

Use of Educational Tool for Modeling Temperature Control Systems as a Remote Teaching and Learning Strategy in Engineering

Wânderson O. Assis, Alessandra D. Coelho

Abstract—This paper presents the use of a teaching tool that allows the model identification of a temperature control system by applying several methods such as Step Response Method, Least Squares Method, Moving Average Least Square Method and Weighted Least Square Method. The system is an application of remote laboratory access, a WebLab, with real equipment and systems that can be remotely controlled by the web. The WebLab allows the development of multidisciplinary experiments, having as a differential, the possibility of allowing technology-mediated teaching, aiming to get a better assimilation of the contents presented in the theory for effective learning.

Index Terms—Education, remote laboratories, control systems, system identification, modelling control systems.

I. INTRODUCTION

IN nearly 2020, the world was surprised by an alert regarding the pandemic of a new virus, which was named COVID-19.

Needing to interrupt room classes to avoid the virus' proliferation, the vast majority of public and private schools in Brazil, especially in higher education, started to use digital technologies in order to continue the teaching-learning process. So, all of a sudden, even the most resistant teachers to online education began to use it as an alternative to continuing classes.

With the change, there was a revolution in the way we use technologies. The classes popularity was greatly increased with videoconference and distance classes, which started to be carried out by using teaching resources. The fact is that, in the world, higher education in general will never be the same after the pandemic [1]. During this period of social distancing, new teaching and learning alternatives based on the use of technologies became part of teachers' daily lives. Some features that have been employed as alternatives include Virtual Environment Learning, videoconferencing

applications, videos and audios stored in the cloud (lives, recorded lessons, use of whiteboard, podcasts), software simulations, remote laboratories (WebLabs), learning activities with virtual group classes, in addition to interactive resources such as quiz, games, among others. There is a multitude of educational opportunities and resources available. Often, the main difficulty is to be able to define the alternative that best fits the topic to be addressed in the specific discipline or course. Additionally, it is essential to assess the impact of applying these new teaching strategies on learning [2]. Factors such as motivation, physical interaction, advanced technological resources, continuous feedback, and proximity in teaching are implications that must be considered for improvement in a teaching and learning process that meets quality education.

Among the available technologies, especially for content that demand the use of practical resources, there is the use of remote laboratories. The WebLabs are clear examples of computing use in education, but with a large differential, because they provide a possibility of real learning by distance supervision. This is possible, mainly because the WebLabs allow the development of practical experiments by accessing the equipment through the net in real laboratories and thus allowing control and data acquisition in real-time.

WebLabs have been implemented in several institutions presenting solutions for remote operation, generally using commercially available software or dedicated networks. Several researchers reported their experience in using WebLabs for teaching chemistry [3], electronics [4]-[7], microwave [8], process control [9], among others [10], [11]. At the Instituto Mauá de Tecnologia (IMT), several WebLabs related to the area of process control were recently developed [12]-[20]. Areas such as robotics [21], oil industry, and image processing [22] were also covered.

In this sense, the objective of this work is to present a didactic tool for teaching in the area of "Process Control" describing some of the experiments that can be carried out in order to understand the different ways to implement techniques for identifying systems. The chosen application is a didactic temperature control system that presents resources available for teaching control in engineering courses. The main differential of this work in relation to the proposed in [15] is the adaptation of the tool for better remote use, as well as the introduction of several other resources that provide a very comprehensive approach to several techniques for

Manuscript received July 13, 2022; revised January 18, 2023

This work was supported by the Instituto Mauá de Tecnologia which provides resources for the construction of remote laboratories in the research project, in addition to offering scholarships to students who worked on the project.

W. O. Assis is a professor of Electrical Engineering Department, Instituto Mauá de Tecnologia, 09080-320, SP, Brazil (e-mail: wanderson.assis@maua.br).

A. D. Coelho is a professor of Electrical Engineering Department, Instituto Mauá de Tecnologia, 09080-320, SP, Brazil (e-mail: alessandra.coelho@maua.br).

system identification.

The use of identification techniques, such as the Least Squares Method, Recursive Least Square Method [16], Kalman Filters [17], Auto Regressive Moving Average method for systems with exogenous inputs, among others, are adopted in several works, being an interesting alternative to allow the implementation of adaptive control systems [18], [19].

In this work, it is intended to evaluate the performance of the Step Response Method, Least Squares Method, Moving Average Least Square Method and Weighted Least Square Method applied in a temperature control system.

II. REMOTE LABS (WEBLABS)

Due to the advancement of technologies, most industrial control systems are carried out using digital technologies, typically industrial controllers or computerized systems, where the control is executed through computational algorithms. This characteristic is propagated in the academic environment, therefore the vast majority of educational kits aimed at the area of "Process Control" use microcontrolled systems or data acquisition systems connected to computers. Didactic software is used to implement the control algorithm as well as to change the controller parameters, in order to allow the designer to evaluate the dynamic behavior of the systems, resulting from these changes.

The concept applied to WebLabs at IMT can be generalized by the block diagram in Fig. 1, and presents the following elements:

- the system or equipment which will be remotely controlled;
- sensor monitoring system, which measures process variables in real time; it may include electronic signal conditioning circuits and transducers;
- data acquisition system that captures signals at the appropriate sampling frequency and makes the data available to the WebLab system; these variables can be viewed (graphically or numerically) in the user interface;
- actuators and drivers, that is necessary when the designer wants not just to monitor the variables, but also to control them; drivers are electronic circuits used to convert control signals to appropriate levels depending on the application;
- control algorithm, which can be a conventional controller normally adopted in control applications (for example, a PID control) and/or algorithms that allow

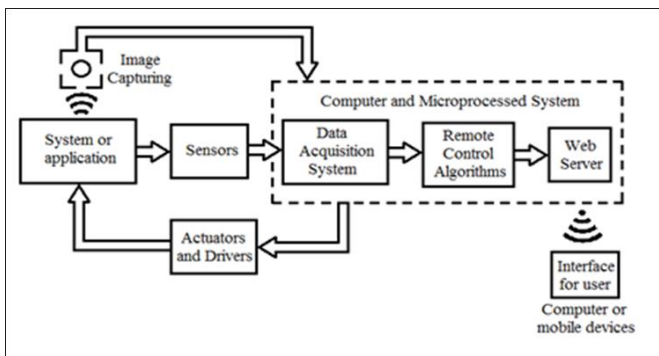


Fig. 1. Block diagram of Weblabs developed at IMT.

processing information and defining actions according to the values of the measured variables;

- image capture system, which collects images in real-time;
- web server, responsible for providing system information (sensor signals, equipment images, etc.) in the remotely accessible interface by the user;
- user interface, is how the user accesses the remote experiment over the internet, which can be done through a computer terminal, mobile device, HMI (Human-Machine Interface), among others.

The main limitation in the use of these remote labs by students is the need to use the LabVIEW™ software, which requires users to install a plugin so that the complete interface can be viewed.

Another limitation is the need to leave the equipment in continuous operation, during the day and night, usually for a large number of days. This brings flexibility to the use of the application by the user but makes the solution unsustainable. This limitation can be minimized if the equipment is put into operation just on demand, that is, when someone is interested in remotely controlling the equipment or when the teacher proposes an activity or a project using WebLab. Thus, unnecessary energy consumption is avoided, making solutions more sustainable and avoiding the need to keep a technician continuously monitoring the system's operation to put it back into operation in case of technical problems (for example, power outage or drop in internet signal). Another interesting alternative is accessing the laboratory remotely, using virtual machine concepts, for example, during an online class [13].

