



Position Control of a Single Arm Manipulator Using GA-PID Controller

Y.S. El-Koteshy* A.S. El-Wakeel[†], A.E. Elawa[†]

Abstract: This paper demonstrates in detail how to employ the genetic algorithm (GA) optimization technique to search efficiently the optimal proportional-integral-derivative (PID) controller gains to control the position of a fixed arm manipulator system. The system identification technique is used to find an equivalent transfer function for the system under study. GA is applied off-line to find the optimal PID controller parameters based on the identified model. The experimental and simulation results of the actual system and its identified model under the influence of the optimal PID controller are explored. The proposed approach shows superior features, including easy implementation, stable convergence characteristic, and good computational efficiency.

Keywords: Fixed arm manipulator system, System identification, Genetic algorithm, PID controller.

I. Introduction

During the past decades, the process control techniques in the industry have made great advances. Numerous control methods such as adaptive control, neural control, and fuzzy control have been applied [1, 2]. Among them, the best known is the PID controller, which has been widely used in the industry because of its simple structure and robust performance in a wide range of operating conditions [3].

The tuning problem of PID controller consists of determining the values of the three gains, K_p , K_d , and K_i with the aim of satisfying different control specifications such as set-point control and tracking control [4].

Over the years, several heuristic methods have been proposed for the tuning of PID controllers such as the classical tuning rules proposed by Ziegler and Nichols [5]. In general, it is often hard to determine optimal or near optimal PID parameters with the Ziegler-Nichols formula in many industrial plants burdened with problems such as high order, time delay, and nonlinearities. For these reasons, it is highly desirable to increase the capabilities of traditional PID controller tuning techniques [6].

Many stochastic search methods, such as genetic algorithm (GA) have recently received much interest for achieving high efficiency and searching global optimal solution in problem space. GA has parallel search techniques, which emulate natural genetic operations. Due to its high potential for global optimization, GA has received great attention in control systems such as the search of optimal PID controller gains [7, 8].

However, the evolutionary process adapted by the genetic algorithm technique requires performing a large number of closed-loop step responses on the system under study. This

* Egyptian Armed Forces, Egypt. yasser.elkoteshy@hotmail.com

[†] Egyptian Armed Forces, Egypt.

clearly precludes this technique. In order to circumvent this problem, it is firstly proposed to use the real-process data in the form of open or closed-loop step response to identify a model of the system under study (using system identification technique) [9]. The GA can then be used off-line using simulation methods to tune the PID controller so as to minimize a time or frequency domain based optimization function. Finally, when the tuning process is complete, the optimum PID controller can be applied on-line to the real system, hence, completing the auto-tuning procedure [10].

In the first part of this paper, the explanation and hardware implementation of the system under study is considered. In the second part, an introduction and application of system identification technique to the collected data from the real system are presented. In the third part, GA is applied to find the optimum PID controller for the system under study. The experimental and simulation results of the actual system and its identified model under the influence of the optimum PID controller are evaluated. Finally the conclusion and references are explored.

II. Fixed Arm Manipulator System

A. Explanation of the System under Study

The mechanical system under study is a fixed arm manipulator system. The position control of the fixed arm can be selected to be manually controlled using a driving wheel or electrically controlled using DC motors. The selection between manual and electric (automatic) position control is achieved using a mechanical clutch.

The system is actuated by a 1-HP, 2000 r.p.m, 110 V separately excited DC motor. A 1/1000 reducer is coupled to the motor; hence, the maximum speed of the arm is 2 r.p.m. The motion is transferred from the reducer to the arm through a rack and pinion system.

The angle of rotation of the arm has maximum and minimum values with respect to the datum (zero position) due to mechanical consideration. The maximum and minimum angles of rotation are 84° and 10° respectively.

B. Hardware Implementation

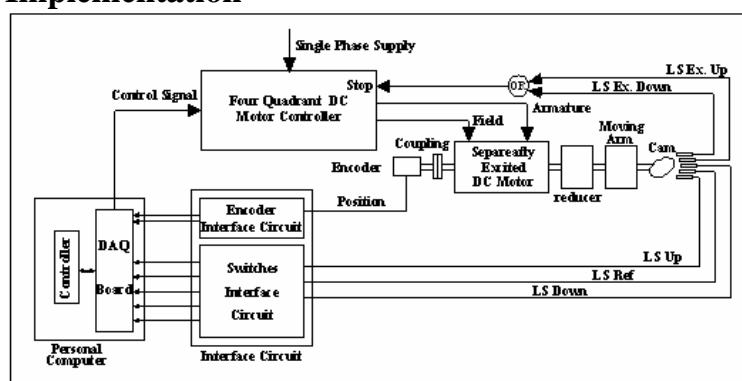


Fig. 1 Schematic diagram of the proposed digital position control system

The schematic diagram of the proposed digital closed loop control system is shown in figure (1). The proposed digital control system is composed of the following components:

- 1) **Position Sensors:** An incremental encoder of 2500 PPR is used as a position sensor for the arm. The encoder is supported on the trailing side of the motor casing to reduce the measuring error as shown in figure (2).

