Implementation of remote control for time delay systems based on adaptive and internal model approaches

YAHYA Olfa, Bedoui Saïda and ABDERRAHIM Kamel
Numerical Control of Industrial Processes,
National School of Engineers of Gabes, University of Gabes,
St Omar Ibn-Khattab, 6029 Gabes, Tunisia.
Email: olfayahya@gmail.com, bedoui.sayda@gmail.com, kamelabderrahim@yahoo.fr
Telephone: +216 94 73 88 90; Fax: +216 75 39 21 90

Abstract—**This work focus on practical implementation of remote control. Adaptive and internal model control (IMC) strategies have been adopted. Thus, the adaptive auto-adjustable controller is based on pole placement control method and the hierarchical identification strategy. Bluetooth has been used as wireless communication protocol. Practical results for two processes, first order system with variable time delay and fluid level control process, show satisfactory results of the application.**

Keywords—Remote Control, Time Delay Systems, Bluetooth, Adaptive Control, Internal Model Control.

I. Introduction

In engineering fields, even if a process doesn't include intrinsic delays, these may appear in the control loop by the response times of its devices as actuators and sensors, the necessary time for the computation of control laws or for data transmission in networks. So, most of practical systems presents delays which are often simply neglected. However, their presence can induce poor performance and instabilities of the controlled system, therefore it is important to take them into account.

To be competitive, an industrial process has to be automated. Competitiveness means keeping it as close as possible to a specified reference trajectory. The synthesis and the implementation of a numerical control requires the knowledge of a model describing the dynamic of the controlled system. This model can be obtained by physical laws presenting the natural evolution of the process. But, this method usually allows complicated models difficult to be exploited in implementation. In the literature, different approaches have been proposed for the identification of this type of models [1], [2], [3]. In the following, the estimation of monovariable discrete time delay systems will be achieved based on gradient iterative approach which has been developed in [1].

In this paper, we present remote application results of two control strategies: adaptive and internal model methods. Adaptive control is the set of techniques allowing, in real time, automatic adjustment of the controller parameters, to achieve and keep some desired control performance when parameters of the process are variable in time and/or disturbances change their dynamics [4], [5], [6]. It has been successfully applied to control many industrial processes. We aim to apply adaptive pole placement control with RST structure for a time varying linear system based upon identification of changes of system characteristics in the time response. The principle is relocating the overall closed loop poles to obtain desired response taking into account process variations at each time step.
Internal model control (IMC) is an advanced control technique that has been introduced for single-input-single-output (SISO) systems by Garcia and Morari in 1982 [7]. It has been widely used in control applications [8]. The choice of this typical structure is essentially due to its robustness to model errors and inaccuracies. Also, it is distinguished by a simple and intuitive design with reduced number of online adjusted parameters. By integrating a filter into the controller, IMC satisfies perfect tracking performance despite effects of any disturbances.

We choose for the implementation of such controllers (adaptive and IMC), the remote strategy via the Bluetooth communication protocol as it is innovative, easy to adapt in practice even in the industrial field and available in a large number of devices. We'll use the bluetooth module HC-05 connected to an Arduino board that is used as an acquisition card, and the control signal will be sent by a computer equipped with bluetooth adapter.

This paper is structured as follows. Section II presents a gradient based iterative algorithm for the identification of discrete time delay systems using the hierarchical principle. In section III we propose adaptive pole placement control of RST structure using hierarchical identification approach. Section IV adresses IMC method. Experimental results of remote adaptive and internal model control applications are shown in section

978-1-5386-1084-8/17/$31.00 ©2017 IEEE

Authorized licensed use limited to: Sami Shamoon College of Engineering. Downloaded on November 10,2023 at 12:44:14 UTC from IEEE Xplore. Restrictions apply.