Using WebLab, several people can access the same experiment simultaneously, although just one of them can interact. In any case, when another student is controlling the equipment, it is possible to visualize the results. In addition, the student can request access to the experiment. WebLabs have a system that manages access control, limiting each student a maximum time of use and releasing the equipment to other students, as long as they have requested access.

III. TEMPERATURE CONTROL SYSTEM

The process of the control system, illustrated in Fig. 2, is a didactic heating system, built with wood, including a lamp, where the heating is controlled by adjusting the applied voltage. A thermocouple (PT100) is used as a temperature sensor and provides a voltage proportional to temperature. An electrical panel including electronics signal conditioning circuits provides the conversion of voltage levels of sensor voltage, whereas a solid-state relay produces adequate power in the lamp based on a control signal produced by a data acquisition system. Cooling is produced by the control voltage applied to a cooler.

The temperature control system was implemented by illustrated at the block diagram in Fig. 3. The control was developed using ELVIST™ (Educational Laboratory Virtual Instrumentation Suite) toolkit (Fig. 4) and LabVIEW™ programming language with a graphical man-machine interface being constituted of one data acquisition board model NI PCI-6251 and one work station equipped with one protoboard where the applications can be developed.



Fig. 2. Temperature Control System.

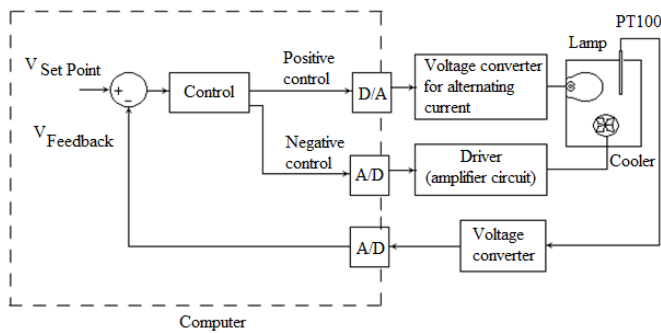


Fig. 3. Temperature Control System - Block diagram

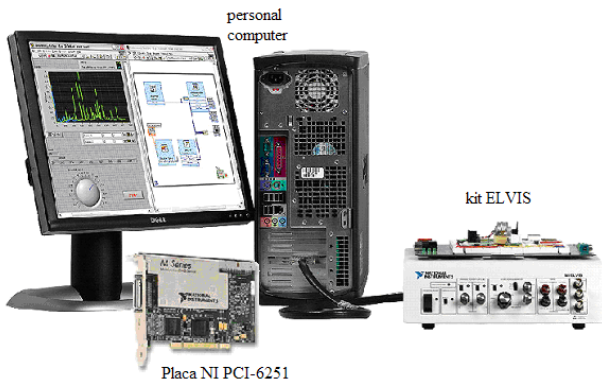


Fig. 4. Data Acquisition System ELVIS™

IV. MODELLING TEMPERATURE CONTROL SYSTEM

The temperature control system can be modelled by the following differential equation [23]:

$$RC \frac{d\theta_0}{dt} + \theta_0 = Rh + \theta_i, \quad (1)$$

where R is the thermal resistance ($^{\circ}C s / kcal$), C is the thermal capacitance ($kcal / ^{\circ}C$), θ_0 and θ_i are respectively the deviation of temperature output and input over an operating point in steady state ($^{\circ}C$) and h is the deviation of heating over the operating point in steady state ($kcal/s$).

Applying the Laplace Transform and assuming $\theta_0(0) = 0$ results:

$$\theta_0(s) = \frac{R}{RCs+1}H(s) + \frac{1}{RCs+1}\theta_i(s), \quad (2)$$

whereas the variation of outside air temperature is a disturbance $\theta_i = 0$; ignoring the dynamics of the temperature sensor and assuming the voltage produced by this sensor $V_0(s)$ is proportional to the temperature, i.e., $V_0(s) = K_0\theta_0(s)$, and assuming the heating is proportional to the output voltage produced by the data acquisition system, i.e., $H(s) = K_i V_i(s)$, we have:

$$V_0(s) = \frac{K_0 K_i R}{RCs+1} V_i(s) = \frac{K_S}{T_S s + 1} V_i(s), \quad (3)$$

where K_S is static gain and T_S is the time constant of the system. Note that the system has a heating and cooling system separately driven, respectively to positive and negative control signals $V_i(s)$ as illustrated in Fig. 3.

The discrete state space of (3), using sampling time T results is:

$$v_0(K+1) = e^{-\frac{1}{T_S}T} v_0(K) + \left(1 - e^{-\frac{1}{T_S}T}\right) K_S v_i(K). \quad (4)$$

A. Identification by Step Response Analysis

The temperature control system (3) is a first-order model and its parameters can be determined by the step response, as illustrated in Fig. 5, where it is considered, in the specific case, the initial condition $V_i(0) = 0$ V and $V_i(t) = V_{max}$ para $t \geq 0$ s. The parameter K_S can be obtained by the Final Value Theorem whereas T_S corresponds to the time required for the output to reach 63.2% of the output variation after the step applied and up to the steady state.

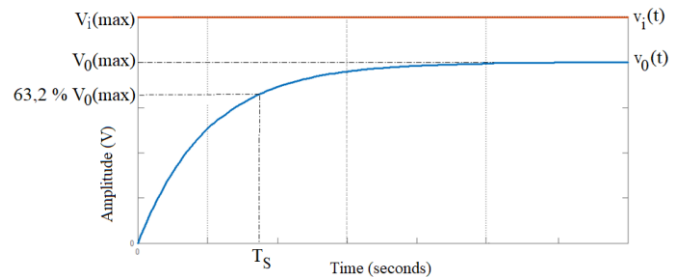


Fig. 5. Step Response of First Order System

B. Identification by Least Squares Method and Moving Average Least Squares Method

Consider a discrete system that can be represented by the equation (5), which is an ARX model (Autoregressive Model With Exogenous Inputs), but as matrix inputs dependent on $v_i(k)$ and $B(k)$: in which $A_y(q) v_0(k)$ represents the autoregressive portion, $e(q)$ is the error represented by a moving average of the white noise and $B(q) v_i(k)$ represents an external input [24].

$$A_y(q)v_0(k) = B(q)v_i(k) + e(k) \quad (5)$$

Assuming that the discrete system (4) is an ARX model, equation (4) can be adjusted in a specific case to (6), considering the first-order system and where $A(q)$ and $B(q)$ represent the parameters of the system:

$$v_0(k+1) = A(q)v_0(k) + B(q)v_i(k) + e(k) \quad (6)$$

Comparing the equations (4) and (6) results:

$$v_0(K + 1) = W(k)^T D(k)^T + e(k) \\ = D(k)W(k) + e(k) \quad (7)$$

Defining $W(k)$ the parameters matrix of the system, where $D(k)$ and $Y(k) = v_0(k+1)$ are the measured values considering certain input values at each sample, resulting:

$$v_0(K + 1) = W(k)^T D(k)^T + e(k) \\ = D(k)W(k) + e(k) \quad (8)$$

So, it is defined as matrix $D(k)$:

$$D(k) = [v_0(k) \quad v_i(k)]; W(k) = \begin{bmatrix} e^{\frac{-1}{T}T} \\ \left(1 - e^{\frac{-1}{T}T}\right) K_S \end{bmatrix} \quad \text{and}$$

$$Y(k) = v_0(k + 1).$$

If it is desired to minimize the error $e(k)$ and estimate the parameters that produce the minimum error given by $e(k) = v_0(k + 1) - D(k)W(k)$ we can define the cost function (9) to minimize the squared error [24], [25]:

$$J = \sum_{i=1}^n e(k)^2 \quad (9)$$

Using (8):

$$J = (Y(k) - D(k)W(k))^T (Y(k) - D(k)W(k)) \quad (10)$$

The equation (10) results:

$$J = Y(k)^T Y(k) - W(k)^T D(k)^T Y(k) \\ - Y(k)^T D(k)W(k) + W(k)^T D(k)^T D(k)W(k) \quad (11)$$

