SRTTU

JCARME
ISSN: 2228-7922
Journal of Computational and Applied Research in Mechanical Engineering
Vol. 10, No. 1, pp. 111-123
jcarme.sru.ac.ir

LabVIEW implementation of an enhanced nonlinear PID controller based on harmony search for one-stage servomechanism system

Mohamed A. Shamseldin[a,*], Mohamed Sallam[b], A. M. Bassiuny[b] and A. M. Abdel Ghany[c]

[a]*Mechanical Engineering Department, Future University in Egypt, Cairo, Egypt*
[b]*Mechanical Engineering Department, Helwan University, Cairo, Egypt*
[c]*Electrical Engineering Department, October 6 University (On Leave from Helwan University), Cairo, Egypt*

Article info:

Type;	Research
Received:	02/05/2018
Revised:	01/04/2019
Accepted:	14/04/2019
Online:	14/04/2019

Keywords:
Nonlinear PID,
Harmony search,
Servomechanism,
System identification.

Abstract

This paper presents a practical implementation for a new formula of nonlinear PID (NPID) control. The purpose of the controller is to accurately trace a preselected position reference of one stage servomechanism system. The possibility of developing a transfer function model for the experimental setup is elusive because of the lack of system data. So, the identified model is developed via gathering experimental input/output data. The performance of the enhanced nonlinear NPID controller had been investigated by comparing it with a linear PID controller. The harmony search tuning system is built to determine the optimum parameters for each control technique based on an effective objective function. The experimental outcomes and the simulation results show that the proposed NPID controller has a minimum rise time and settling time through constant position reference test. Also, the NPID control is faster than the linear PID control by 40% in the case of the variable position reference test.

1. Introduction

The latest progress of machine tools is to develop high-speed spindle and feed drives, which lead to high performance and reduce the machining cycle times [1]. Moreover, the development of feed drives with adequate dynamic response and good performance became essential in many industrial applications [2]. The purpose of servo control systems is to maintain the stage follows a preselected position profile along complicated trajectories at high feed speeds [3]. The machine tool with traditional feed drives uses the proportional position control which leads to high fluctuation in the stage and large tracking errors at high speeds [4]. The tracking error is eliminated using high-performance feed drive motors with advanced control techniques [5]. However, friction on lead screws and guides, cutting force disturbance, and changes in the work-piece mass in linear drives are obstacles to achieve good contouring accuracy at high feeds [6]. The requirements for high speed and accurate contouring have led to the investigation of efficient control algorithms in recent years [7].
Several common industrial applications have utilized the traditional Proportional-Integral-Derivative (PID) controllers for the control of processes [8]. The PID controller algorithm is simple and has acceptable performance for most of the common systems, which makes it used for several decades. The behavior of the PID controller is highly sensitive by determining its

[*]*Corresponding author*
email address: eng.moh.abd88@gmail.com

parameters [5]. Until now, there is no a definite ideal method to select the proper PID controller parameters for a certain system [9]. In the last period, the researchers avoid this problem using the optimization techniques, for instance, Genetic Algorithm (GA), Ant Colony Optimization Particle Swarm Optimization (PSO), and Harmony Search (HS) optimization to find the optimum parameters of PID controller away from the rough methods (Ziegler-Nichols method) which cannot guarantee the proper selection for those parameters [10]. The optimization techniques cannot be carried out practically in real-time, where it takes a long time to find the optimum solution. So, most of the research resort to simulating the actual system using an accurate mathematical model for building the tuning system [11].

Most of the real systems are nonlinear systems, but the nonlinearity percentage tolerances from system to another [12]. It is known that mechanical systems have complex nonlinear behavior because of friction and backlash problems [13], like the one-stage servomechanism system. So, the traditional PID controller with linear parameters cannot achieve high performance for this type of system [14]. Also, the PID controller still has fixed gains, which are not enough to deal with high complicated dynamic systems [15].

Recent research has proceeded to design nonlinear PID control to overcome the nonlinearity and uncertainty of the system. The nonlinear PID (NPID) controller contains nonlinear gains incorporated with the fixed gains of the PID controller [13]. These nonlinear gains enjoy the advantage of high initial gain to attain a fast dynamic response, followed by a low gain to prevent an oscillatory behavior [14]. The present studies propose a nonlinear gain (one scalar gain) which is multiplied with the output of linear PID control [16]. So, this paper presents a new formula for NPID control where the nonlinear gain is a vector gain that contains three values special to the proportional, integral, and derivative gain. Also, most of the current studies don't present an obvious method to obtain the NPID control parameters [15]. Moreover, previous research is limited to simulations without practical implementation. This study presents a practical methodology to design a new form of nonlinear PID control. It is known that MATLAB Simulink is a powerful tool to simulate and design most of the control systems but it is not suitable and reliable with the real-time implementation of control systems [17]. So, this study uses MATLAB Simulink only for purposes of control design and the obtained results from the simulation is executed practically by LABVIEW software using NI 6009 DAQ card. Several parameters of the experimental setup aren't known, so this research resorts for using the system identification techniques (linear and nonlinear least-squares methods) to develop an identified model that is used in control design. The optimum parameters of the proposed controller are obtained offline using HS optimization technique based on a certain cost function.

2. System modeling and identification

2.1. System modeling

This section presents the differential equations to describe the dynamic behavior of a one-stage servomechanism system. The structure of the one-stage servomechanism system is shown in Fig. 1. The stage is loaded on linear guides and the screw shaft. The rotor of the DC motor is coupled directly with a screw shaft. The DC motor rotates the screw shaft while the nut converts rotary motion to translational motion and drives the stage. The speed and position of the stage are determined by a rotary encoder coupled with a screw shaft. The DC motor model can describe from Eq. (1) to Eq. (3) where v_a is the armature voltage, R_a is the armature resistance, I_a is armature current, L_a is the inductance of motor windings, K_b is the back emf constant, ω is the rotor speed, T_m is the motor torque, K_t is the torque constant, J_e is the total equivalent moment of inertia as seen by the motor, B is the viscous damping coefficient, and T_s is the total static torque reflected in the lead screw shaft.

$$v_a = R_a I_a + L_a \frac{di_a}{dt} + K_b \omega \quad (1)$$

$$T_m = K_t \, i_a \quad (2)$$

$$T_m = J_e \dot{\omega} + B\omega + T_s \quad (3)$$

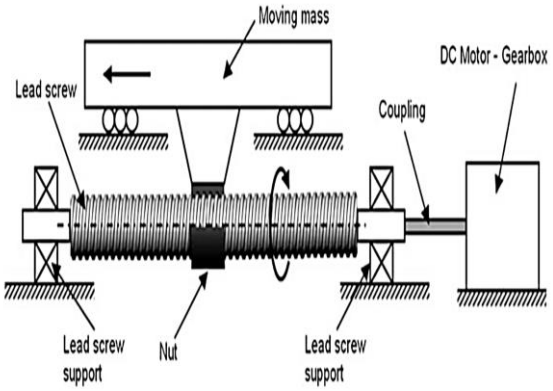

Fig. 1. The structure of lead screw drive system [18].