V. Finally, section VI concludes the paper.

II. Identification of time delay systems using hierarchical gradient method

In this section, we address the problem of identifying a system whose bahavior can be described by the following equation,

$$A(z^{-1})y(k) = z^{-d}B(z^{-1})u(k) + v(k) \tag{1}$$

where $u(k)$ and $y(k)$ are respectively the system input and output, $v(k)$ is a sequence of random and independent variables with zero mean and finite variance, d is the pure delay of the system, $A(z^{-1})$ and $B(z^{-1})$ are two polynomials defined by

$$\begin{cases} A(z^{-1}) = 1 + a_1 z^{-1} + ... + a_{n_A} z^{-n_A} = 1 + \sum_{i=1}^{n_A} a_i z^{-i} \\ B(z^{-1}) = b_1 z^{-1} + ... + b_{n_B} z^{-n_B} = \sum_{i=1}^{n_B} b_i z^{-i} \end{cases}$$

Using the expressions of $A(z^{-1})$ and $B(z^{-1})$, equation (1) can be rewritten as follows:

$$y(k) = -\sum_{i=1}^{n_A} a_i y(k-i) + \sum_{i=1}^{n_B} b_i u(k-i-d) + v(k) \tag{2}$$

We assume that $A(z^{-1})$ and $B(z^{-1})$ are coprime polynomials and that orders n_A and n_B are known.

- **Development of the algorithm:**

Referring to the works developed in [1], we consider the following criterion:

$$J(\hat{\theta}, \hat{d}) = e^2(k) \tag{3}$$

where $\hat{\theta}$ is the estimated parameter vector given by,

$$\hat{\theta}^T := \left[\hat{a}_1(k), \hat{a}_2(k), ..., \hat{a}_{n_A}(k), \hat{b}_1(k), \hat{b}_2(k), ..., \hat{b}_{n_B}(k)\right] \tag{4}$$

$\hat{d}$ is the estimated value of the process delay, $e(k)$ is the prediction error defined by

$$e(k) = y(k) - \hat{y}(k) \tag{5}$$

and $\hat{y}(k)$ represents the output prediction such that.

$$\hat{y}(k) = -\sum_{i=1}^{n_A} \hat{a}_i(k-1) y(k-i) + \sum_{i=1}^{n_B} \hat{b}_i(k-1) u(k-i-\hat{d}(k-1)) \tag{6}$$

The principle of the hierarchical method consists in decomposing the nonlinear cost function J into two simple costs: $J(\hat{\theta}, d)$ for fixed delay $d = \hat{d}(k-1)$ and $J(\theta, \hat{d}(k))$ for fixed $\theta = \hat{\theta}(k)$. This is equivalent to minimize the two costs given below,

$$J_1(\theta) = J_1(\theta, \hat{d}(k-1)) = e^2(k)|_{\hat{d}(k-1)} \tag{7}$$

$$J_2(d) = J_2(\hat{\theta}(k), d) = e^2(k)|_{\hat{\theta}(k)} \tag{8}$$

Determination of the estimated parameters vector $\hat{\theta}(k)$ and of the time delay $\hat{d}(k)$ is based on the gradient method.
Minimizing the cost function (7) provides computing the identified parameter vector iteratively:

$$\hat{\theta}(k) = \hat{\theta}(k-1) - \frac{\mu_1(k)}{2} \frac{\partial J_1(k)}{\partial \hat{\theta}(k-1)} \tag{9}$$

Then,

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \mu_1(k)\phi(k)e(k)|_d \tag{10}$$

where the observation vector is:

$$\phi(k) = \begin{bmatrix} -y(k-1) \\ \vdots \\ -y(k-n_A) \\ z^{-\hat{d}(k-1)} u(k-1) \\ \vdots \\ z^{-\hat{d}(k-1)} u(k-n_B) \end{bmatrix} \tag{11}$$

and $\mu_1(k)$ is the convergence factor.
Minimizing the cost $J_2(d)$ through the gradient algorithm allows estimating the time delay value iteratively according to the following formula:

$$\hat{d}(k) = \hat{d}(k-1) - \frac{\mu_2(k)}{2} \frac{\partial J_2(k)}{\partial \hat{d}(k-1)} \tag{12}$$

where $\mu_2(k)$ is the convergence factor.
Nextly, we obtain

$$\hat{d}(k) = \hat{d}(k-1) - \mu_2(k) \frac{\partial}{\partial \hat{d}(k-1)} \left[z^{-\hat{d}(k-1)} \sum_{i=1}^{n_B} b_i(k) u(k-i) \right] e(k)|_\theta \tag{13}$$