In order to find the minimum value and deriving and equating to zero results in obtaining the \hat{W} of equation (12) which represents the W parameters of the system estimated from collected data to minimize errors and inaccuracies in measurements.

$$\hat{W} = (D(k)^T D(k))^{-1} D(k)^T Y(k) \quad (12)$$

The identification can be obtained with relative precision depending on the degree of imprecision in the collected data. In any way, better accuracy is obtained, if a larger dataset is considered. However, the main inconvenience of the methodology is the need to accumulate an adequate number of samples and just after collecting these samples perform the calculation of the \hat{W} parameters. An alternative to allowing determining the calculations in each sample is the use of a moving data set (Moving Average Least Squares Method), performing the calculation for a defined set of collected data. That is, for each sample collected, the measurement is added to the data set, and the sample is removed from the set resulting in FIFO (First In First Out) structure.

C. Identification by Weighted Least Squares Method

Other methods can be used for mathematical model identification of the discrete system (7), the advantage of getting a real-time application and defining the parameters at each sampling time [24], [26]. An example is the Weighted Least Squares Method (WLS Method) an extension of the

Least Squares Method that is an efficient and accurate estimation algorithm suitable particularly when the designer has the a priori approximation of parameters values.

It should be noted that the estimation in real-time allows the determination of the parameters considering the variation of operating conditions, for example, due to external disturbances. Additionally, it makes possible the implementation of an adaptive control system where the controller parameters can be automatically adjusted in real-time by the estimated mathematical model.

The main feature of the WLS Method is the use of initial values for the parameters and the determination of errors in real-time. To achieve convergence, it uses a gain matrix (Kalman gain) which is automatically adjusted in the algorithm to produce the minimization of errors, considering the previously calculated parameters and those obtained in the current sampling. This is accomplished by adjusting the “forgetting factor” γ and the variable α . By adjusting γ , the user is given greater importance (higher degree of weighting) to the latest measurements, as these contain the most current information and should have greater influence in the estimation. This is enforced in practice by setting $0.9 < \gamma < 1$.

The structure of the estimation algorithm is presented below.

(a) Choose a diagonal weighting matrix $P(k)$, known as the “covariance matrix” with order $n \times n$, where n is the number of variables representing the system states whose parameters will be estimated. $P(k)$ reflects the degree of uncertainty in the knowledge of the estimated parameters. However, if there is some knowledge about the amounts expected in the estimation, we use a diagonal matrix $P(k)$ with parameters set at low magnitude values. However, if the estimated parameters are completely unknown, the “covariance matrix” should present terms with high values on the main diagonal.

(b) Set a “forgetting factor” γ such that: $0 < \gamma \leq 1$ and $\alpha = (1 - \gamma)$.

(c) Initialize $V_0(k), V_0(k-1), V_0(k-2), V_i(k-1), V_i(k-2)$ and $D(k) = [-V_0(k-1) -V_0(k-2) \quad V_i(k-1) \quad V_i(k-2)]^T$.

(d) Assume initial values for $\hat{W}(k + 1)$ and $\hat{W}(k)$ where $\hat{W}(k)$ is the array of estimated parameters at each sampling time.

Repeat the operations listed from (e) to (i) while $\max |\hat{W}(k+1) - \hat{W}(k)| \geq \varepsilon$ where ε is a tolerance for the estimation error. However, if the goal is to develop a control based on the estimated parameters, then the algorithm should be used continuously in each sample.

(e) Update $\hat{W}(k) = \hat{W}(k+1)$.

(f) Calculate the Kalman gain of the system by:

$$K(k) = \frac{P(k)}{\gamma} D(k) \left(\frac{1}{\alpha} D(k)^T \frac{P(k)}{\gamma} D(k) \right)^{-1}, \quad (13)$$

(g) Estimate a new value of $\hat{W}(k+1)$ using:

$$\hat{W}(k + 1) = \hat{W}(k) + K(k)(V_0(k) - D(k)^T \hat{W}(k)), \quad (14)$$

(h) Calculate the new weighting matrix by:

$$P(k) = \frac{1}{\gamma} [I - K(k)D(k)^T]P(k), \quad (15)$$

where I is an identity matrix with the same dimension as matrix A of the system.

(i) Update: $V_0(k) = V_0(k + 1)$; $V_0(k-1) = V_0(k)$; $V_0(k-2) = V_0(k-1)$; $V_i(k-1) = V_i(k)$; $V_i(k-2) = V_i(k-1)$ and $D(k) = [V_0(k-1) V_0(k-2) V_i(k-1) V_i(k-2)]^T$.

The tolerance setting is made just if the goal is to estimate the system parameters to achieve their convergence. However, as mentioned before, if the goal is to identify the parameters in real-time, the algorithm, considering the operations (e) to (i) can be repeated continuously.

V. RESULTS AND DISCUSSION

This section presents the approaches developed for the application and the results obtained through practical tests compared with simulation results.

A. WebLabs for Temperature Control System Identification

The main machine interface to identify the model of the temperature control system was developed by using LabVIEW™. Basically, three steps were required: (i) create LabVIEW™ application as an appropriate main-machine interface to the user; (ii) enable the *WebServer* option to obtain remote access and thus configure the access managing panel; (iii) modify the *HTML* code, inserting text containing introduction and fundamentals, specific information about the school and general instructions to the user.

If you are interested in making full remote access to the application, the user first needs to install a plugin file for access to the Web Server. In this sense, these instructions are informed by the teacher to the student who wants to install the system on their computer, with a different plugin

depending on the operating system.

If the access will be through a virtual machine, the student will need to install a tool to allow the access to be done remotely. There are several tools available on the market, such as TeamViewer, Chrome Remote Desktop, Getscreen.me, Microsoft Remote Desktop, AnyDesk, DameWare Mini Remote Control, and GoToMyPC (LogMeIn). In teaching “Process Control” the TeamViewer tool has been used, which works on desktops and mobile devices with Windows, MacOS, Linux, Android, iOS, Chrome OS, and Raspberry Pi.

Using the applications, with interfaces as presented in Fig. 6, Fig. 7, and Fig. 8, it is possible to carry out the identification of the system by respectively the following methods:

- a) Identification by Step Response Analysis and Identification by Least Squares Method (simultaneously);
- b) Identification by Moving Average Least Squares Method;
- c) Identification by Weighted Least Squares Method.

In the first algorithm, using the interface presented in Fig. 6, the first step is to define the sampling time (typically $T = 1$ s) and the data set that will be used to determine the mathematical model (number of samples). Then, it is necessary to define the initial value of the input (set point). Avoid keeping this value continuously at zero as the least squares method can result in indefiniteness or error, due to the division by zero. After that, the user must click on the **START** button and then change the set point value to start the step response test. Once this is done, the user needs to wait until the test is concluded and the output approaches its steady-state value, as illustrated in Fig. 5 and in the graph in Fig. 6. When this occurs, the user must click on the **END** button and then the model deduced by the step response method will be determined and displayed in the interface.

