VOL. 15, NO. 19, OCTOBER 2020 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

CLASSICAL AND NON-LINEAR PREDICTIVE CONTROL APPLIED TO A NON-LINEAR LIQUID LEVEL SYSTEM

Mairon Steven Sanchez Perez and Diego F. Sendoya-Losada
Department of Electronic Engineering, Faculty of Engineering, Surcolombiana University, Neiva, Huila, Colombia
E-Mail: diego.sendoya@usco.edu.co

ABSTRACT

In this work two model-based controllers have been designed in order to regulate a non-linear liquid level system. First, a Proportional-Integral (PI) controller was designed. This algorithm requires a linear model, so the model was linearized around a certain equilibrium point. This gives bad results when the setpoint lies far from the equilibrium output level. Secondly, a Non-linear Extended Prediction Self-Adaptive Control (NEPSAC) was designed. A big advantage is that no linearization is required. Consequently, a correct model is available at each point. This explains why NEPSAC gives the best results: a low settling time, no overshoot, and equally good results for all setpoints. Finally, the performance of the controllers is evaluated, in order to carry out a tracking to a reference level and an effective rejection of the disturbances.

Keywords: level control, MBPC, PI, NEPSAC, RMSE.

1. INTRODUCTION

In almost all the applications of industrial processes, the control of the variables is critical for the safe and efficient operation of the same. The most common controlled variables are pressure, level, temperature, and flow. Level control loops are very common in the industry, in fact, they occupy the second place after the flow control loops. Due to the importance and the large number of processes that require a precise level control, the Surcolombiana University recently acquired a tank system called CE105 MV. As can be seen in Figure-1, this system presents a configuration similar to that which can be found in many industrial applications or as part of a much larger and more sophisticated plant.

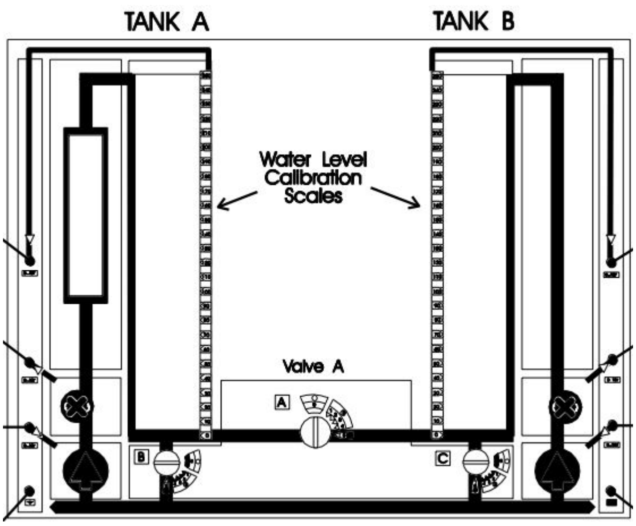

Figure-1. CE105MV System.

Usually the control of the liquid level in a tanks system is done by classical control techniques such as PI or PID, due to its well-known and simple structure [1]-[3]. For the design of these conventional controllers it is necessary to choose a setpoint and then to find a linear model of the system, ensuring that the control works well in this region, but when it moves away from the setpoint, the controller loses effectiveness [4], [5]. In this contribution, Model-Based Predictive Control (MBPC) is applied to a coupled tank system - CE105 MV [6]. For this paper, a Single-Input Single-Output (SISO) system configuration has been considered, as can be seen in Figure-2.

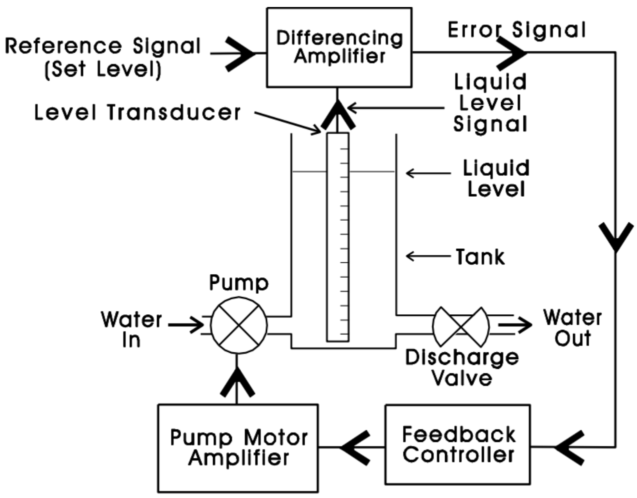

Figure-2. Liquid level plant.