Using the approximation $Ln(z) = 1 - z^{-1}$, we have:

$$\hat{d}(k) = \hat{d}(k-1) - \mu_2(k) z^{-\hat{d}(k-1)} \sum_{i=1}^{n_B} b_i(k) \Delta u(k-i) e(k)|_\theta \tag{14}$$

where $\Delta u(k) = u(k) - u(k-1)$.
The convergence of $\hat{\theta}(k)$ and $\hat{d}(k)$ to their true values is guaranteed if $\mu_1(k)$ and $\mu_2(k)$ satisfies these inequalities:

$$0 < \mu_1(k) < \frac{2}{\phi(k)\phi^T(k)} \tag{15}$$

$$0 < \mu_2(k) < \frac{2}{\hat{\Gamma}\phi^T(k)\hat{\theta}(k)} \tag{16}$$

where $\hat{\Gamma}(k) = (1 - z^{-1}) \sum_{i=1}^{n_B} \hat{b}_i(k-1) \Delta u(k-i)$

Proof:
The proof is detailed in [1].
The implementation of the described hierarchical identification approach is summarized in the algorithm 1.

III. Adaptive pole placement controller design with RST structure

In this section, we propose the synthesis of self-adjusting controller through the indirect adaptive control scheme using the RST structure with the pole placement strategy. The principle of the method is illustrated in figure 1.

Authorized licensed use limited to: Sami Shamoon College of Engineering. Downloaded on November 10,2023 at 12:44:14 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1

step 1: Initialization:
put $\hat{\theta}(k) = \theta_0$, $d = 0$ and $k = 0$.

step 2: Increment k, construct the regressor vector $\phi(k)$ and select the values of $\mu_1(k)$ and $\mu_2(k)$ using (15) and (16) respectively.

step 3: Compute:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \mu_1(k)\phi(k)e(k)|_d$$

$$\hat{d}(k) = \hat{d}(k-1)$$

$$-\mu_2(k)z^{-\hat{d}(k-1)} \sum_{i=1}^{n_B} b_i(k)\Delta u(k-i)e(k)|_\theta$$

step 4: Stop iterating the algorithm if there is no more identification measurements. Otherwise, return to step 2 with $k = k+1$.

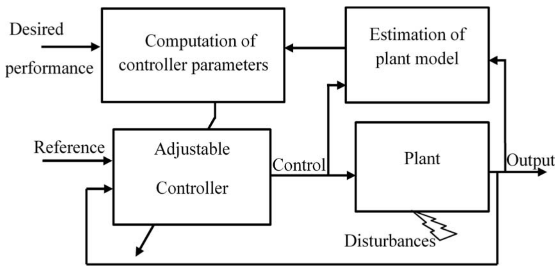

Fig. 1. Indirect adaptive control.

This type of scheme is developed from a natural approach of the adaptive concept. It consists in identifying in real time the model parameters of the process and to use them for the computation of the control law. The online estimation of the system time delay and parameters is performed using the hierarchical gradient method described in the previous section.

The synthesis of a controller using the classical pole placement strategy with RST structure is summarized in the loop of figure 2.

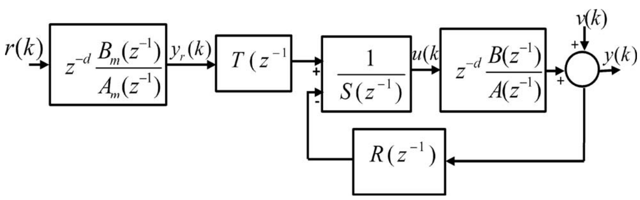

Fig. 2. RST control structure.

It leads to generate the control law $u(k)$,

$$u(k) = \frac{T(z^{-1})y_r(k+d+1) - R(z^{-1})y(k)}{S(z^{-1})} \quad (17)$$

where $y_r(k)$ is the reference signal obtained from the following model,

$$H_m(z^{-1}) = z^{-d}\frac{B_m(z^{-1})}{A_m(z^{-1})} \quad (18)$$

$R(z^{-1})$ and $S(z^{-1})$ are determined by solving the called *Diophantine* equation:

$$A(z^{-1})S(z^{-1}) + z^{-d}B(z^{-1})R(z^{-1}) = P(z^{-1}) \quad (19)$$

Since in tracking disturbances are assumed to be zero, we choose $T(z^{-}1)$ allowing a unitary static gain between $y_r(k+d+1)$ and $y(k)$, and compensating the regulation dynamic $P(z^{-}1)$, then we can choose that

$$T(z^{-1}) = GP(z^{-1}) \quad (20)$$

where,

$$G = \begin{cases} \dfrac{1}{B(1)} \; if \; B(1) \neq 0 \\ 1 \; if \; B(1) = 0 \end{cases} \quad (21)$$

To summarize the principle of implementing an auto-adjusting pole placement controller with an RST structure, it is performed according to the diagram steps given in figure 3.