There are three sources for the total static torque reflected in the lead screw shaft (T_s), as in Eq. (4), where T_{gf} is the static torque contributed by the friction in the guideways, T_{bf} is the torque lost in bearings due to friction, and T_f is the torque required to overcome the feed forces.

$$T_s = T_{gf} + T_{bf} + T_f \tag{4}$$

In the case of gear between the motor shaft and the lead screw shaft, the T_s is determined as follows where r_g is the gear reduction.

$$T_{sr} = \frac{T_s}{r_g} \tag{5}$$

Eq. (6) illustrates the static torque due to guideways friction where h_p is the pitch of lead screw, μ_{gf} is the friction coefficient in guides, m_t is the table mass, m_w is the maximum mass for the workpiece, and F_z is the maximum vertical force.

$$T_{gf} = \frac{h_p}{2\pi}\,\mu_{gf}(\,(m_t + m_w)g + F_z) \tag{6}$$

Eq. (7) demonstrates the lost static torque due to bearing friction where μ_b is the friction coefficient of bearings, d_p is the leadscrew diameter, F_f is the maximum feeding force, and F_p is the preload force in the thrust bearings.

$$T_{bf} = \frac{d_p}{2}\,\mu_b(F_f + F_p) \tag{7}$$

The required torque to overcome feed forces can be calculated using Eq. (8).

$$T_f = \frac{h_p}{2\pi}F_f \tag{8}$$

The total equivalent moment of inertia, as seen by the motor (J_e), in Eq. (9), depends on J_{tw}, the moment of inertia of table and workpiece reflected on the lead screw shaft, where J_L is the lead screw inertia and J_m is the motor shaft inertia.

$$J_e = \frac{J_{tw}+J_L}{r_g^2} + J_m \tag{9}$$

The moment of inertia of table and workpiece can be determined based on the mass table (m_t) and maximum mass of workpiece (m_w) as given in Eq. (10).

$$J_{tw} = (m_t + m_w)(\frac{h_p}{2\pi})^2 \tag{10}$$

The lead screw moment inertia is determined using lead screw mass as Eq. (11).

$$J_L = \frac{1}{2}\,m_L(\frac{d_p}{2})^2 \tag{11}$$

Eq. (1) is reorganized to obtain Eq. (12):

$$i_a = \frac{v_a - K_b\omega}{L_a s + R_a} \tag{12}$$

Also, Eq. (3) is reformed to result Eq. (13)

$$\omega = \frac{T_m - T_s}{J_e s + R_a} \tag{13}$$

Eqs. (12) and (13) are used to build the DC motor model incorporated with the servomechanism system as shown in Fig. 2.

$$\omega(s) = G_1(s)v_a(s) + \,G_2(s)\,T_s(s) \tag{14}$$

To find the transfer function between ω and v_a, it is assumed $T_s = 0$ as demonstrated in Fig. 3, and the following transfer function is obtained:

$$\frac{\omega(s)}{v_a(s)} = \frac{K_t}{L_a J_e s^2 + J_e R_a s + K_t K_b} \tag{15}$$

To obtain the transfer function between ω and T_s assumes $v_a = 0$ as shown in Fig. 4 will result the following transfer function.

$$\frac{\omega(s)}{T_s(s)} = \frac{-(L_a s + R_a)}{L_a J_e s^2 + (L_a B + J_e R_a)s + (R_a B - K_t K_b)} \quad (16)$$

The actual position table can be calculated using Eq. (17).

$$X_{act}(s) = \frac{h_p}{2\pi}\frac{1}{s}\omega(s) \quad (17)$$

2.2. System identification and experimental setup

The lack of model data parameters are the problem to obtain a mathematical model for a certain system [19]. So, the purpose of system identification fabricates an approximate model system using experimental input/output data.

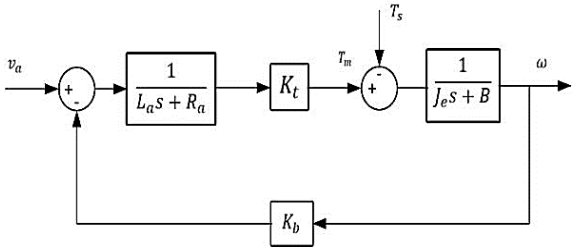

Fig. 2. Block diagram of DC motor model incorporated with servomechanism system.

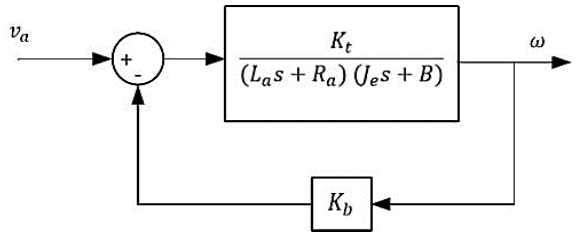

Fig. 3. Block diagram of DC motor model incorporated with servomechanism system without load.

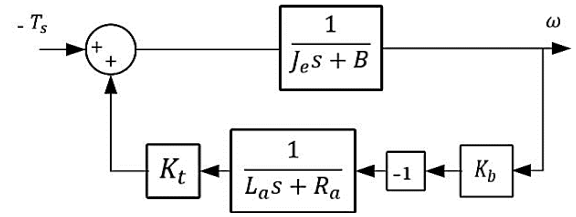

Fig. 4. Block diagram of DC motor model incorporated with servomechanism system at $v_a = 0$.

The method to develop a model involves three basic steps. The first step is the input and output data to be collected from the experiment. In the second step, many candidate models are developed. The third step is selecting an appropriate model from the set of candidate models.

The general linear transfer function of such a system may be written as follows:

$$\frac{w(s)}{v_a(s)} = \frac{k}{b_n . S^n + b_{n-1} . S^{n-1} + \cdots + b_0} \quad (18)$$

where $k, b_n, \ldots, b_0$ are the estimated parameters of the approximate transfer function. It is known that the nonlinear system cannot be represented exactly by linear models. The accuracy of the model can be increased by increasing the order of the linear system. However, there is often a limitation that increasing order cannot improve the model accuracy sufficiently. Therefore, it is necessary to explicitly add the nonlinearities into the system.