MBPC principle is simple. A process model is determined in advance, for example by modeling or identification. With the aid of this model the process output corresponding with different inputs is calculated. The input which gives the best results is then applied to the system. It allows controlling processes with unusual dynamic behaviors, such as non-minimal phase processes, highly oscillatory processes, or unstable processes [7].

There are many papers about liquid level control using classical or advanced control techniques [8]-[14]. In this contribution, the Proportional-Integral (PI) controller is used, in order to carry out a tracking to a reference level and an effective rejection of the disturbances. Then a comparison with the Non-linear Extended Prediction Self-

Adaptive Control (NEPSAC) algorithm is done. This is the non-linear variant of MBPC. The big difference is that NEPSAC nowhere requires the linearization of the non-linear process model. The performance of these controllers is tested and evaluated in a simulation environment. The simulation is done using the Matlab®/Simulink® software.

2. MATERIALS AND METHODS

2.1 Process Model

The mathematical model presented is given by the equations that describe the complete system. Figure-2 shows its respective scheme. The dynamic model is determined by the relationship between the input flow $q_i(t)$ and the output flow $q_o(t)$ through the discharge valve. Equation (1) describes this relationship.

$$q_i(t) - q_o(t) = \dot{V}(t) = A\dot{h}(t) \quad (1)$$

$V(t)$ is the volume of the tank, A is the cross-sectional area of the tank and $h(t)$ is the liquid level to be controlled. The output flow through the discharge valve is related to the level of liquid in the tank by (2):

$$q_o(t) = a_v C_v \sqrt{2gh(t)} \quad (2)$$

a_v is the cross-sectional area of the valve orifice, C_v is the discharge coefficient of the valve and g is the gravity constant. Combining (1) and (2) gives:

$$\dot{h}(t) + a_v C_v \sqrt{2gh(t)}/A = q_i(t)/A \quad (3)$$

The input flow is related to the voltage applied to the pump in a linear manner, by means of (4):

$$q_i(t) = K_b v_i(t) \quad (4)$$

K_b is the gain of the pump and $v_i(t)$ is the input voltage of the system. Finally, (5) shows a non-linear relationship between the input voltage $v_i(t)$ and the liquid level inside the tank $h(t)$.

$$\dot{h}(t) + a_v C_v \sqrt{2gh(t)}/A = (K_b/A) v_i(t) \quad (5)$$

Although the water enters the tank from the bottom, the dimensions of the system are small enough so that no delay in the change of level is observed. The values of the system parameters are shown in Table-1.

Table-1. System parameters [15].

Symbol	Description	Value
A	Cross-sectional area of the tanks	$9350\text{x}10^{-6}$ m^2
a_v	Cross-sectional area of the valve orifice	$78.50\text{x}10^{-6}$ m^2
C_v	Discharge coefficient of valve	0.2
h_{max}	Maximum liquid level	0.25 m
v_{max}	Maximum input voltage	10 V
K_b	Pump gain	$6.66\text{x}10^{-6}$m^3 /sV
g	Gravity constant	9.8 m/s^2

2.2 PI Controller

The non-linear dynamics (5), must be linearized before PI algorithm can be applied. The linearization was made using the first order Taylor expansion around the equilibrium point $h^* = 0.125$ m. For the equilibrium point: $f(v_i^*, h^*) = 0$. Linearizing the tank dynamics yields the following transfer function:

$$\frac{\bar{h}(s)}{\bar{v}_i(s)} = \frac{0.03391}{95.12s+1} \quad (6)$$

Discretization of the continuous-time transfer function with a sampling period of $T_s = 1$ s results in the following pulse transfer function:

$$\frac{\bar{h}(z)}{\bar{v}_i(z)} = \frac{0.0003547}{z-0.9895} \quad (7)$$

The algorithm that describes the behavior of the PI controller is presented in (8):

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau \quad (8)$$

$u(t)$ is the output signal of the PI controller, which in this case corresponds to the voltage applied to the pump. $e(t)$ is the input signal of the PI controller, which is defined as $e(t) = r(t) - y(t)$, where $r(t)$ is the setpoint and $y(t)$ is the output of the process, that is, the level of liquid in the tank. K_p is the proportional gain and K_i is the integral gain.

