-algebraic equation:

$$\begin{cases} w(V)=0.0054V^2-0.036V+0.085 \\ w(v)=5\dot{p}+p \end{cases} \tag{14}$$

The linearization was done around the equilibrium pressure p* = 0.1 bar, which correspond to a voltage about v* =7.0 V. The linearized model is presented in Eq. (15):

$$\begin{cases} \Delta\dot{p}=\left[-\frac{1}{2}\right]\Delta p+\left[\frac{\partial \frac{w(v)}{2}}{\partial v}\right]\Delta v \\ \Delta p=[1]\Delta p \end{cases} \tag{15}$$

and the corresponding transfer function, obtained by using the formula in Eq. (7), is:

$$\frac{\Delta P}{\Delta V}=\frac{0,0396}{5s+1} \tag{16}$$

The corresponding PID controller, obtained also with the aid of MATLAB/Sisotool, has the following parameters: $K_p = 47$; K_i 50; $K_d = 15$; $T_f = 0.05$. The pressure step response, obtained by simulation, is presented in Fig. 5, where it can be seen the pump's input voltage (applied by the PID controller) and the corresponding output response (with the reference). One can see an overshoot of less than 15% and a settling time of about 25 seconds. The limit voltage of 10 V was respected in the simulation. Two different figures were generated due to the difference in the graphic scales.

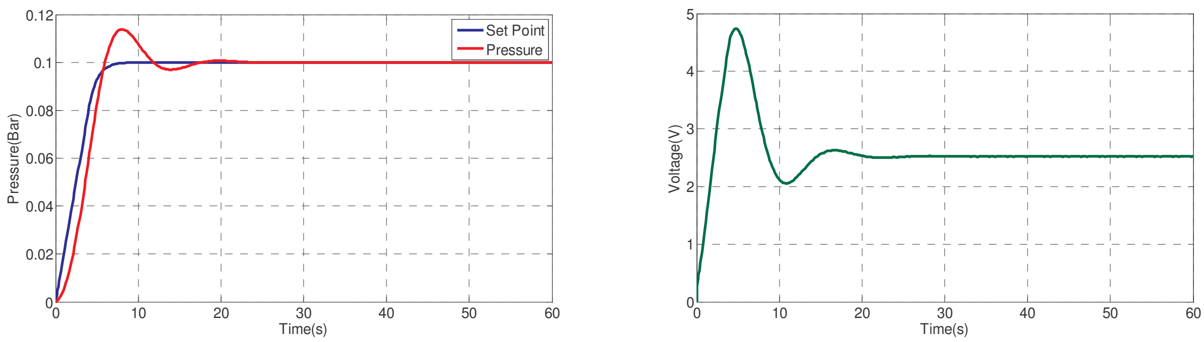

Figure 5. Pressure and control signal response for a step of 0.1 bar as reference.

3.4. ON/OFF Temperature controller design.

The ON/OFF controller was design by using the ON/OFF methodology, which is a hybrid system control technique. The block diagram of the controller uses logic and relational operations, as shown in Fig. 6. There are two limits that, if passed, the heating E101 element is switched off/on. There is a relay block that introduces a hysteresis in the controller, which avoids chattering (Colón and Pait, 2000)

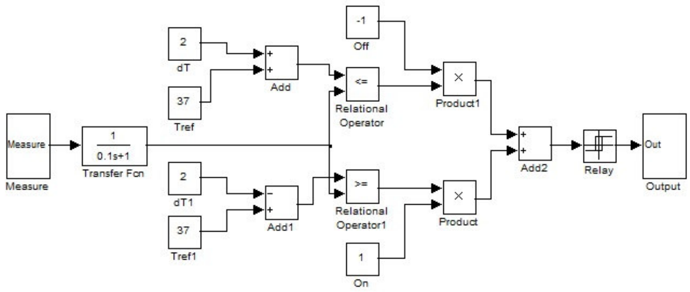

Figure 6. Block diagram of the ON/OFF controller.