In this paper, it is tried to use the nonlinear ARX model structure to model such systems where AR refers to the autoregressive part and X to the extra input. A nonlinear ARX model can be understood as an extension of a linear model as shown in Fig. 5.

Assume nonlinear function has two unknown parameters b_0 *and* b_1.

$$f(\mathrm{x}) = b_0\left(1 - e^{-b_1 x}\right) + e \quad (19)$$

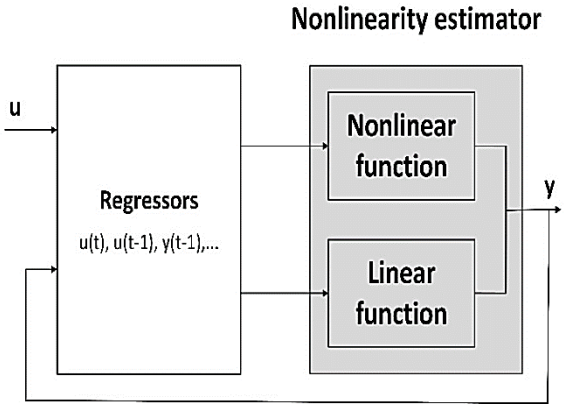

Fig. 5. The structure of a nonlinear ARX model.

The relationship between the nonlinear equation and the data can be described as follows:

$$y_i = f(x_i; b_0, b_1, \dots\dots, b_m) + e_i \quad (20)$$

where y_i is a measured output value, $f(x_i; b_0, b_1, \dots\dots, b_m)$ is a function of the independent variable x_i and the parameters $b_0, b_1, \dots\dots, b_m$, and e_i is a random error. This model can be rewritten in a short form by ignoring the parameters:

$$y_i = f(x_i) + e_i \quad (21)$$

The nonlinear equation can be analyzed in a Taylor series around the parameter values,

$$f(x_i)_{j+1} = f(x_i)_j + \frac{\partial f(x_i)_j}{\partial b_0}\Delta b_0 + \frac{\partial f(x_i)_j}{\partial b_1}\Delta b_1 \quad (22)$$

where j stands for the initial values, and j + 1 is the prediction.

$$\Delta b_0 = b_{0,j+1} - b_{0,j} \text{ and } \Delta b_1 = b_{1,j+1} - b_{1,j} \quad (23)$$

Assume the initial values of $b_{0,j}$ and $b_{1,j}$, and substituting Eq. (22) in Eq. (21) results in:

$$y_i - f(y_i)_j = \frac{\partial f(y_i)_j}{\partial a_0}\Delta b_0 + \frac{\partial f(y_i)_j}{\partial a_1}\Delta b_1 + e_i \quad (24)$$

or in matrix form of:

$$\{H\} = [Q_j]\{\Delta B\} + \{E\} \quad (25)$$

where $[Q_j]$ is the matrix of partial derivatives of the function evaluated at the initial values j.

$$[Q_j] = \begin{bmatrix} \partial f_1/\partial b_0 & \partial f_1/\partial b_1 \\ \partial f_2/\partial b_0 & \partial f_2/\partial b_1 \\ \dots\dots & \dots\dots\dots \\ \partial f_n/\partial b_0 & \partial f_n/\partial b_1 \end{bmatrix} \quad (26)$$

where n is the number of data points.
The vector {H} consists of the differences between the speed measurements and the function values.

$$\{H\} = \begin{Bmatrix} y_1 - f(x_1) \\ y_2 - f(x_2) \\ \dots\dots\dots \\ y_n - f(x_n) \end{Bmatrix} \quad (27)$$

The vector $\{\Delta B\}$ consists of the changes in the parameter values,

$$\{\Delta B\} = \begin{Bmatrix} \Delta b_0 \\ \Delta b_1 \\ \dots\dots \\ \Delta b_m \end{Bmatrix} \quad (28)$$

Subjecting linear least-squares theory to Eq. (25),

$$\{[Q_j]^T\{H\}\} = [Q_j]^T[Q_j]\{\Delta B\} \quad (29)$$

$$\{\Delta B\} = \left[[Q_j]^T[Q_j]\right]^{-1}\{[Q_j]^T\{H\}\} \quad (30)$$

For {ΔB}, which can be applied to determine upgraded values for the parameters, as in:

$$b_{0,j+1} = b_{0,j} + \Delta b_0$$
$$b_{1,j+1} = b_{1,j} + \Delta b_1 \quad (31)$$

where $b_{0,j+1}$ and $b_{1,j+1}$ are the new values of b_0 and b_1.
These steps are repeated until the estimated parameters have a satisfied error between the actual output and model output. The main components of the one-stage table servomechanism experimental setup, as illustrated in Fig. 6, are as follows:

1. **One-stage table:** It consists of a DC motor driving a lead screw on which a sliding block is installed. The DC motor has a nominal speed of 1800 rev/min, and an armature voltage of 90 V.
2. **Optical encoder:** An encoder is an electrical mechanical device that can monitor motion or position. The optical encoder provides position feedback signals (100 pulses per revolution).
3. **Limit switches:** Two magnetic limit switches detect the sliding block when reaches the start or end position.

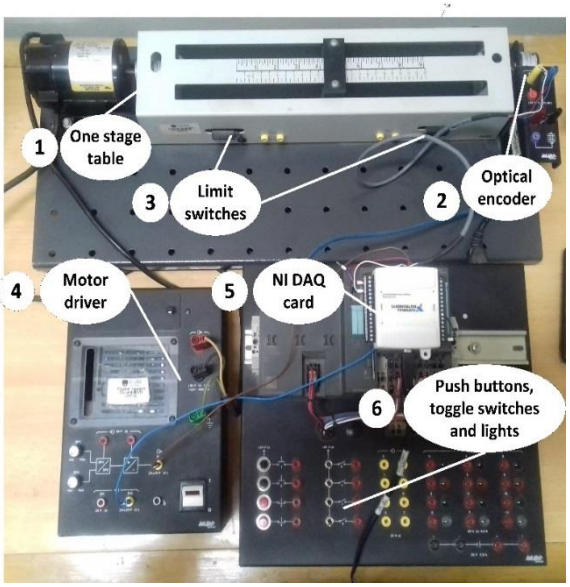

Fig. 6. One-stage table servomechanism actual setup.