Applying the Laplace transform to (8), the transfer function of the PI controller is found:

$$U(s)/E(s) = 80.1449 + 0.8426/s \quad (9)$$

2.3 NEPSAC

The NEPSAC algorithm is based on the following model for its implementation:

$$y(t) = x(t) + n(t) \quad (10)$$

Where $y(t)$ is the measurement of the process output, $x(t)$ is the output of the process model, and $n(t)$ is the disturbance.

This algorithm requires a prediction of future results to perform the control action, for this the term of prediction horizon N_2 is introduced, which can be defined as:

$$y(t+k|t) = x(t+k|t) + n(t+k|t) \quad (11)$$

In this way the future output can be described as the contribution of two important parts, a base component and an optimal component.

$$y(t+k|t) = y_{base}(t+k|t) + y_{opt}(t+k|t) \quad (12)$$

However, for nonlinear systems, the term $y_{opt}(t+k|t)$ can gradually be made equal to zero in an iterative way by selecting the base control strategy $u_{base}(t+k|t)$ appropriately. The superposition principle is then no longer involved, and the algorithm leads to the optimal solution, even for non-linear systems.

At each sampling instant an initial $u_{base}(t+k|t)$ is selected. It should be the objective to obtain finally (in an iterative way) a control policy $u_{base}(t+k|t)$ which is as close as possible to the optimal strategy $u(t+k|t)$. In order to minimize the number of iterations, it is thus wise to make a good initial guess for $u_{base}(t+k|t)$. A simple but effective choice is to start with $u_{base}(t+k|t) = u(t+k|t-1)$, i.e. the optimal control policy derived at the previous sampling instant. Once a $u_{base}(t+k|t)$ has been chosen, $\delta u(t+k|t)$ is calculated. For a non-linear model, this is not the optimal control because the principle of superposition does not hold. However, it can be expected that the resulting u signal is closer to the optimal control than the previous guess u_{base}. So, for a non-linear model, it is suggested to continue the procedure -at the same sampling instant- by taking $u(t+k|t)$ as a new set $u_{base}(t+k|t)$. Continuing this iterative procedure, it can be expected that u_{base} will converge to the optimal u. Indeed, each time that u_{base} is closer to u, it means that the δu is smaller; and thus, also the term y_{opt} becomes smaller. The superposition principle has less impact. Finally, when the δu is practically zero, the superposition principle is no longer involved and the calculated control signal will thus be optimal, also for the non-linear system. The whole procedure is thus certainly not based on any local linearization of the non-linear model.

NEPSAC algorithm does not require that (5) is linearized, but it does have to be discretized. A first-order backward Euler discretization scheme is used in this work, which leads to the following discretized tank dynamics:

$$h(t) = \left(1 - \frac{T_s a_v C_v \sqrt{2g}}{2A\sqrt{h^*}}\right) h(t-1) + \frac{T_s K_b}{2A} v_i(t-1) \quad (13)$$

Note that the discretization introduces a time delay of a single sampling period to the input voltage $v_i(t)$. When the process parameters listed in Table-1 are used, the following equation is obtained:

$$h(t) = 0.9895h(t-1) + 0.0003547v_i(t-1) \quad (14)$$

The step response, which is constant for linear (or linearized) models differs in each point for non-linear models. Hence, the step response must be recalculated for each iteration. Therefore, the model is linearized around the point of interest, and consequently the step response is calculated. Then the step response of the discretized model can be calculated and used in the calculation of U^*.

An appropriate (i.e. close to the optimal value) initial guess for $u_{base}(t|t)$ lowers the number of iterations to get $\delta u(t|t) = 0$. Therefore, the initial guess is set equal to the actual control action at the previous time sample, $u(t-1)$. Since the control horizon $N_u = 1$, $u_{base}(t+k|t) = u_{base}(t|t)$ for $0 < k < N_2$. Iterations are stopped once $\delta u(t|t)$ is lower than a threshold (10^{-6}). It can be said that $u(t) = u_{base}(t|t)$.

At the end, clipping is applied to the control action $u(t)$, so that it is in the range $0 \leq u(t) < 10$ V.