4. CONTROLLER IMPLEMENTATION AND NONLINEAR EVALUATION

The controllers obtained above were implemented in MATLAB/Simulink, as presented in Fig. 7. The block "Measure" receives the sensor's signal, subtracts the equilibrium value in order to produce the variation signal Δh (or Δq or Δp), and transmits it to the closed loop controller (block "Controller"). The anti-windup mechanism (Seborg, 2004), which avoids the deep saturation due to the integral action, is implemented by the block "Add2". The signal generated by the controller, which is a voltage variation Δv, is added to the equilibrium value v*, and then sent to a saturation block, to finally reach the "output" block, that sends the signal to the acquisition board. The block "Controller" is presented in Fig. 8. This action is necessary when the controller is used for large signal values, in which the model's linear approximation is no longer valid, and the controller designed for this situation saturates frequently. In that situation, the pump saturates, as the control signal reaches values above 10.0 Volts (or below 0.0 V), and the PID memory, represented by the integral action, acts like a capacitor that must be discharged before the saturation ends. The acquisition board used in this work is the NI 6009 from National Instruments, whose output signal values are between 0 to 5.0 Volts. An analog electronic circuit, based in operational amplifiers, was designed in order to amplify the signal by a factor of two (not shown here).

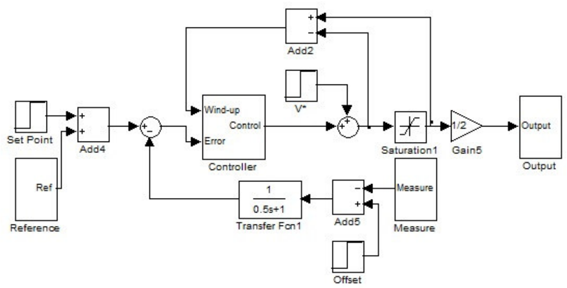

Figure 7. PID controller implementation in Simulink.

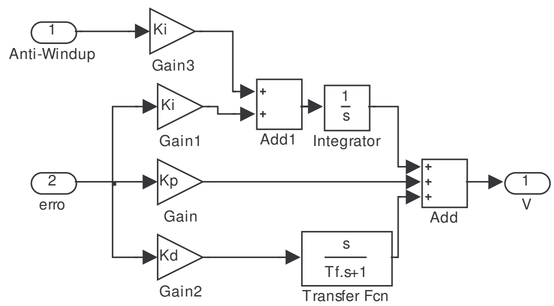

Figure 8. Controller block.

In the following, the controllers obtained are tested in the experimental apparatus.

4.1. Level PID Control Implementation

In Fig. 9, it is shown the reference signal (blue) for the level control designed in section 3.1, and the corresponding real level signal (measured by the ultra-sonic sensor LIC 102.1, in red). The corresponding voltage signal is presented in Fig. 10. The reference signal begins at the level h* = 12.0 cm, which is the equilibrium level for which the PID controller was designed. The initial voltage applied to the pump is about 4.0 V, which is near the 3.7 V equilibrium voltage calculated by using the mathematical model. The transient response is second order and underdumped, as predicted in Fig. 3. The voltage applied to the pump, in Fig. 10, is also in good agreement with Fig. 3, as can be seen in the transition from 13 cm to 12 cm in instant 160 seconds. The voltage peak is about 4.0 V which is near the 3.0 V predicted by the model. The differences between the simulation and the experimental result are due to the nonlinearities in the real system. It is easy to see the noise effect in the voltage signal, which is amplified by the derivative part. It could be worse if the filter were not added.

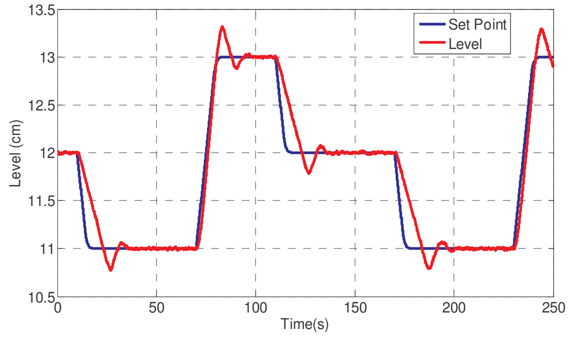

Figure 9. Level signal and reference signal.

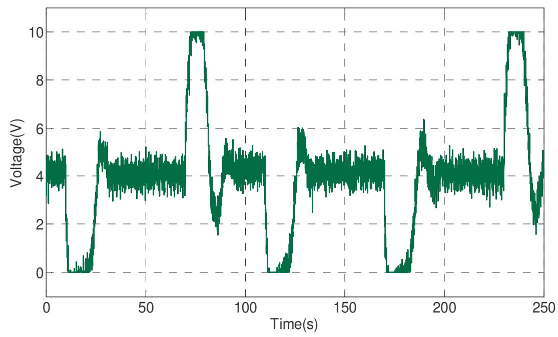

Figure 10. Voltage signal applied to the pump.