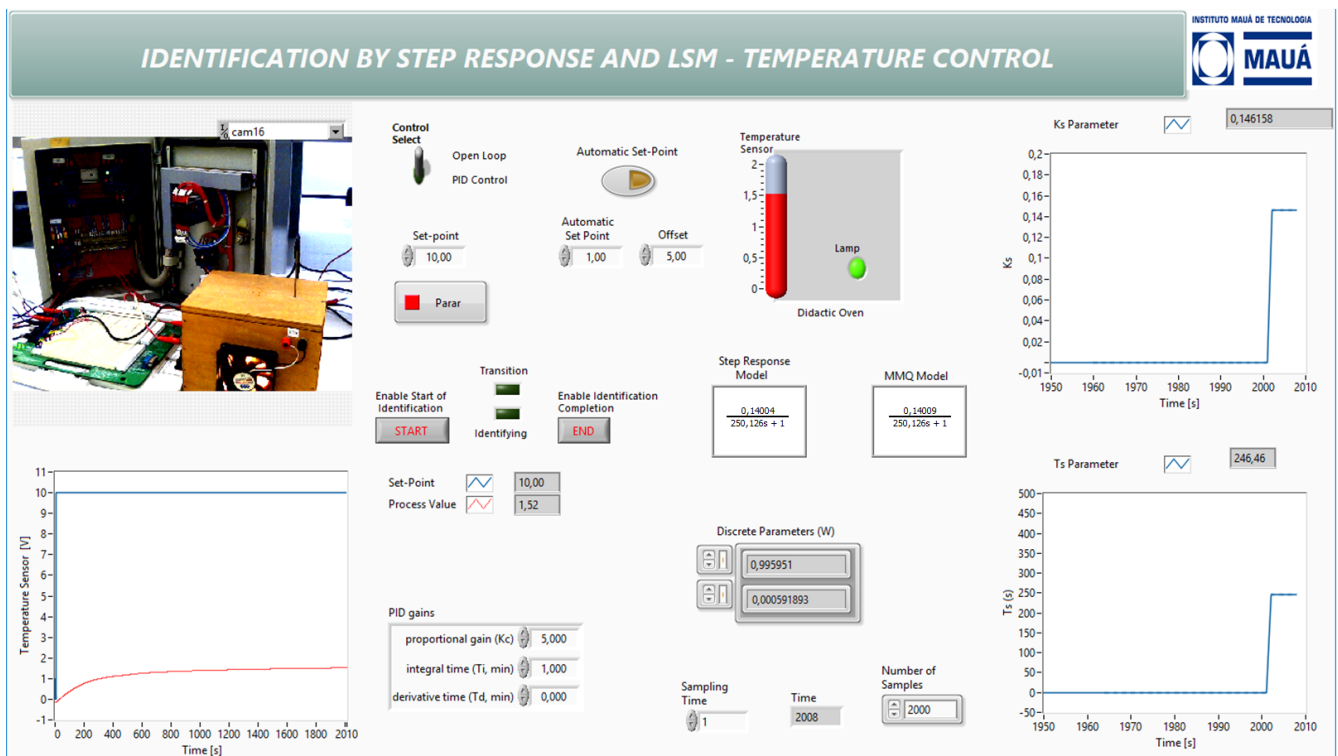


Fig. 6. Interface for Temperature Control Model Identification - Step Response and Least Squares Method Methods

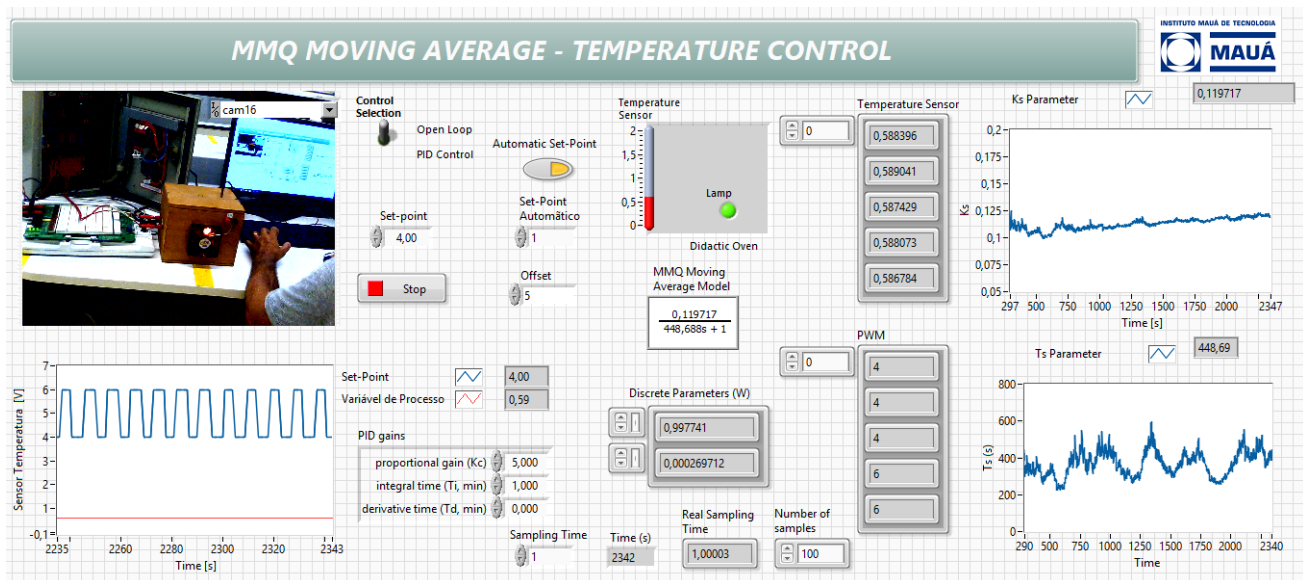


Fig. 7. Interface for Temperature Control Model Identification - Step Response and Least Squares Method Methods