3. RESULTS AND DISCUSSIONS

The simulation was conducted towards two scenarios to compare the behavior of the PI and NEPSAC algorithms. The performance of the controllers is evaluated, in order to carry out a tracking to a reference level and an effective rejection of the disturbances.

3.1 SCENARIO 1: Disturbance Rejection

In order to evaluate the performance of the controlled system when a disturbance is applied, the setpoint is fixed at 0.125 m. After the response reaches the steady state a disturbance is applied to the system, which consists of a change in the discharge coefficient of valve, that is, C_v goes from 0.2 to 0.4. Finally, when the response reaches the setpoint again, the discharge coefficient is changed from 0.4 to 0.2. Figure-3 shows the output level when PI and NEPSAC (prediction horizon $N_2 = 20$) algorithms are used. This prediction horizon is used because it allows the response to have the shortest settling time with a small overshoot.

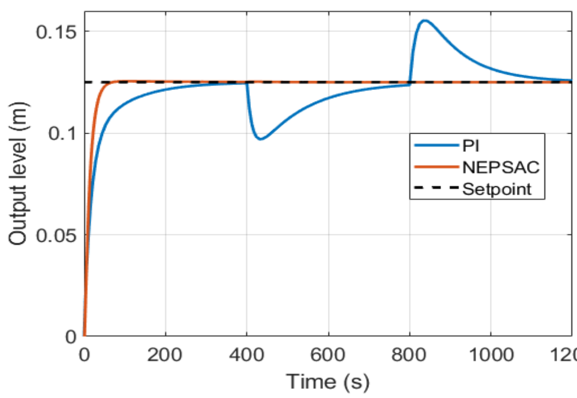

Figure-3. Disturbance rejection with PI and NEPSAC.

ARPN Journal of Engineering and Applied Sciences

Initially the response does not present overshoot in both cases. When the system stabilizes the discharge coefficient of valve is changed from 0.2 to 0.4, this disturbs the system because the outflow increases. At 400 seconds the output level of the system is reduced due to this situation. However, the controllers increase the control effort (as can be observed in Figure-4) to compensate for the reduction in the output level and increase it again to 0.125 m. When NEPSAC algorithm is used, there is no significant change in the output level of the system.

Finally, at 800 seconds, the discharge coefficient of valve is changed from 0.4 to 0.2. Because of this generates a reduction in the output flow, it is observed how the output level increases when PI acts. Again, this change is negligible when NEPSAC in used. Both controllers adjust the voltage applied to the system to compensate for the increase in level. The output stabilizes again at 0.125 m, showing the robustness of the controllers against changes in the process parameters.

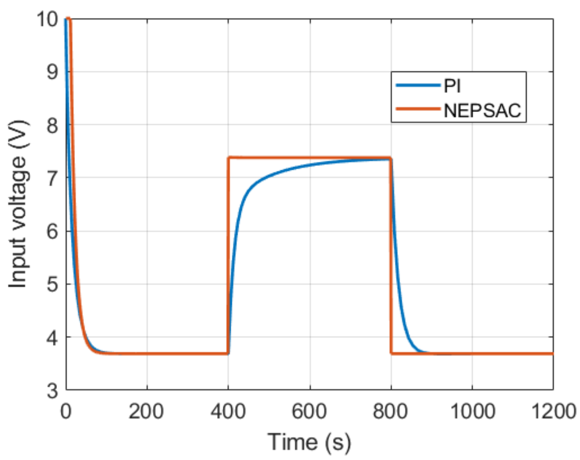

Figure-4. Optimal input for disturbance rejection with PI and NEPSAC algorithms.

3.2 SCENARIO 2: Setpoint Tracking

In order to evaluate how the system behaves in closed loop at different points of operation, a reference signal composed of 4 steps with amplitudes of 0.05, 0.10, 0.15 and 0.20 m is applied. Figure-5 shows how the process output $y(t)$ tracks a setpoint for a prediction horizon $N_2 = 20$.