4. **Motor driver:** The DC motor drive controls the DC motor electro-mechanical module, Model 3293.
5. **A data acquisition card (DAQ) NI USB-6009:** It has the following specifications:
   - 8 analog inputs (12-bit, 10 kS/s).
   - 2 analog outputs (12-bit, 150 S/s).
   - 12 digital I/O.
   - USB connection, no extra power supply needed.
6. **Push buttons, toggle switches, and lights:** They are used to operate the DC motor driver manually.
7. **Computer**: It is used to perform the control algorithms and receive and send the signals from the NI DAQ card.

3. Enhanced nonlinear PID nontrol

It is well known that the transfer function of the linear PID controller is $K(s) = K_P + \frac{K_i}{s} + K_d s$. In the above equation, K_p, K_i, and K_d are fixed gains. These gains can be defined as follows. K_p is the proportional gain which attempts to reduce the error responses. K_i is the integral gain and its role dampen the steady-state error. k_d is the differential gain which decreases the overshoot of the system. It also ensures the system stability [20, 21].

In spite of linear fixed parameters, PID controllers are often suitable for controlling a simple physical process. The demands for high performance control with different operating point conditions or environmental parameters are often beyond the abilities of simple PID controllers [22, 23]. The performance of linear PID controllers can be enhanced using several techniques, which will be developed to deal with sudden disturbances and complex systems, for example, the PID self-tuning methods, neural networks, and fuzzy logic strategies, and other methods [24, 25].

Among these techniques, the NPID control is presented as one of the most appropriate and effective methods for industrial applications. The NPID control is carried out in two broad categories of applications. The first category is particular to nonlinear systems, where the NPID control is used to absorb the nonlinearity. The second category deals with linear systems, where the NPID control is used to obtain enhanced performance which is not realizable by a linear PID control, such as reduced overshoot, diminished rise time for a step or rapid command input, obtained better tracking accuracy, and used to compensate the nonlinearity and disturbances in the system [26, 27]. The NPID controllers have the advantage of high initial gain to achieve fast dynamic response, followed by a low gain to avoid unstable behavior. In this study, the traditional linear PID controller can be enhanced by combining a sector-bounded nonlinear gain into a linear fixed gain PID control architecture.

The proposed enhanced NPID controller consists of two parts. The first part is a sector bounded nonlinear gain $K_n(e)$ while the second part is a linear fixed-gain PID controller (K_p, K_i, and K_d). The nonlinear gain $K_n(e)$ is a sector-bounded function of the error e(t). The previous researches have been considered the nonlinear gain $K_n(e)$ as a one scalar value.

The new in this research, the one scalar value of $K_n(e)$ is replaced with a row vector and can be expressed as: $K_n(e) = K_{n1}(e)\, K_{n2}(e)\, K_{n3}(e)]$ as shown in Fig. 7, which leads to improve the performance of NPID controller where the values of nonlinear gains are adjusted according to the error and the type of fixed parameters (K_p, K_i, and K_d).

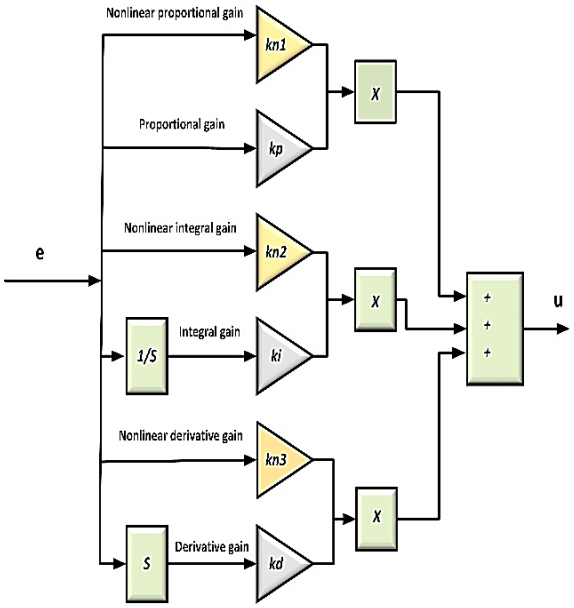

Fig. 7. The enhanced nonlinear PID controller structure.

The proposed form of NPID control can be described as follows:

$$u(t) = K_p[K_{n1}(e).e(t)] + K_i \int_0^t [K_{n2}(e).e(t)]\,dt + K_d\left[K_{n3}(e).\frac{de(t)}{dt}\right] \quad (30)$$

where $K_{n1}(e), K_{n2}(e), and\ K_{n3}(e)$ are nonlinear gains. The nonlinear gains represent any general nonlinear function of the error which is bounded in the sector $0 < K_n(e) < K_n(e)$ max. There is a wide range of choices available for the nonlinear gain $K_n(e)$.
One simple form of the nonlinear gain function can be described as:

$$K_{ni}(e) = ch(w_i e) = \frac{\exp(w_i e)+\exp(-w_i e)}{2} \quad (31)$$

where $i = 1, 2, 3$.

$$e = \begin{cases} e & |e| \le e_{max} \\ e_{max} sgn(e) & |e| > e_{max} \end{cases} \quad (32)$$

The nonlinear gain $K_n(e)$ is lower bounded by $K_n(e)$min = 1 when e = 0, and upper-bounded by $K_n(e)$max = ch($w_i\ e_{max}$). Therefore, e_{max} stands for the range of deviation, and w_i describes the rate of variation of $K_n(e)$.

The one-stage servomechanism system needs two cascaded controllers; the first controller is the speed controller while the second one is the position controller. Usually, in the position controller, the integral part is eliminated to be the output of the position controller reference to the speed controller.
The critical point in the PID and NPID controllers is to select the proper parameters to be appropriate for the controlled plant. There are different approaches to find the parameters of the PID controller, for instance, try and error and Ziegler-Nichols method, but most of these approaches are rough roads. In this paper, the HS optimization technique is used to obtain the optimal values of both PID and NPID controller parameters according to the objective function as shown in Eq. (33) [28].

$$f = \frac{1}{(1-e^{-\beta})(M_p+e_{ss})+e^{-\beta}(t_s-t_r)} \quad (33)$$

where e_{ss} is the steady-state error, M_p is the overshoot of system response, t_s is the settling time, and t_r is the rise time. Also, this objective function is able to compromise the designer requirements using the weighting parameter value (β). The parameter is set larger than 0.7 to reduce overshoot and steady-state error. If this parameter is adjusted to smaller than 0.7, the rise time and settling time is reduced. HS was suggested by Zong Woo Geem in 2001 [29]. It is well known that HS is a phenomenon-mimicking algorithm inspired by the improvisation process of musicians [30]. The initial population of Harmony Memory (HM) is chosen randomly. HM consists of Harmony Memory Solution (HMS) vectors. The HM is filled with HMS vectors as given in Eq, (34).
Fig. 8 illustrates the block diagram for the overall HS tuning system using the enhanced nonlinear PID controller. Table 1 demonstrates the values of the obtained parameters using the HS tuning system.