4.2. Flow PID control

The controller configuration shown in Fig. 7 and Fig. 8 was used in the flow control, but using the parameters calculated in section 3.2. In Fig. 11 it is presented the flow reference signal, the real flow signal (measured by the flow sensor FIC 101.1) and in Fig. 12 the voltage signal in the pump. It can be seen that the reference signal begins in the equilibrium value of q*=1.2 L/min, and the voltage signal begins at about the value calculated section 3.2, that is v*=3.7 V. In this case, on the other hand, the difference between the simulation in Fig. 4 and the response in Fig. 11 is larger than in the level control case (this difference is due to the nonlinearities of the real system). Also, the voltage presents differences. Despite that, the controller designed is efficient in following the reference signal.

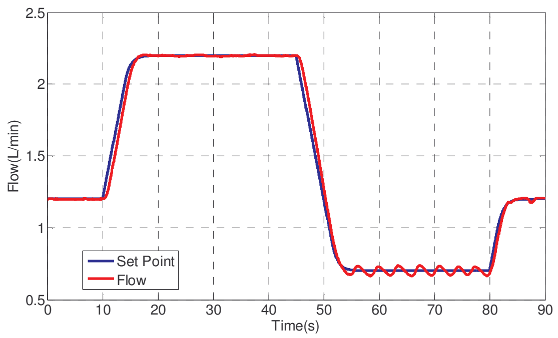

Figure 11. Flow signal and reference signal .

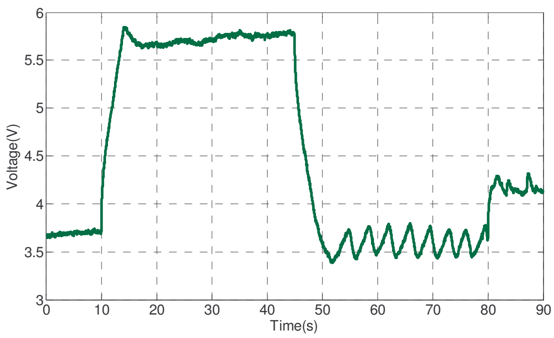

Figure 12. Voltage signal applied to the pump.

4.3. Pressure PID Control

The same PID implementation, shown in Fig. 7 and Fig. 8, was used in the pressure control, and the results are shown in Fig. 13 and Fig. 14. The reference signal begins at the equilibrium value of p* = 0.1 bar, as calculated in section 3.3. The pump's voltage signal begins near the calculated equilibrium value, that is v* =7.0 V, that is in good agreement with the model. The dynamic response shown in Fig. 13, on the other hand, shows an underdamped response with a dumping coefficient less than the one shown in Fig. 5. In order to the controller follows the reference signal, a very oscillatory voltage must be applied to the pump.

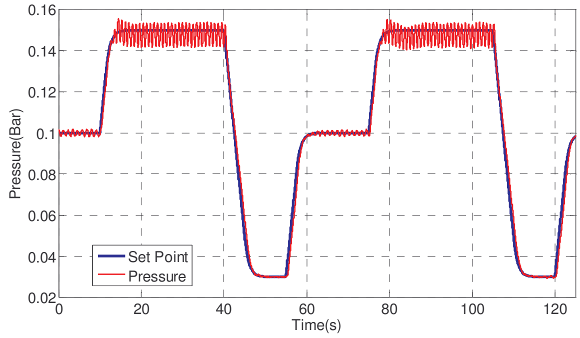

Figure 13. Pressure signal and reference signal.

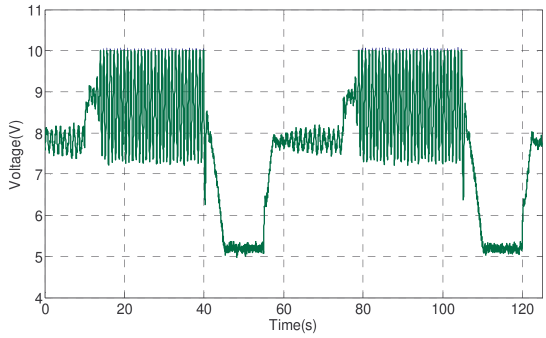

Figure 14. Voltage signal applied to the pump.