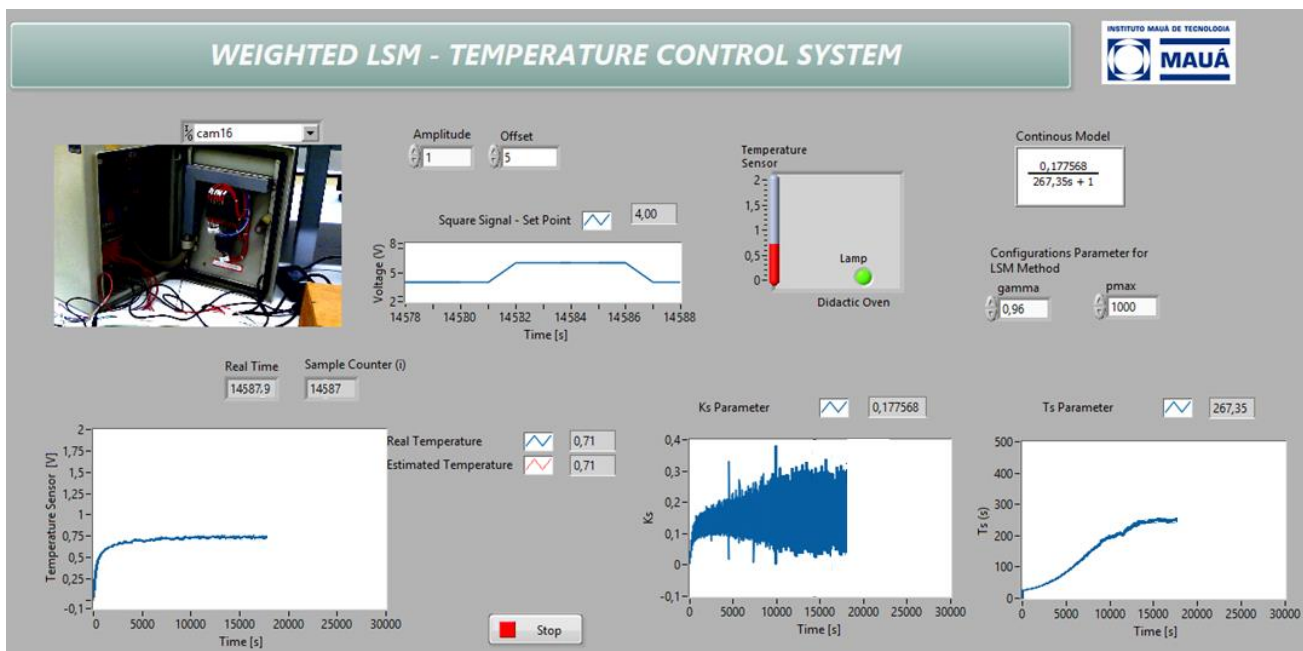


Fig. 8. Interface for Temperature Control Model Identification - Step Response and Least Squares Method Methods

For this case, the results of the step response method will be displayed just after several samples are collected. However, it is necessary to pay attention to the following requirements:

- the test must be started with the system at low temperature, with a variation among a considerable amplitude being applied at the input (a variation of at least 1 V) and the variation at the output being observed;
- the maximum and minimum amplitudes at the input and output cannot exceed the limits of ± 10 V;
- the number of samples must be established as greater than or equal to 2000 in order to obtain an adequate transient to carry out the identification of system parameters, whether for heating or cooling.

Simultaneously, at the same application (Fig. 6), the model obtained by the Least Squares Method (LSM) will also be displayed, once the defined number of samples has been collected. The interface shows the calculated parameters K_s

and T_s obtained by this method (numerically and graphically). In any form, the user will also need to define the sampling time adopted in the application and the data set that will be used to determine the mathematical model.

If you prefer, it is possible to enable an Automatic option, which uses a square wave with input voltage ranging between maximum and minimum values and using a set frequency ten times lower than the sampling frequency.

Finally, the application provides a real image of the experiment. Moreover, a switch allows selecting PID control. However, this option is not addressed in this work.

The procedure of using the LMS Moving Average (Fig. 7) is simpler. The user just needs to define the Sampling Time, the number of samples, and the Set Point, varying its value as desired directly in the interface, or using the Automatic option, which uses a square wave with input voltage ranging between maximum and minimum values and using a set

frequency ten times lower than the sampling frequency (this value can be adjusted at algorithm). The calculation will be performed at each sampling time, with the values calculated for each sample collected.

The procedure using the WLSM method (Fig. 8) is similar. The difference is that the user also needs to set the γ (gamma) and $pmax$ parameters. The parameter $pmax$ represents the main diagonal terms of the diagonal matrix $P(k)$. The calculation for parameter identification is performed at each sampling time, using a moving data set sample collected, but keeping the specified number of samples.

In both cases mentioned above, the calculation will be performed at each sampling time, with the values obtained for each sample collected.

B. Experimental and Simulation Results

For comparative purposes, the Step Response Method was implemented in a real system using the application illustrated in Fig. 6 to obtain the model presented in (16).

$$V_0(s) = \frac{Ks}{T_s s + 1} = \frac{0,140090}{250,126233s + 1} V_i(s) \quad (16)$$

This method was chosen because it produces a relatively accurate model when using a suitable dataset. This model will be considered in the simulations as a "reference model" and used to evaluate the performance and results obtained by the other methodologies. In practical tests, the model was automatically identified and the results can also be compared with the same reference model in order to get the deviation. Table 1 and Table 2 present a summary of the parameters obtained in the simulations and practical tests respectively, where:

- SRM – Step Response Method (reference)– test with step input with amplitude ranging from 0 to 10 V and duration of 2000 seconds; the practical result obtained by SRM (reference model) is presented in Fig. 6;

- LSM(1) – Least Square Method - test with step input with amplitude ranging from 0 to 10 V, duration of 2000 seconds, and calculation of parameters performed after these 2000 samples; the practical result obtained by LSM(1) is also presented in Fig. 6;

- LSM(2) – Least Square Method - test with step input with amplitude ranging from 0 to 10 V, duration of 2000 seconds, and calculation of parameters performed after each 100 samples;

- LSM(3) – Least Square Method - test with oscillatory input with amplitude varying from 4 to 6 V at a frequency of 0.1 Hz, duration of 2000 seconds, and calculation of parameters performed after every 100 samples;

- MALSM – Moving Average Least Square Method - test with oscillatory input with amplitude varying from 4 to 6 V at a frequency of 0.1 Hz, duration of 2000 seconds, and calculation of the first parameters performed after 100 samples; the other parameters are calculated after every sample; the practical result obtained by MALSM is presented in Fig. 7;

- WLSM(1) – Weighted Least Square Method - test with oscillatory input with amplitude varying from 0 to 4 V at a frequency of 0.1 Hz, duration of 2000 seconds after

convergence, parameters adjusted: $\gamma = 0,96$ and $P(k)$ with diagonal terms 10000;

- WLSM(2) – Weighted Least Square Method - test with oscillatory input with amplitude varying from 0 to 4 V at a frequency of 0.1 Hz, duration of 2000 seconds after convergence, parameters adjusted: $\gamma = 0,7$ and $P(k)$ with diagonal terms 10000;

- WLSM(3) – Weighted Least Square Method - test with oscillatory input with amplitude varying from 0 to 4 V at a frequency of 0.1 Hz, duration of 2000 seconds after convergence, parameters adjusted: $\gamma = 0,96$ and $P(k)$ with diagonal terms 1000;

- WLSM(4) – Weighted Least Square Method - test with oscillatory input with amplitude varying from 4 to 6 V at a frequency of 0.1 Hz, duration of 2000 seconds after convergence, parameters adjusted: $\gamma = 0,96$ and $P(k)$ with diagonal terms 10000;

- WLSM(5) – Weighted Least Square Method - test with oscillatory input with amplitude varying from 4 to 6 V at a frequency of 0.1 Hz, duration of 2000 seconds after convergence, parameters adjusted: $\gamma = 0,7$ and $P(k)$ with diagonal terms 10000;

- WLSM(6) – Weighted Least Square Method - test with oscillatory input with amplitude varying from 4 to 6 V at a frequency of 0.1 Hz, duration of 2000 seconds after convergence, parameters adjusted: $\gamma = 0,96$ and $P(k)$ with diagonal terms 1000; the practical result obtained by WLSM(6) is presented in Fig. 8.

The results from WLSM(1) to WLSM(3) and WLSM(5) are not presented in Table 2, because the practical test did not present convergence.

All results consider a sampling time of $T = 1$ s and the maximum, minimum, and average values are calculated. In the case of longer tests (WLSM), the calculations of parameters were performed considering the last 2000 samples collected after convergence. In this case, the approximate number of samples needed for convergence is also presented in Table 1 and Table 2. The following initial values considered were $W = [0,96329 ; 9,9082 \times 10^{-4}]$, resulting $Ks = 0,0269905$ and $Ts = 26,73742$ s.