4. Experimental and simulation results

4.1. Identified model validation

This section demonstrates the practical steps to develop the identified model for a one-stage servomechanism system prototype.

$$HM = \begin{bmatrix} K_{p\,(1,1)} & K_{i\,(1,2)} & K_{d\,(1,3)} & w_{1\,(1,4)} & w_{2\,(1,5)} & w_{3\,(1,6)} & K_{p\,(1,7)} & K_{d\,(1,8)} & w_{1\,(1,9)} & w_{3\,(1,10)} \\ K_{p\,(2,1)} & K_{i\,(22)} & K_{d\,(2,3)} & w_{1\,(2,4)} & w_{2\,(2,5)} & w_{3\,(2,6)} & K_{p\,(2,7)} & K_{d\,(2,8)} & w_{1\,(2,9)} & w_{3\,(2,10)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ K_{p\,(HMS,1)} & K_{i\,(HMS,2)} & K_{d\,(HMS,3)} & w_{1\,(HMS,4)} & w_{2\,(HMS,5)} & w_{3\,(HMS,6)} & K_{p\,(HMS,7)} & K_{d\,(HMS,8)} & w_{1\,(HMS,9)} & w_{3\,(HMS,10)} \end{bmatrix} \quad (34)$$

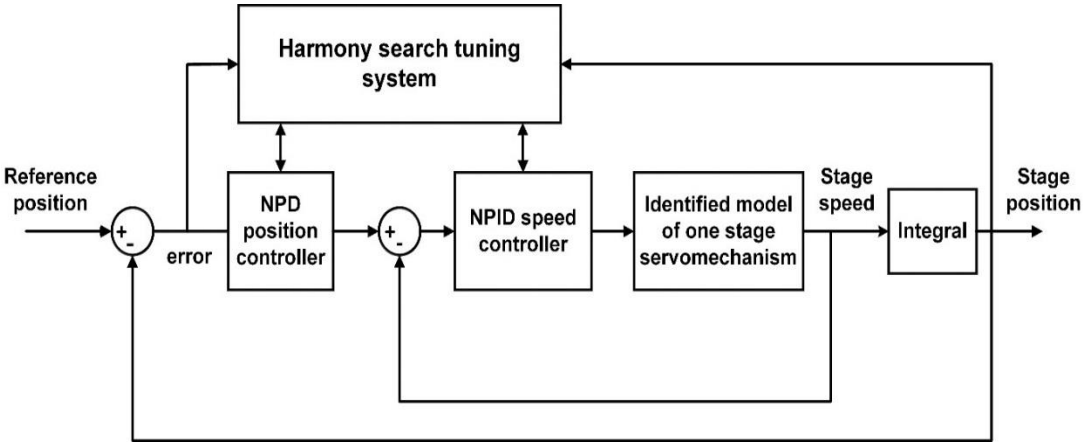

Fig. 8. Closed loop system with HS tuning system.

Table 1. The obtained parameters of each control technique.

Controller type	Position controller (without integral part)				Speed controller					
NPID controller	K_p	w_1	K_d	w_3	K_p	w_1	K_i	w_2	K_d	w_3
	10.45	0.67	0.0045	0.93	50.56	0.96	0.0067	0.034	5.054	0.0456
PID controller	K_p		K_d		K_p		K_i		K_d	
	26.1677		1.7924		47.4250		23.0674		0.7043	

Also, Fig. 9 shows the connection of the components between each part through the experiment. The candidates identified models are investigated by comparing the output of each identified model with actual experimental data. In the beginning, the experimental input/output data are collected. The NI DAQ card generates random signal ranges from -5V to +5V with a sample rate of 50 milliseconds as illustrated in Fig. 10 where this signal is the input for the DC motor drive.

The speed of DC motor will fluctuate with change the generated signal. The positive ranges of voltage signal make the DC motor speed fluctuate in the forward direction while the DC motor speed fluctuates in reverse direction through the negative ranges of voltage. This continuous change in input signal makes the speed of the one-stage table fluctuate proportionally, as illustrated in Fig. 11.

Fig. 12 demonstrates the actual position of the one-stage table through the experiment. It can be noted that the table position increases in positive ranges of the input signal to the DC motor driver while the position decreases in negative ranges of the input signal. The data are collected and stored in an excel sheet file, and then the data are used to fabricate the identified model for the one-stage servomechanism system.

Fig. 13 demonstrates a comparison between the actual stage speeds of the servomechanism system with the output of two candidates identified models. The first identified model presents a second-order system while the second one includes a nonlinear ARX model. It is obvious that the nonlinear ARX identified can simulate the behavior of the actual experimental setup compared to the second-order identified model.

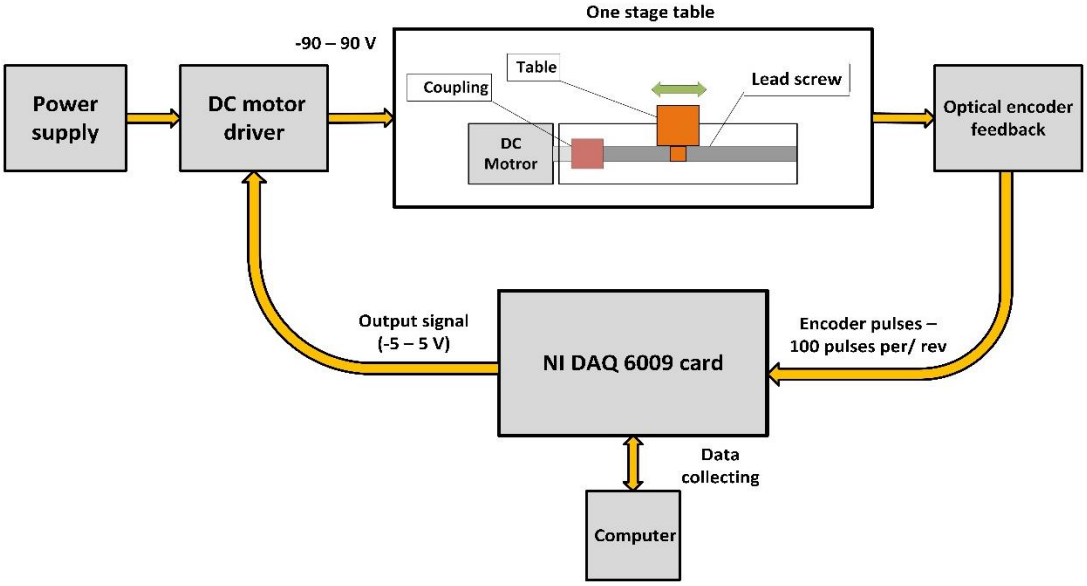

Fig. 9. Block diagram of experimental setup servomechanism system.