This difference between the linear model's prediction and the real system response probably is due to a nonlinear effect that could not be captured by the modeling process used in this work. On the other hand, the controller achieved its objective.

4.4. Temperature Control

In the temperature control, after a reconfiguration of the piping system, the controller presented in Fig. 6 was applied to the resistance E101, and the experimental results are presented in Fig.15. The set point temperature was fixed in 37°C with an interval of about 2°C (relay characteristic). The blue curve, that is the measured temperature by the TIC 104 sensor, clearly shows that the system dynamics has delay. In fact, despite the fact that the resistance is turned off when the maximum temperature is reached, the temperature continues to rise, due to the water convection. A better performance could be achieved if a PID controller were designed for this case, which would require a mathematical model, like in the cases above.

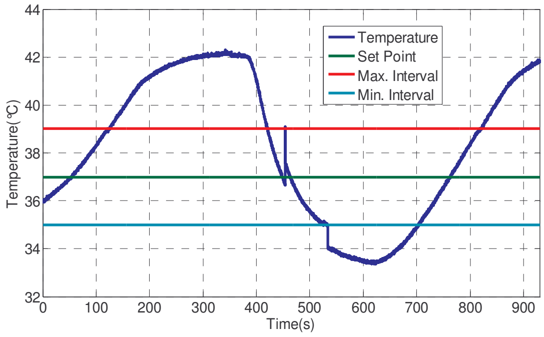

Figure 15. Temperature ON/OFF controller result.

5. CONCLUSIONS

In this work, several mathematical models for a didactical plant were determined, by using system identification ideas. The models have a static nonlinear part in series with a dynamic linear part. The system has 4 sensors (water level, water flow, water pressure and water temperature) and two actuators (pump and resistance) and the models related the pump's voltage with flow, level and pressure (the piping configuration must be changed in order to obtain those models). PID controllers were designed around equilibrium points with the aid of the linearized models (around equilibrium points). The temperature controller, on the other hand, was an ON/OFF controller, and no model for this case was determined. The controllers were implemented in a PC computer in MATLAB/Simulink, by using an acquisition board, and they were used to control the real variables. The controllers had good performance, and the voltages generated were always in the limits expected, despite the fact that the simulation predictions had some differences in relation to the experimental case (mainly the control signal). Practical PID implementation mechanisms were also used (filters, saturations and anti-windup). The results obtained will be used for didactical purposes in process control courses.

Future works include the identification of more precise nonlinear models, the design of more advanced control techniques, like multivariable robust and model predictive control, in which two or more measured variables are controlled, considering its coupling, and possibly nonlinear controllers, that are not restricted to work in a neighborhood of an equilibrium point. This work was partly developed as an undergraduate project.

6. REFERENCES

Colón, D and Pait, F. Análise de Estabilidade e Síntese de Sistemas Híbridos, Anais do XIII Congresso Brasileiro de Automática, Natal, RN, 2000.

DIN 19227: "Graphical symbols and identifying letters for process, control engineering".

Dorf, R. C.; Bishop, R. H. "Modern Control Systems". 12th edition. Prentice-Hall, 2010.

Garcia, C., "Modelagem e Simulação", Editora da USP, 1992

Helmich J., "Manual PCS Compact Workstation", ADIRO - Alemanha, Novembro de 2004.

Keyser, R., Naşcu, I.; Buzdugan, T. "Modeling And Simulation of a Level Control System", Dept. of Autom., Tech. Univ. of Cluj Napoca, Cluj-Napoca, Romania. Published in May 28-30, 2010.

Liduário, P. M., Colón, D., Bueno, A. M., Godoy E. P., Balthazar, J. M., "Nonlinear Model and Control of a Fluid Process Plant", Proceedings of the VIII Congresso Nacional de Engenharia Mecânica (CONEM 2014), Uberlândia, MG, 2014.

Ljung, L., 1999. "System Identification: Theory for the User", 2nd ed. PTR Prentice Hall Information and System Science Series, PTR Prentice-Hall, Upper Saddle River, N.J. 2nd edition.

Ogata, K. 2010. "Modern Control Engineering". Pearson, Upper Saddle River, 5th edition.

Seborg, D. E. et al, "Process Dynamics and Control", second edition, Wiley, 2004.

7. RESPONSIBILITY NOTICE

The author(s) is (are) the only responsible for the printed material included in this paper.