Convergence was considered when the measurements presented values within a tolerance range with a modulus less than or equal to $\pm 1\%$ close to the stabilization value. For the practical test, the stabilizing parameters are observed in different ranges. In this case, we assume as convergence the values within a tolerance range with a modulus less than or equal to $\pm 1\%$ close to the stabilization value either for the upper or lower contours.

The simulation results allow the following discussions:

a) All results showed efficient identification and reduced deviation, indicating an opportunity to apply all methods.

b) The main disadvantage of using the SRM method is the need to submit the system to a step-on input, which may not be feasible in some applications. Additionally, it takes considerable time for the signal to stabilize to enable the parameters to be determined.

c) The LSM method allows the reduction of the necessary time for identifying the system parameters, just using a few samples. However, the method is not indicated in situations where there is no variation in the input and when the output is stable, which can produce considerable inaccuracies.

TABLE I
SIMULATION RESULTS

Methods \ Parameters	K_s	Absolute Deviation	T_s (s)	Absolute Deviation (s)
SRM	0,140042	-48×10^{-6}	250,126233	---
LSM(1)	0,140090	---	250,126233	---
LSM(2)	Average: 0,140090	---	Average: 250,125871	-362×10^{-6}
	Maximum: 0,140090	---	Maximum: 250,126237	4×10^{-6}
	Minimum: 0,140089	1×10^{-6}	Minimum: 250,120736	-5.497×10^{-3}
LSM(3)	Average: 0,140090	---	Average: 250,126233	---
	Maximum: 0,140090	---	Maximum: 250,126233	---
	Minimum: 0,140090	---	Minimum: 250,126233	---
MALSM	Average: 0,140090	---	Average: 250,126233	---
	Maximum: 0,140090	---	Maximum: 250,126233	---
	Minimum: 0,140090	---	Minimum: 250,126233	---
WLSM(1)	Average: 0,140088	-2×10^{-6}	Average: 250,126231	-2×10^{-6}
	Maximum: 0,140136	46×10^{-6}	Maximum: 250,126233	---
	Minimum: 0,139612	-478×10^{-6}	Minimum: 250,125680	-551×10^{-6}
	Number of samples until convergence: 54		Number of samples until convergence: 73	---
WLSM(2)	Average: 0,140090	---	Average: 250,126233	---
	Maximum: 0,140100	10×10^{-6}	Maximum: 250,126233	---
	Minimum: 0,139886	-204×10^{-6}	Minimum: 250,126054	-179×10^{-6}
	Number of samples until convergence: 29		Number of samples until convergence: 37	
WLSM(3)	Average: 0,140086	-3×10^{-6}	Average: 250,117447	-8786×10^{-6}
	Maximum: 0,140101	11×10^{-6}	Maximum: 250,126233	---
	Minimum: 0,139015	-1075×10^{-6}	Minimum: 247,576412	-2.549821
	Number of samples until convergence: 108		Number of samples until convergence: 109	
WLSM(4)	Average: 0,140300	210×10^{-6}	Average: 248,661922	-1.464311
	Maximum: 0,141490	1400×10^{-6}	Maximum: 249,362239	-0.763994
	Minimum: 0,139242	-848×10^{-6}	Minimum: 247,636940	-2.489293
	Number of samples until convergence: 9816		Number of samples until convergence: 11724	
WLSM(5)	Average: 0,140255	165×10^{-6}	Average: 248,343654	-1.782579
	Maximum: 0,141506	1416×10^{-6}	Maximum: 248,912988	-1.213245
	Minimum: 0,139149	-941×10^{-6}	Minimum: 247,625883	-2.50035
	Number of samples until convergence: 14981		Number of samples until convergence: 18924	
WLSM(6)	Average: 0,140333	243×10^{-6}	Average: 248,349385	-1.776848
	Maximum: 0,142151	2061×10^{-6}	Maximum: 248,916889	-1.209344
	Minimum: 0,138689	-1401×10^{-6}	Minimum: 247,633954	-2.492279
	Number of samples until convergence: 13849		Number of samples until convergence: 18933	

Therefore, the results obtained in LSM(3) are significantly better than those obtained in LSM(2).

d) The MALSM method is a better alternative, as it allows the determination of parameters after each sample, being designated for identification of systems that present the parameters variation, for example, when resulting in changes

in operating conditions. However, it has the same limitations mentioned in the LSM method. In the test carried out with variation in the input signals and with the absence of noise and inaccuracies in the measurements, it was the method that presented the best result.

e) The WLSM proved to be an excellent alternative,

TABLE II
PRACTICAL RESULTS

Methods \ Parameters	K_s	Absolute Deviation	T_s (s)	Absolute Deviation (s)
SRM	0,140090	(reference model)	250,126233	(reference model)
LSM(1)	0,146182	0,006092	246,475224	-3,651009
LSM(2)	Average: 0,100838	-0,039252	Average: 224,249508	-25,876725
	Maximum: 0,142732	0,002642	Maximum: 907,329291	657,203058
	Minimum: 0,082644	-0,057446	Minimum: 15,681797	-234,444436
LSM(3)	Average: 0,094957	-0,045133	Average: 325,185226	75,058993
	Maximum: 0,106209	-0,033881	Maximum: 431,056943	180,93071
	Minimum: 0,077448	-0,062642	Minimum: 258,737916	8,611683
MALSM	Average: 0,112899	-0,027191	Average: 369,825797	119,699564
	Maximum: 0,123043	-0,017047	Maximum: 597,040448	346,914215
	Minimum: 0,098985	-0,041105	Minimum: 224,454322	-25,671911
WLSM(4)	Average: 0,152507	0,012417	Average: 220,832099	-29,294134
	Maximum: 0,423520	0,283430	Maximum: 237,775547	-12,350686
	Minimum: 0,003272	-0,136818	Minimum: 208,739914	-41,386319
	Number of samples until convergence: 11750		Number of samples until convergence: 13179	
WLSM(6)	Average: 0,153380	0,01329	Average: 242,853094	-7,273139
	Maximum: 0,313257	0,173167	Maximum: 251,138825	-1,012592
	Minimum: 0,030078	-0,110012	Minimum: 236,522510	-13,603723
	Number of samples until convergence: 10182		Number of samples until convergence: 13333	

through adequate adjustment of the parameters. As the values of the parameters assumed as initial are relatively far from the ideal, the high $P(k)$ adjustment results in an improvement in the response time to convergence. Additionally, the use of a smaller γ allows assigning greater weight to the last measurements, which also contributes to the fast convergence, in addition to allowing the use of the method in applications where it is desired to quickly detect eventual changes in the system parameters. However, it was observed that the algorithm performs much better when there is a variation in the input next to zero. This is proven in the simulations shown in Fig. 9 and Fig. 10 which show the convergence of parameters through the WLSM simulations under the conditions defined in WLSM(2) and WLSM(6). Note that the times required for convergence are significantly different.