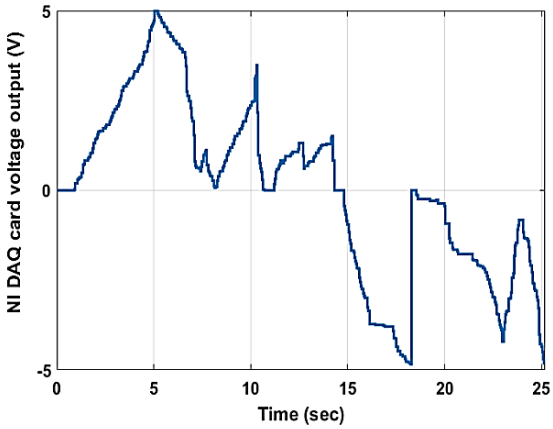

Fig. 10. The random input signal to DC motor driver.

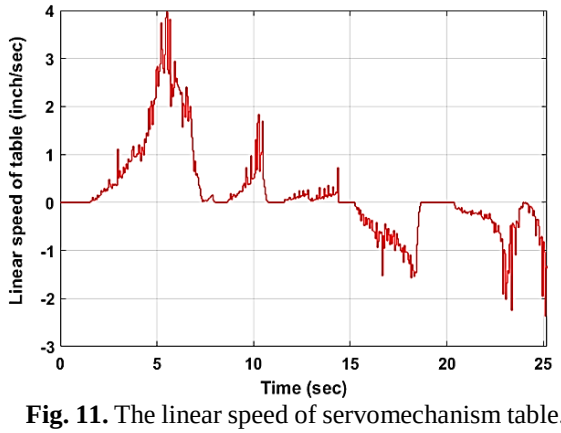

Fig. 11. The linear speed of servomechanism table.

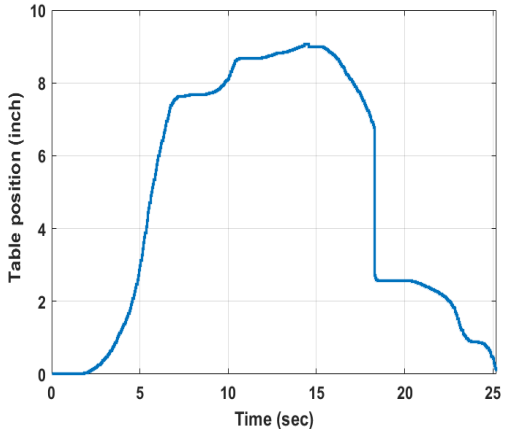

Fig. 12. The position of servomechanism table.

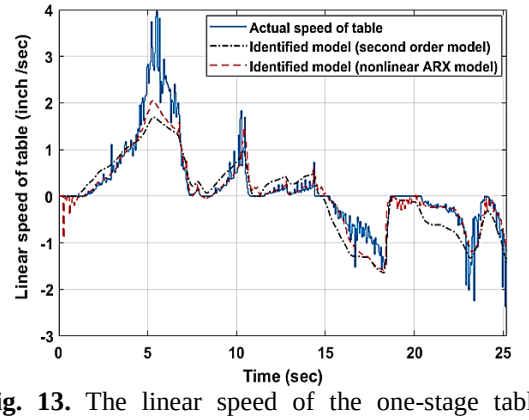

Fig. 13. The linear speed of the one-stage table servomechanism for actual experimental setup and identified models.

Table 2 exhibits the mean square error of each identified model. It can be noted that the nonlinear ARX identified model has the minimum mean square with respect to the second-order identified model. Finally, it can be summarized that the nonlinear ARX identified model can represent the one-stage table servomechanism significantly. So, this model will be used to help us to design and implement the enhanced nonlinear PID controller.

This section demonstrates the dynamic analysis of the one-stage servomechanism system prototype using the enhanced NPID control and linear PID control based on the HS tuning system. To ensure the robustness of the enhanced NPID control, several tests are performed such as fixed table position reference and variable table position reference. These tests are applied to the identified model (simulation results) and one stage servomechanism system prototype (experimental results). The purpose of the proposed controllers is to follow a certain position reference trajectory in a short time with good accuracy.

4.2. Performance of proposed controllers

Fig. 14 shows the performance of a one-stage servomechanism drive system with each control technique at a constant preselected position reference test where the reference of stage position is adjusted to 7 inches. The proposed controllers are applied to the identified model (simulation results) and the experiment setup (experimental results). The stage begins moving from the zero position to the reference position. The behavior of the stage from zero position to the reference position depends on the type of each control technique. It is noted that the performance of NPID controller is better than the PID controller where the NPID controller can reach the reference position rapidly compared to the PID controller. Also, the simulation results expect the behavior of each control technique without using the actual experimental setup. Small differences between the simulation and the experimental results because of the noise, system uncertainty, and NI card time delay. Both simulation and experimental results are summarized in Table 3.

Table 2. The mean square error of identified models.

No.	System identification method	Mean square error
1	Linear identified system (Second order)	0.1973
2	Nonlinear identified system	0.05912

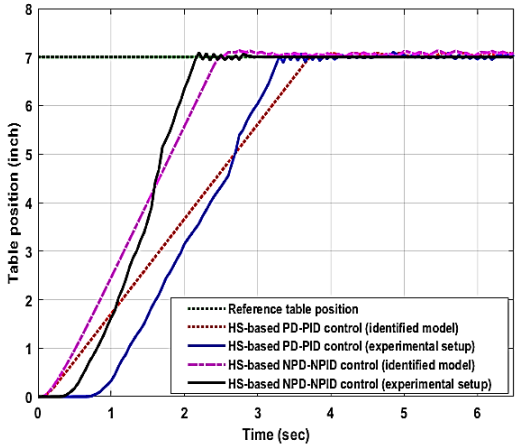

Fig. 14. The response of each control technique at a constant reference position applied to the identified model and the experimental setup.