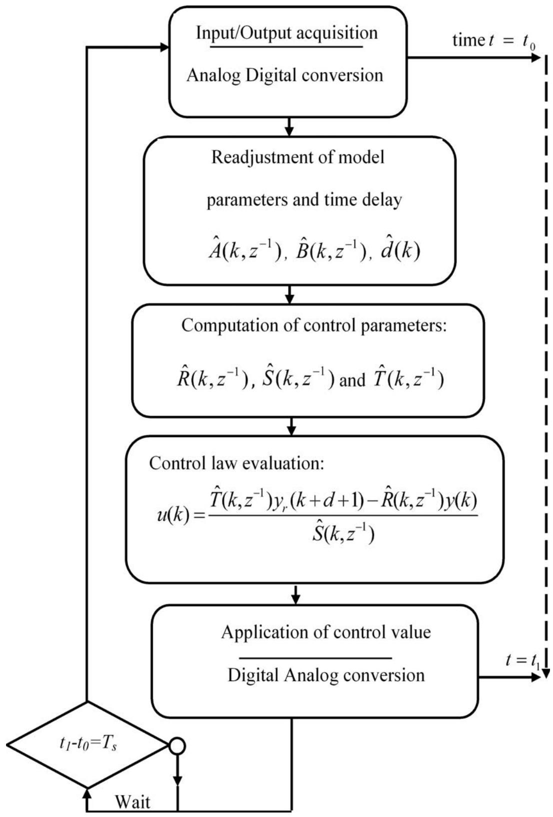

Fig. 3. diagram of adaptive RST control structure.

such that T_s is the considered sampling time.

IV. Internal Model Control

As indicating its name, the Internal Model Control (IMC) incorporates a simulation of the process by an internal model in its control loop. Consider the familiar IMC structure shown in figure 4.

In fact, the control value $u(k)$ generated by $Q(z)$ is simultaneously applied to the plant $G(z)$ and to its model $G_m(z)$). Comparison of their outputs generates a gap that will be used by the controller in order to yield better performance. The main advantages of IMC lie in its easy synthesis and the

Authorized licensed use limited to: Sami Shamoon College of Engineering. Downloaded on November 10,2023 at 12:44:14 UTC from IEEE Xplore. Restrictions apply.

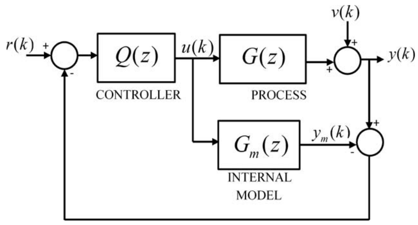

Fig. 4. IMC structure.

explicit manner to adjust the robustness. From figure 4, we have the expression of the closed loop output

$$y(k) = \frac{G(z)Q(z)}{1+Q(z)(G(z)-G_m(z))}y_r(k) + \frac{1-G_m(z)Q(z)}{1+Q(z)(G(z)-G_m(z))}v(k) \tag{22}$$

In the case of a perfect model $G(z) = G_m(z)$, to ensure good tracking or rejection of disturbances it is enough to have:

$$Q(z) = \frac{1}{G_m(z)} \tag{23}$$

The above function constitutes the ideal theoretical controller. It can not be achievable as the degree of the denominator of the system transfer function is strictly greater than its numerator degree. Furthermore, positive zeros of the process or which have positive real part become poles for the controller being unstable, and the pure delays of the process will cause phase advances. Thus leads to impracticable controller design. To solve this problem a filter F is inserted to make the controller robust to any model mismatch due to disturbances $v(k)$, or any unwanted features. Consequently, the controller is chosen with the following form,

$$Q(z) = Q_0(z)F(z) \tag{24}$$

The procedure of constructing suitable IMC controller design is carried out through these rules:

- The zeros of Q_0 are selected as the poles of G_m.
- The poles of Q_0 can be derived as below:
  - The stable zeros of G_m with positive real part.
  - The inverses of unstable zeros of G_m with positive real part.
  - One pole at the origin for each zero with negative real part.