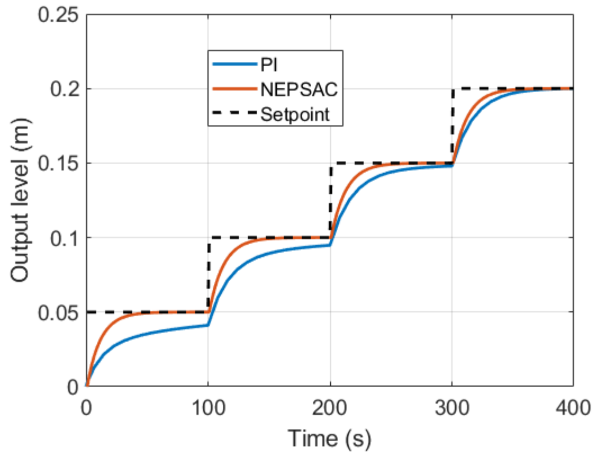

Figure-5. Setpoint tracking with PI and NEPSAC.

For PI controller, the model is linearized in a point corresponding with an output level $h^* = 0.125$ m. As can be observed, the response is better for the last steps. But as soon as the setpoint lies a bit further, results worse fast, for instance, the first setpoint step shows a longer settling time. As soon as the setpoint comes in a value close to 12.5 cm, the controller starts giving good results.

It can be seen that NEPSAC gives very good results. The output follows the setpoint changes without overshoot and within a small time. Notice that results are equally good for all setpoints. This is because the model is not linearized. The correct model is being used in all points. Consequently, the model is very good. This leads to good predictions and a very good control of the process.

Finally, Figure-6 shows the control effort for both controllers.

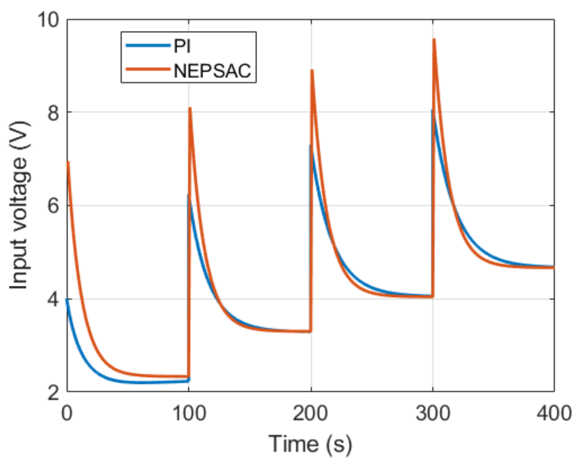

Figure-6. Optimal input with PI and NEPSAC algorithms.

Because of NEPSAC uses a more precise model, depending on the setpoint in which it works, then the control effort required is greater.

In order to obtain a detailed results analysis of the comparative study between the PI and NEPSAC

ARPN Journal of Engineering and Applied Sciences

algorithms, the Root Mean Square Error (RMSE) variations were used.

$$RMSE = \sqrt{\frac{\sum_{t=1}^{N}[r(t)-y(t)]^2}{N}} \tag{15}$$

where $r(t)$ is the setpoint signal, $y(t)$ is the output signal, and N is the number of samples. Table-2 shows the RMSE variations for both the PI algorithm and the NEPSAC algorithm when the two simulation scenarios are applied.

Table-2. RMSE Variations for PI and NEPSAC.

Scenario	PI (%)	NEPSAC (%)
1	6.56	4.41
2	29.39	12.24

The implemented NEPSAC algorithm presents a better behavior in the two scenarios compared to the PI algorithm.

Although the computational cost of the implementation of the NEPSAC algorithm is higher, the system response is noticeably faster than when the PI algorithm is used. This can be an advantage in systems where it is necessary to minimize the values of overshoot and settling time. However, for processes where these requirements are not necessary, the PI algorithm would be more viable because of its simplicity of implementation.

4. CONCLUSIONS

In this work, the performance of PI and NEPSAC algorithms was evaluated. It can be observed that the NEPSAC is more effective in rejecting disturbances. This is because it uses a greater control effort when there are changes in the system parameters. In the same way, because the NEPSAC uses the non-linear model of the plant to make the predictions, then the follow-up to the variations in the reference level is much better. In addition, as the model used is more precise, the control effort required is greater when changing from one setpoint to another.

REFERENCES

[1] J. F. Caipa-Roldán, J. M. Salamanca, and J. L. Rodríguez-Herrera. 2010. Control digital de nivel para sistema de tanques interconectados mediante servo-válvula. Ingeniería Investigación y Desarrollo. 10(1): 55-63.

[2] E. Govinda-Kumar and J. Arunshankar. 2018. Control of nonlinear two-tank hybrid system using sliding mode controller with fractional-order PI-D sliding surface. Computers & Electrical Engineering. 71: 953-965.