The speed controller output for each control type is shown in Fig. 15 through a constant reference position applied to the identified model and the experimental setup. It is clear that the simulation and practical results are close. Also, there is shuttering at the steady-state in each control technique due to the noise and the friction.

Fig. 16 shows the performance of each control technique at different commands of reference position test where the reference position is changed through this experiment. It can be noted that the NPID controller can follow the reference position rapidly compared to the PID controller where the proposed NPID control is approximately 40% faster than the PID control with which leads to save the machining time cycles and increase the productivity.

Fig. 17 demonstrates the speed controller output for each control type applied to the identified model and the experimental setup at different commands of the reference position. It is clear that the simulation and practical results are identical approximately. Moreover, the shuttering is appear when the stage reaches the required reference position command because of the noise and the friction between the nut and the screw.

Table 3. Controller techniques performance.

Controller type		Rise time	Settling time	Overshoot
Linear PD – PID controller	Identified model	2.6625	3.3963	1.1772
	Experimental setup	1.9810	3.6529	0.7861
Nonlinear PD - PID controller	Identified model	1.5492	2.4466	0.6991
	Experimental setup	1.2958	2.1295	1.2964

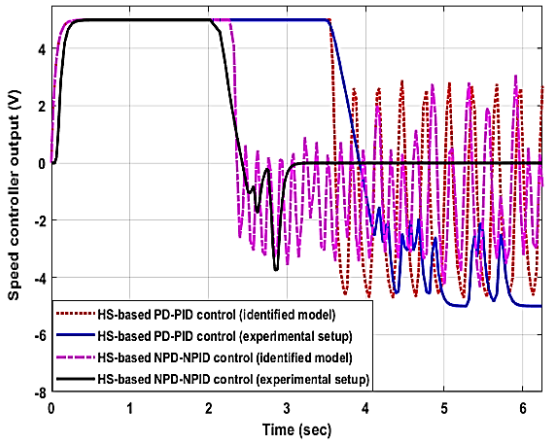

Fig. 15. The output of each control technique at a constant reference position applied to the identified model and the experimental setup.

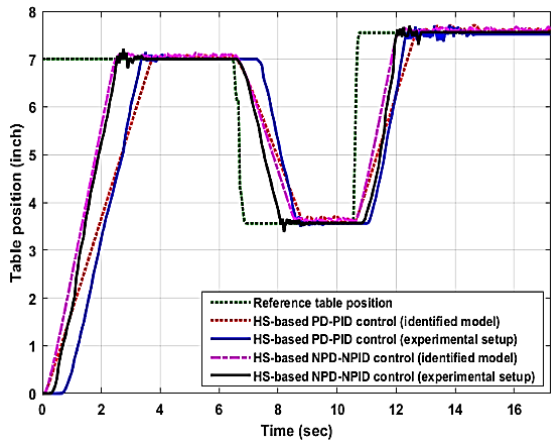

Fig. 16. The response of each control technique at different commands of reference position applied to the identified model and the experimental setup.

5. Conclusions

A LABVIEW implementation for a new form of nonlinear PID (NPID) control is presented to achieve high-performance motion control of a one-stage servomechanism system. Firstly, an identified model is implemented by collecting the experimental input/output data and entering it into the MATLAB system identification toolbox. Secondly, harmony search optimization is used to obtain the optimum values of controller parameters based on a certain cost function. Lastly, the simulation results are executed in real-time by LABVIEW software. Also, the performance of an enhanced NPID controller compared to the linear PID controller to ensure the robustness.. Processing experiments demonstrate that the enhanced NPID controller is more robust and can accommodate rapidly the position error compared to the linear PID control. The enhanced NPID control has a minimum rise and a settling time which reduce the machining cycles times in industrial processes.

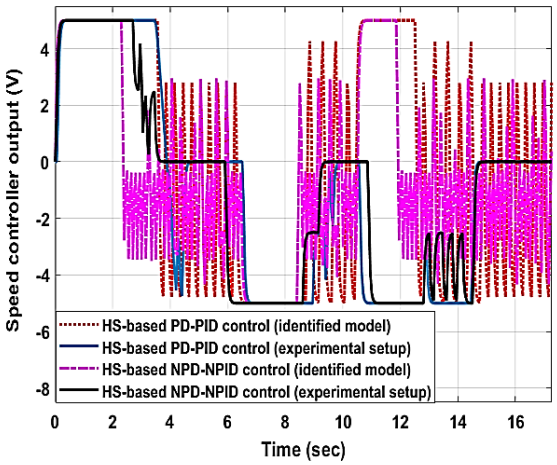

Fig. 17. The output of each control technique at different commands of reference position applied to the identified model and the experimental setup.

References

[1] E. Yuliza, H. Habil, R. A. Salam, M. M. Munir and M. Abdullah, "Development of a Simple Single-Axis Motion Table System for Testing Tilt Sensors," *Procedia Eng.*, Vol. 170, pp. 378–383,

(2017).

[2] P. Zhao, J. Huang and Y. Shi, "Nonlinear dynamics of the milling head drive mechanism in five-axis CNC machine tools," *Int. J. Adv. Manuf. Technol.*, Vol. 91, No. 9, pp. 3195–3210, (2017).

[3] P. Perz, I. Malujda, D. Wilczy and P. Tarkowski, "Methods of controlling a hybrid positioning system using LabVIEW," *Procedia Eng.*, Vol. 177, pp. 339-346, (2017).

[4] F. L. Li, M. L. Mi and Y. Z. N. Jin, "Friction identification and compensation design for precision positioning," *Springer*, Vol. 5, pp. 120-129, (2017).

[5] M. Irfan, M. Effendy, N. Alif, S. Lailis, I. Pakaya and A. Faruq, "Performance Comparison of Fuzzy Logic and Proportional-integral for an Electronic Load Controller," *Int. J. Power Electron. Drive Syst.*, Vol. 8, No. 3, pp. 1176-1183, (2017).

[6] A. Franchi and A. Mallet, "Adaptive Closed-loop Speed Control of BLDC Motors with Applications to Multi-rotor Aerial Vehicles," *IEEE International Conference on Robotics and Automation (ICRA) Singapore*, No. 978, pp. 5203-5208, (2017).