  This rule ensures the controller stability, in fact it guarantees that all the poles are inside the unit circle.
- A pole in the origin is added to Q_0.
- The gain of Q_0 requires checking the following condition:

$$Q_0(1)G_m(1) = 1 \tag{25}$$

- Include the filter F of the form:

$$F(z) = \frac{1-\alpha}{z-\alpha}\ ;\ 0<\alpha<1 \tag{26}$$

  This filter is considered as the most important block in IMC approach. It avoids the degradation of performances in the presence of model mismatch reflected by $y(k) - y_m(k)$. It provides softening the controller response to protect the loop devices.

V. Experimental results: Remote adaptive and internal model control

In this section, we'll focus on real-time application of remote control based on adaptive pole placement and IMC strategies. The standard protocol Bluetooth will be used as it is easy to adapt in practice and available in many devices [9], [10]. Bluetooth is oriented for wireless communication with a short range (from few meters until $100m$). It avoids direct exposure to dangerous areas with low power consumption (from less than $1mW$ until $100mW$). It is a wireless approach essentially used to control a far, inaccessible or mobile device which can not be directly operated. Experimental application is performed for two processes: first order system with variable delay, and level control process.

A. Hardware and software development

Figure 5 shows different blocks used for the remote implementation of pole placement and IMC controllers.

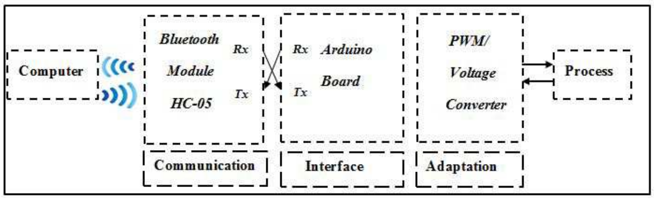

Fig. 5. Control loop components.

The used module is $HC-05$ adding a bluetooth communication function to an Arduino Uno card as depicted in figure 6. Thus, it can communicate wirelessly at medium distance with any other bluetooth device that is integrated into the computer in our case.

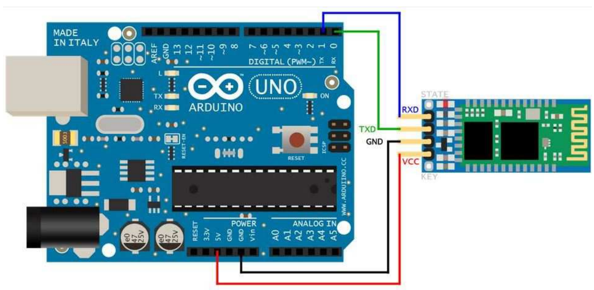

Fig. 6. $HC-05$ bluetooth module wiring.

Arduino is used as an acquisition card to convert the control signal received by the Bluetooth module $HC-05$ and computed by the computer into PWM (PulseWidth Modulation). It converts the analog output of the process into digital signal received by the computer via its Bluetooth device.

Since the Arduino board generates a PWM signal as output (it does not have analog outputs), to control systems its signal

Authorized licensed use limited to: Sami Shamoon College of Engineering. Downloaded on November 10,2023 at 12:44:14 UTC from IEEE Xplore. Restrictions apply.

must be converted to continuous one. For this purpose, an electronic board is designed to convert the PWM into DC voltage whose value is proportional to the duty cycle of the PWM signal over a range of 0 to $10V$.

A voltage bucker has been designed to decrease the value of a DC voltage applied to its input. Its gain is fixed to 0.5, in fact, the output voltage of the level control system, presented in the following, is from 0 to $10V$, while the Arduino card used for data acquisition supports $5V$ as maximum voltage value for its analog inputs.
To implement practical tests, MATLAB and Arduino have been used as software tools. two different systems are chosen: a first order process with variable delay and a level control system on which, respectively the adaptive control with RST structure and the internal model control IMC have been applied.

B. Practical results

1) First order system with variable time delay: Consider a first order system for wich the delay value can be manually changed. For the moment, its continuous model is:

$$H(p) = e^{-1.2p}\frac{0.6}{1+0.8p} \tag{27}$$

For time delay variation, we'll choose an instant on each its value will be changed. To supervise the dynamic evolution of the process, a graphical interface has been constructed as shown below,

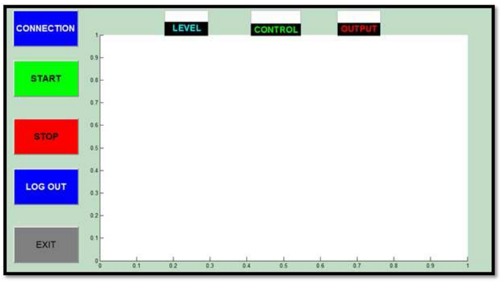

Fig. 7. Control interface.