The practical results allow the following considerations:

- a) All results were very oscillatory due to imprecision in measurements and sample time;
- b) Better results were obtained with tests that use a larger data set, that is, SRM and LSM(1);
- c) The tests with 100 samples (LSM(2), LSM(3), and MALSM) showed a considerable variation in each calculated value;
- d) Tests with WLSM present considerable difficulty in convergence. This method is not indicated when there is an imprecision or noise in the measurements, which can cause considerable oscillations in the identification of real systems. This explains the reason why some WLSM algorithms did not

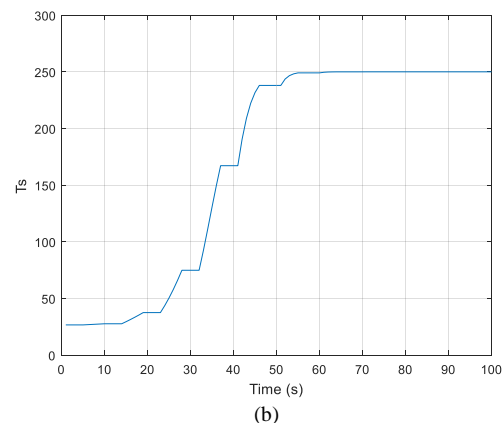
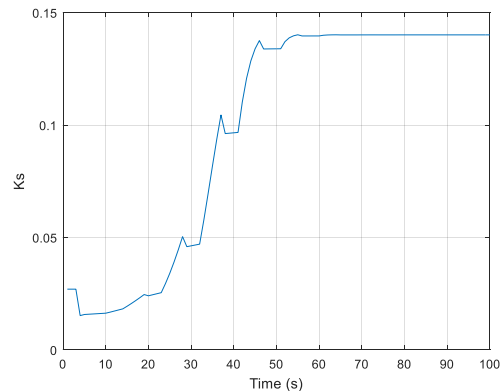


Fig. 9. Step Response of First Order System Parameters Identification – Simulation of WLSM(2) a) K_s Parameter; b) T_s Parameter.

converge in the practical test and even the WLSM(5) presented considerable oscillations as shown in Fig. 11. For comparative purposes, the values presented in Table 2 consider the last 2000 samples of the test performed, where visually the convergence of the parameters was considered, but with relative oscillation, as shown in the tests in Fig. 8 and Fig. 11.

e) Although the identification produces significant variations in K_S and T_S parameters, it is observed that the obtained parameters) $W(k)$ obtained by

$$W(k) = \begin{bmatrix} e^{-\frac{1}{T_S}T} \\ (1 - e^{-\frac{1}{T_S}T})K_S \end{bmatrix}$$

present much less significant variations, due to the exponential component of the equation.

VI. CONCLUSION

The interfaces of accessing the remote equipment proved to be quite didactic and allowed the performance of activities mediated by technologies in association with teaching.

The WebLabs were used as a teaching tool in the discipline "Process Control" of the Electronic Engineering course and in the Post-Graduate course in "Engineering of Industrial Process Control", as well as in complementary activities. Typically, the use of didactic activities was carried out using virtual machines.

In any case, the differential is the possibility of using the didactic resources remotely, which provides versatility for the student, having the differential more flexibility and autonomy, since the student can control the step by step of carrying out the experiment allowing greater learning possibilities.

Detailed research point shows that the methods of systems identification could be evaluated under different contexts with adjustment of the initial and operational conditions, such as changing amplitudes, changing the sampling time, and modifying the algorithm's parameters, among others.

The results presented in Table 1 and Table 2 show a quantitative and qualitative comparison of the performance produced by each identification method, allowing the assessment of the pros and cons of each methodology. In the simulations, the MALSM and WLSM methods (for certain initial conditions) presented very satisfactory performance, being advantageous over the other methods for allowing the parameters identification in each sampling time. Additionally, they have the potential to allow real-time identification by the detection of parameter variation, with the system in operation. However, the practical system used presents considerable imprecision in the measurements performed at each sampling time. Besides, the data acquisition system implemented allows a small variation in the sampling time, which, although not very significant, can compromise the stability and convergence in parameter identification. Additionally, it should be noted that a small change in the sampling time or the discrete parameters can produce considerable changes in the continuous parameters of the system.

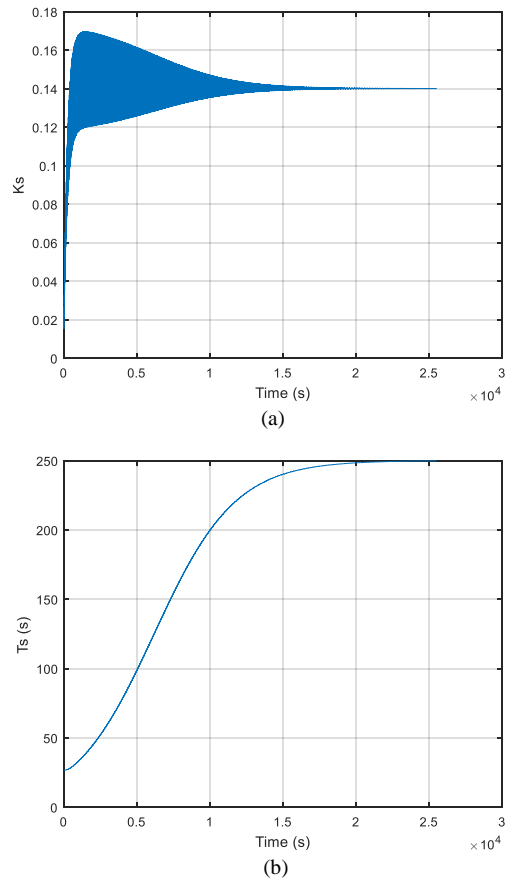


Fig. 10. Parameters Identification – Simulation of WLSM(5) a) K_S Parameter; (b) T_S Parameter.

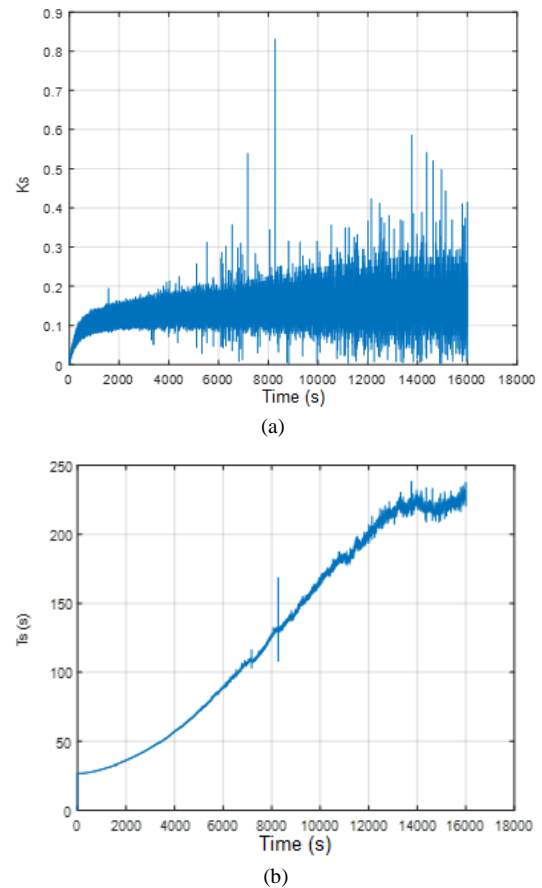


Fig. 11. Parameters Identification – Practical Application of WLSM(5) a) K_S Parameter; (b) T_S Parameter.