[7] S. Wen, T. Wang, Z. Ma and X. Li, "Dynamics Modeling and Fuzzy PD Control of Humanoid Arm," in *Proceedings of the 36th Chinese Control Conference*, No. 3, pp. 616-621, (2017).

[8] M. Engineering and S. Issn, "Second order sliding mode control for direct drive positioning system," *J. Mech. Eng. Sci.*, Vol. 11, No. 4, pp. 3206-3216, (2017).

[9] V. Nguyen and C. Lin, "Adaptive PD Networks Tracking Control with Full-State Constraints for Redundant Parallel Manipulators," in *IFSA-SCIS*, No. 4, pp. 0-4, (2017).

[10] R. Madiouni, "Robust PID Controller Design based on Multi-Objective Particle Swarm Optimization Approach," in *ICEMIS2017*, pp. 1-7, (2017).

[11] M. Chang, G. Guo and S. Member, "Sinusoidal Servocompensator Implementations With Real-Time Requirements and Applications," *IEEE Trans. Control Syst. Technol.*, Vol. 25, No. 2, pp. 1–8, (2016).

[12] Altuğ İftar, "Robust Servomechanism Problem for Robotic Systems Described by Delay-Differential-Algebraic Equations," *IEEE 7th Int. Conf. CIS RAM*, Vol. 2, No. 1, pp. 13-18, (2015).

[13] J. Cloutier, "*Simulation and Control of a Ball Screw System Actuated by a Stepper Motor with Feedback by a Stepper Motor with Feedback*," Masters Thesis., (2014).

[14] C. Abeykoon, "Control Engineering Practice Single screw extrusion control : A comprehensive review and directions for improvements," *Control Eng. Pract.*, Vol. 51, pp. 69-80, (2016).

[15] M. Omar, M. A. Ebrahim, A. M and F. Bendary, "Tuning of PID Controller for Load Frequency Control Problem via Harmony Search Algorithm," *Indones. J. Electr. Eng. Comput. Sci.*, Vol. 1, No. 2, pp. 255-263, (2016).

[16] B. Feng, D. Zhang, J. Yang and S. Guo, "A Novel Time-Varying Friction Compensation Method for Servomechanism," *Hindawi Publ. Corp. Math. Probl. Eng.*, Vol. 2015, p. 16, (2015).

[17] B. Zhang, G. Cheng and J. Hu, "An Expanded Proximate Time-optimal Servo Controller Design for Fast Set-point Motion," *Proc. 35th Chinese Control Conf. July*, No. 2, pp. 4465-4470, (2016).

[18] C. Wang, M. Yang, W. Zheng, X. Lv, K. Hu and D. Xu, "Analysis of Limit Cycle Mechanism for Two-mass System with Backlash Nonlinearity," *Major Proj. Minist. Sci. Technol. China*, pp. 500-505, (2016).

[19] W. Lee, C. Lee, Y. Hun and B. Min, " Friction compensation controller for load varying machine tool feed drive," *Int. J. Mach. Tools Manuf.*, Vol. 96, pp. 47-54, (2015).

[20] L. Abdullah, Z. Jamaludin, Q. Ahsan, J. Jamaludin, N. A. Rafan, C. T. Heng, K.

Jusoff and M. Yusoff, "Evaluation on Tracking Performance of PID , Gain Scheduling and Classical Cascade P / PI Controller on XY Table Ballscrew Drive System" World Applied Sciences Journal, Vol. 21, pp. 1-10, (2013).

[21] T. Nadu and P. Magnet, "Modeling and Implementation of Intelligent Commutation System for BLDC Motor in Underwater Robotic Applications," in *1st IEEE International Conference on Power Electronics. Intelligent Control and Energy Systems (ICPEICES-2016) Modeling*, pp. 1-4, (2016).

[22] Y. X. Su, D. Sun and B. Y. Duan, "Design of an enhanced nonlinear PID controller," *Mechatronics*, Vol. 15, No. 8, pp. 1005–1024, (2005).

[23] S. B. U, "Multivariable Centralized Fractional Order PID Controller tuned using Harmony search Algorithm for Two Interacting Conical Tank Process," in *SAI Intelligent Systems Conference 2015 November 10-11,|London, UK*, pp. 320-327, (2015).

[24] P. Zhao and Y. Shi, "Robust control of the A-axis with friction variation and parameters uncertainty in five-axis CNC machine tools," *J. Mech. Eng. Sci.*, Vol. 228, No. 14, pp. 2545–2556, (2014).

[25] B. B. Reddy, "Modelling and Control of 2-DOF Robotic Manipulator Using BLDC Motor," *Int. J. Sci. Eng. Technol. Res. (IJSETR), Vol. 3, Issue 10, Oct. 2014 Model.*, Vol. 3, No. 10, pp. 2760-2763, (2014).

[26] M. A. Shamseldin and A. A. El-samahy, "Speed Control of BLDC Motor By Using PID Control and Self-tuning Fuzzy PID controller," *15th International Workshop on Research and Education in Mechatronics (REM)*, (2014).

[27] D. V. L. N. Sastry and M. S. R. Naidu, "An Implementation of Different Non Linear PID Controllers on a Single Tank level Control using Matlab," *Int. J. Comput. Appl.*, Vol. 54, No. 1, pp. 6-8, (2012).

[28] A. A. Elsamahy and M. A. Shamseldin, "Brushless DC motor tracking control using self-tuning fuzzy PID control and model reference adaptive control," *Ain Shams Eng. J.*, Vol. 9, No. 3, pp. 341-352, (2016).

[29] M. Omar, A. M. A. Ghany and F. Bendary, "Harmony Search based PID for Multi Area Load Frequency Control Including Boiler Dynamics and Nonlinearities," *WSEAS Trans. CIRCUITS Syst.*, Vol. 14, pp. 407-414, (2015).

[30] M. A. Ebrahim and F. Bendary, "Reduced Size Harmony Search Algorithm for Optimization," *J. Electr. Eng.*, pp. 1-8, (2016).

How to cite this paper:

Mohamed A. Shamseldin, Mohamed Sallam, A. M. Bassiuny and A. M. Abdel Ghany, *"*LabVIEW implementation of an enhanced nonlinear PID controller based on harmony search for one-stage servomechanism system
", Journal of Computational and Applied Research in Mechanical Engineering, Vol. 10, No. 1, pp. 111-123, (2020).

DOI: 10.22061/jcarme.2019.2959.1308

URL: http://jcarme.sru.ac.ir/?_action=showPDF&article=1045