In this interface, we use five buttons:

- **CONNECTION:** Pressing this button, the computer will be connected to the Bluetooth module $HC-05$. If the connection is established, a ringtone is heard.
- **START:** This button allows remote control of the system via Bluetooth.
- **STOP:** The function of this button is stopping the control loop.
- **LOG OUT:** Disconnect the computer and the Bluetooth Module $HC-05$.
- **EXIT:** By clicking this button the supervision interface will be closed.

The text fields that are above the static ones CONTROL and OUTPUT present respectively the control and the system output values at each sampling period. The text box LEVEL includes the desired level value in the case of level control process.
In this application, the Arduino board is considered as an acquisition card, the program loaded allows reading the control value received by the Bluetooth module $HC-05$ from the computer and write it on a PWM pin on the one hand, and remotely send the process output value to the computer on the other hand. The adaptive pole placement control startegy with RST structure is applied to the system. The hierarchical identification method is configured into the adaptive controller. Practical results are shown in figure 8. At the time iteration $k = 1000$, a changement of the delay value is applied.

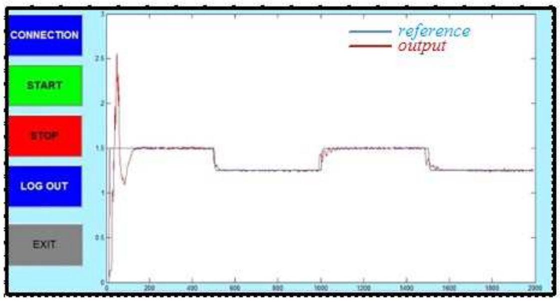

Fig. 8. Output and reference trajectories.

It can be clearly observed that the system output converges to the reference trajectrory despite the variation of the time delay value. This proves that adaptive remote control is a suitable strategy to fulfill good tracking performance for non-stationnary time delay systems.

2) Level control process: The system is one of the workstations of an educational model designed to meet a number of professional requirements. Each workstation consists in a closed loop including sensors and actuators. Figure 9 shows the level control system.

Fig. 9. Level control process.

Let's consider figure 10 to explain the operating principle

Authorized licensed use limited to: Sami Shamoon College of Engineering. Downloaded on November 10,2023 at 12:44:14 UTC from IEEE Xplore. Restrictions apply.

of the process. The pump $P101$ delivers a fluid from the

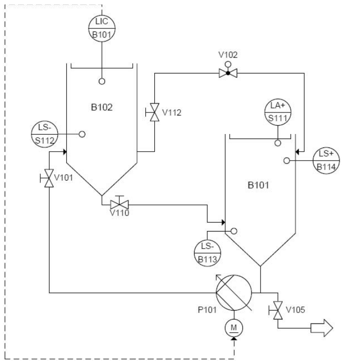

Fig. 10. Operating principle of level control system.

storage tank $B101$ to the tank $B102$ via a piping system. The fluid level in $B102$ is monitored with an ultrasonic sensor at the measurement point $LICB101$. The analog current signal $(4...20mA)$ of the ultrasonic sensor is connected to the measuring transformer which converts the current into a standard voltage signal $(0...10V)$.

To apply a disturbance, it is possible to partially or completely open/close the valve $V102$ to drain the fluid into the lower tank or close/open the manual drain valve $V110$. The characteristics of the pump $P101$ are:

- Flow: $22.5l/min$.
- Pressure: $15KPa/0.15bar$.
- Power: $26W$.
- Pipe diameter: $20mm$.
- Voltage: $24V$.

Before controlling the system, its step response has been generated. Note that it shows that the process includes a time delay of $4s$, which is ignored since the internal model controller is usually robust to model errors. Practical results of remote IMC implementation by Bluetooth to control the fluid level in the process are shown in figure 11. Thus, the desired fluid (water) level is online selected.