[3] V. Bhambhani and Y. Q. Chen. 2008. Experimental study of fractional order proportional integral (FOPI) controller for water level control. 47th IEEE Conference on Decision and Control.

[4] A. Başçi and A. Derdiyok. 2016. Implementation of an adaptive fuzzy compensator for coupled tank liquid level control system. Measurement. 91: 12-18.

[5] S. R. Mahapatro. 2014. Control algorithms for a two tank liquid level system: An experimental study. Diss.

[6] D. F. Sendoya-Losada, D. C. Vargas-Duque and I. J. Ávila-Plazas. 2018. Implementation of a neural control system based on PI control for a non-linear process. 1st IEEE Colombian Conference on Applications in Computational Intelligence.

[7] R. De Keyser. 2003. A ‘Gent’le approach to predictive control. UNESCO Encyclopedia of Life Support Systems (EoLSS), Eolss Publishers Co Ltd, Oxford.

[8] S. Banerjee, S. Das, S. Khatua, A. Gupta, P. Arvind, S. Kumari and S. Das. 2017. Liquid level control system using observer based constrained discrete-time model predictive control. IEEE International Conference on Circuit, Power and Computing Technologies (ICCPCT). pp. 1-8.

[9] A. Shehu and N. A. Wahab. 2016. Applications of MPC and PI controls for liquid level control in coupled-tank systems. IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS). pp. 119-124.

[10] M. Dulău and T. M. Dulău. 2016. Multivariable system with level control. Procedia Technology. 22: 614-622.

[11] P. Gondaliya and M. C. Patel. 2015. Design and implementation of model predictive control for a liquid level system using LabVIEW. International Journal for Scientific Research & Development. 3(6): 1313-1316.

[12] M. Essahafi. 2014. Model predictive control (MPC) applied to coupled tank liquid level system. arXiv preprint arXiv: 1404.1498.

[13] M. U. Khalid and M. D. Kadri. 2012. Liquid level control of nonlinear coupled tanks system using linear model predictive control. IEEE International

VOL. 15, NO. 19, OCTOBER 2020 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

Conference on Emerging Technologies (ICET). pp. 1-5.

[14] Currie and D. I. Wilson. 2010. Lightweight model predictive control intended for embedded applications. IFAC Proceedings. 43(5): 278-283.

[15] TecQuipment. 2016. CE105 and CE105MV Coupled Tanks user manual. TecQuipment Ltd.

[16] Sendoya-Losada, D. F., Faiber Robayo Betancourt, and José (8): Salgado Patrón. 2017. Application of a predictive controller. 2661-2667.

[17] Sendoya-Losada, Diego F. and Johan Julián Molina Mosquera. 2017. Linear with variable time delay ingeneral anesthesia. ARPN Journal of Engineering and Applied Sciences. 12

[18] and nonlinear predictive control algorithms applied to a heated tank system. ARPN Journal of Engineering and Applied Sciences. 12(23): 6895-6903.

[19] Sendoya-Losada, D. F. 2019. Principles, applications and perspectives of predictive control. ARPN Journal of Engineering and Applied Sciences. 14(13): 2464-2467.

[20] D. F. Sendoya-Losada, J. D. Quintero-Polanco. 2018. Control of yaw angle in a miniature helicopter. ARPN Journal of Engineering and Applied Sciences. 13(2): 620-626.

[21] D. F. Sendoya-Losada, J. D. Quintero-Polanco. 2018. PID controller applied to an unmanned aerial vehicle. ARPN Journal of Engineering and Applied Sciences. 13(1): 325-334.

[22] D. F. Sendoya-Losada, J. D. Quintero-Polanco. 2018. Design and validation of a PID auto-tuning algorithm. ARPN Journal of Engineering and Applied Sciences. 13(11): 3797-3802.

[23] D. F. Sendoya-Losada. 2018. KCR PID auto-tuner algorithm applied to a FOT device. ARPN Journal of Engineering and Applied Sciences. 13(7): 2505-2510.

[24] D. F. Sendoya-Lada, J. O. Arroyave-Quezada and y A. J. Velazquez-Pobre. 2019. EPSAC and NEPSAC Algorithms Applied to a Nonlinear Liquid Level System. en Colombian Conference on Automatic Control (CCAC). pp. 1-5.