REFERENCES

- [1] Khaddage, F., Fayad, R., Moussallem, I. "Online Learning and The Role of Technologies During COVID-19 Pandemic Higher Education Lebanon Case", in *Proceedings of EdMedia + Innovate Learning* (pp. 623-630). Online, The Netherlands: Association for the Advancement of Computing in Education (AACE), Waynesville, NC, USA, 2020.
- [2] Pahrudin, P., Liu, LW,m Chang, CY "The Influencing Factor of ICT Use in Online Learning during Covid-19 Pandemic in Indonesia", *Engineering Letters*, vol. 9, no. 2, pp 382-390, 2021.
- [3] Jesus, C. D. F., Giordano, R. C., Cruz, A. J. G., Alliet-Gaubert, M., Joulia, X., Rouzineau, D., Albert, J.; Coufort, C.; Roux; G. C. L. "WebLab in Chemical Engineering Between France and Brazil: Validation of Methodology", in *ICEE 2007 – International Conference on Engineering Education*, Coimbra, Portugal. pp 3-7, 2007.
- [4] Marques, A., Viegas, C., Costa-Lobo, C., Fidalgo, G. A., Rocha, J., Gustavsson, I. "How Remote Labs Impact on Course Outcomes: Various Practices Using VISIR". *IEEE Transactions on Education*, vol. 57, no. 3, pp 151-159, 2014.
- [5] Fidalgo, V.; Alves, G. R.; Marques, M.A.; Viegas, M.C.; Costa-Lobo, M.C.; Hernandez-Javo, U.; Garcia-Zubia, J.; Gustavsson, I. "Adapting Remote Labs to Learning Scenarios: Case Studies Using VISIR and RemotElectLab". *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, vol. 9, no. 1, pp 33-39, 2014.
- [6] Garcia-Zubia, J., Lopes-De-Ipina, D., Orduña, P.; Hernandez-Javo, U. "Experience with WebLab-Deusto". in *IEEE International Symposium on Industrial Electronics*, Montreal, Que., Canada, 2006.
- [7] Rodrigues-Gil, L., Orduña, P., Garcia-Zubia, J., Lopes-de-Ipiña, D. "Advanced Integration of OpenLabs VISIR (Virtual InstrumentSystems in Reality) with Weblab-Deusto". in: *9th International Conference on Remote Engineering and Virtual Instrumentation*, Bilbao, Spain, 2012.
- [8] Landin, P. N., Gustafsson, S., Fager, C., Eriksson, T. "WebLab: A Web-Based Setup for PA Digital Predistortion and Characterization". *IEEE Microwave Magazine*, 2015, vol. 16, no. 1, pp 138-140, 2015 <https://doi.org/10.1109/MMM.2014.2367857>
- [9] Cruz, A. J. G., Jesus, C. D. F., Giordano, R. C., Nascimento, C. A. O., Roux, G. A. C. L., Loureiro, L. V. "Experimento de Controle de Nível Operado Remotamente Via Internet: Projeto WebLab". in: *XXXIII Congresso Brasileiro de Ensino de Engenharia*, Campina Grande, PB, Brasil, vol. 1, pp 1-10, 2005.
- [10] Agrawal, A. and Srivastava, S. "WebLab: A Generic Architecture for Remote Laboratories". in the *15th International Conference on Advanced Computing and Communications (ADCOM 2007)*, pp 301–306, 2007.
- [11] Mohammed, A. K., El Zoghby, H. M., Elmesalawy, M. M. "Remote Controlled Laboratory Experiments for Engineering Education in the Post-COVID-19 Era: Concept and Example". in: *2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, pp 629-634, 2020.
- [12] Assis, W. O., Coelho, A. D. "Uso de Tecnologias de Acesso Remoto e WebLabs como Estratégia de Ensino-Aprendizagem no Contexto da Pandemia do Corona Vírus". in: *COBENGE 2020 - XLVIII Congresso Brasileiro de Educação em Engenharia e III Simpósio Internacional de Educação em Engenharia da ABENGE*, Brasil, 2020.
- [13] Assis, W. O. Coelho, A. D., Gonçalves, H. S. B. "WebLabs: Remote Access Experiments for Teaching Process Control in Engineering Courses". in: *XIV Congreso de Tecnología, Aprendizaje y Enseñanza de la Electrónica (XIV Technologies Applied to Electronics Teaching Conference) (TAEE)*, Porto, Portugal, pp 1-7, 2020.
- [14] Assis, W. O., Coelho, A. D.; Gedraite, R.; Matta, E. N.; Kunigk, L. "An Educational Tool with Remote Access for Teaching Modelling Control Systems for Engineering Courses". In: *Conference: Modelling, Simulation and Identification / 841: Intelligent Systems and Control*, Campinas, Brazil, 2016.
- [15] Assis, W. O.; Coelho, A. D.; Júnior, J. C. S.; Kunigk, L.; Morais, A. S.; Lemos, G. M.; Gedraite, E. S.; Takahashi, E. K.; Gedraite, R. "WebLab for Control Applications in Engineering Education". in: *Proceedings of The 10th IASTED International Conference on Web-based Education*, Innsbruck, Áustria, 2013.
- [16] Sarrafan, K., Muttaqi, K. M., Sutanto, D. "Real-Time Estimation of Model Parameters and State-of-Charge of Li-Ion Batteries in Electric Vehicles Using a New Mixed Estimation Model". *IEEE Trans. Ind. Appl.*, vol. 56, no. 5, pp 5417–5428, 2020.
- [17] Amifia, L. K. "Model Parameter Identification of State of Charge Based on Three Battery Modelling using Kalman Filter". *Engineering Letters*, 30 (3), pp. 1128-1137, 2022.
- [18] Giraldo, E. "Adaptive Indirect Field Oriented Linear Control of an Induction Motor". *IAENG International Journal of Applied Mathematics*, vol. 52, no. 1, pp 18-23, 2022.
- [19] Osorio-Arteaga, F., Marulanda-Durango, J. J, Giraldo, E. "Robust Multivariable Adaptive Control of Time-varying Systems". *IAENG International Journal of Computer Science*, vol. 47, no. 4, pp. 605-612, 2020.
- [20] Assis, W. O., Gonçalves, H. S. B., Silva, G. T., Coelho, A. D. "Uma Aplicação de Controle de Nível com Acesso Remoto pela Web". in: *COBENGE 2019 - XLVII Congresso Brasileiro de Educação em Engenharia*, Fortaleza, CE, Brasil, 2019.
- [21] Assis, W. O., Coelho, A. D., Pacheco, V. G., Gedraite, R., Pires, R. C. "Applying the Concept of WebLab in a Playful Way - The Electronic Engineering Hands On". in: *ICSE 2014 – 2nd International Congress of Science Education*, 2014, Foz do Iguaçu, PR, Brazil, 2014.
- [22] Matta, E. N., Assis, W. O., Gomes, M. M., Barbosa, V. P., Gedraite, R., Ataíde, C. H. "Contribuição para o Melhor Aproveitamento de Fluido de Perfuração de Poços de Petróleo – Uma Abordagem Utilizando Processamento de Imagem". *Brazilian Journal of Development*, Curitiba, vol. 5, no. 4, pp 3645-3659, 2019. <https://doi.org/10.34117/bjdv5n5-1528>
- [23] Ogata, K. *Modern Control Engineering*. Prentice Hall, 2010.
- [24] Aguirre, L. *Introdução à Identificação de Sistemas*, Belo Horizonte - MG, Brazil, Editora UFMG, 2004.
- [25] Verhaegen, M., Verdult, V. *Filtering and System Identification – A Least Squares Approach*, Cambridge, Cambridge University Press, 2007.
- [26] Iserman, R. *Digital Control Systems*. Berlin: Springer-Verlag, 2012.