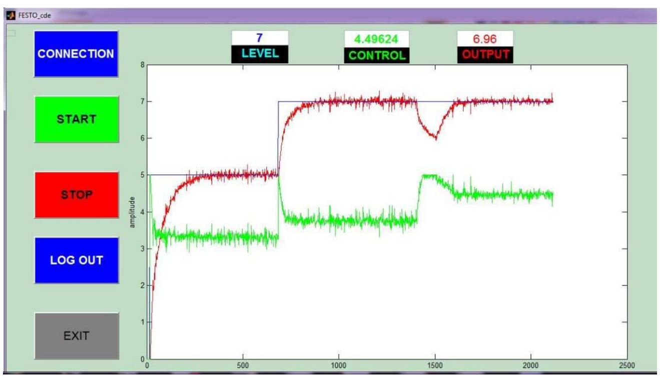

Fig. 11. IMC application: Output, reference and control evolutions.

We remark that the output of the system follows the desired level. In fact, in the beginning the reference level of water is $5L$, we observe that the output stabilizes at this level. Then we increase the setpoint to $7L$, the control value increases in order to adapt the output to the new desired level. Then, the output increases until achieving $7L$. At the time step $k = 1400$, a disturbance is applied to the process by opening the manual drain valve. So, the water level decreases, while the control value increases until returning the output to the desired level. Hence the performance of our application.

VI. Conclusion

In this paper, an application of remote adaptive and internal model control for time delay systems has been presented. We suggested the use of the hierarchical identification method to implement the adaptive pole placement controller. The wirelles communication protocol Bluetooth has been exploited. The obtained results with practical examples showed the effectiveness the approach.

Acknowledgment

This work was supported by the Ministry of the Higher Education and Scientific Research in Tunisia.

References

[1] S. Bedoui, L. Majda, and A. Kamel, *Gradient based iterative identification for discrete-time delay systems*, In the 9th IEEE Asian Control Conference (ASCC), pp. 1-6, 2013.

[2] L. J. Li, T. T. Dong, S. Zhang, X. X. Zhang, and S. P. Yang, *Time-delay identification in dynamic processes with disturbance via correlation analysis*, Control Engineering Practice, pp. 92-101, 2017.

[3] A. Yce, N. Tan, and D. P. Atherton, *Fractional Order PI Controller Design for Time Delay Systems*, IFAC-PapersOnLine, vol. 49, no. 10, pp. 94-99, 2016.

[4] C. C. Cheng, and C. H. Ho, *Design of adaptive sliding mode controllers for mismatched perturbed systems with application to underactuated systems*, In Chinese Control Conference (CCC), IEEE, pp. 1329-1336, 2017.

[5] I. S. Nadezhdin, A. G. Goryunov, and F. Manenti, *Fuzzy Adaptive Control System of a Non-Stationary Plant*, In IOP Conference Series: Materials Science and Engineering,2016.

[6] S. C. K. Junoh, L. Abdullah, N. A. Anang, Z. Retas, and S. N. S. Salim, , *Investigation on performance of Model Reference Adaptive Control (MRAC) and PI controller for machine tool application*, In Systems, Process and Control (ICSPC), pp. 17-22, 2016.

[7] C. E. Garcia, and M. Morari, *Internal model control. A unifying review and some new results*, Industrial and Engineering Chemistry Process Design and Development, Vol. 21, no. 2, pp. 308-323, 1982.

[8] L. J. Kere, S. Kelouwani, K. Agbossou, and Y. Dub. *Improving Efficiency Through Adaptive Internal Model Control of Hydrogen-Based Genset Used as a Range Extender for Electric Vehicles*. IEEE Transactions on Vehicular Technology, Vol. 66, no. 6, pp. 4716-4726, 2017.

[9] B. Champaty, S. K. Nayak, G. Thakur, B. Mohapatra, D. N. Tibarewala, and K. Pal, *Development of Bluetooth, Xbee, and Wi-Fi-Based Wireless Control Systems for Controlling Electric-Powered Robotic Vehicle Wheelchair Prototype*. In Classification and Clustering in Biomedical Signal Processing, pp. 356-387, 2016.

[10] D. Sullivan, W. Chen, and A. Pandya, *Design of remote control of home appliances via Bluetooth and Android smart phones*. In International Conference on Consumer Electronics-Taiwan (ICCE-TW), pp. 371-372 , 2017.

Authorized licensed use limited to: Sami Shamoon College of Engineering. Downloaded on November 10,2023 at 12:44:14 UTC from IEEE Xplore. Restrictions apply.