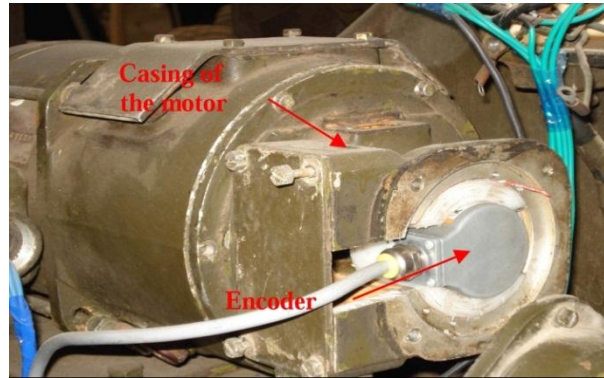


Fig. 2 Fixation of the shaft encoder

- 2) **Motor Drive:** A four-quadrant converter is used as a motor drive. The input to the four-quadrant converter is a 220 V, 50 HZ supply and its output is a variable DC voltage (0 to ± 180 V DC). The output voltage from the converter is adjusted according to the control signal from the system controller by adjusting the firing angle of the converter [11].
- 3) **Limit and Proximity Switches:** To prevent the angle of rotation of the arm from exceeding the maximum angle of rotation, two limit switches (*LS UP* & *LS EX UP*) are used as shown in figure (3). The limit switch *LS UP* is supported in a position to be *ON* when the angle of rotation $\geq 84^\circ$. Hence the control program recognizes the switch action and gives a zero control signal to stop the motor (software limitation). For more safety the limit switch *LS EX UP* is also supported in a position to stop the system when the angle of rotation $\geq 86^\circ$ by enabling the stop terminal of the converter (hardware limitation).

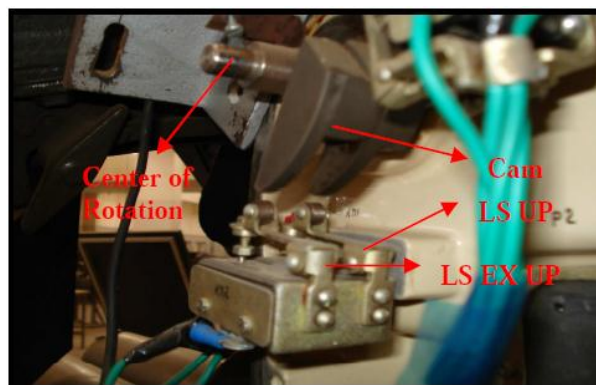


Fig. 3 Position limit switches *LS UP* & *LS EX UP*

By the same way, two another limit switches are used to limit the minimum angle of rotation. The limit switch *LS DOWN* is used to stop the system when the angle of rotation $\leq 10^\circ$ (software limitations), the limit switch *LS EX DOWN* is used to stop the system when the angle of rotation $\leq 8^\circ$ by enabling the stop terminal of the converter (hardware limitations). The fixation of the limit switches with respect to the system is shown figure (4).

An inductive proximity sensor *PS REF* is used to adapt the system in the reference position at the operation starting. The reference position is a fixed position relative to the datum and may take any angle between 10° : 84° and called a reference angle.

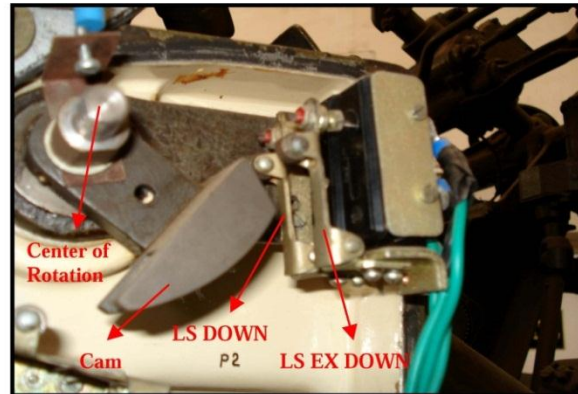


Fig. 4 Position of limit switches *LS DOWN* & *LS EX DOWN*

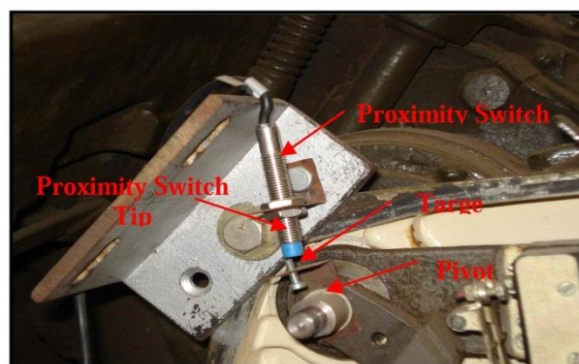


Fig. 5 Fixation of a proximity sensor

- 4) **Interface Circuit:** The designed interface circuit as shown in figure (6) has four parts (a), (b), (c), and (d). These parts are described in the following points:
- Switches interface unit (*part a*): This circuit is designed to sense the activation and deactivation of limit and proximity sensors and gives a digital signal equivalent to the state of each sensor.
 - Encoders interface unit (*part b*): It is simply a quadrature clock converter which converts the output of the encoder into TTL output for providing a high measurement resolution [12].
 - 220V AC/24V DC regulated power supply (*part c*).
 - 220V AC/5VDC regulated power supply (*part d*).
- 5) **Data Acquisition Board Model NI-6036E:** The data acquisition board (DAQ) is used to accept five digital inputs from switches interface circuit and one encoder input from encoder interface circuit. It is also used to give an analogue control signal to the converter.
- 6) **Actuator:** A separately excited 110-V, 1-HP motor is used as an actuator for the system under study.

C. Software Implementation

The control program for the arm is designed using MATLAB SIMULINK package. It is designed to control the operation of the data acquisition unit. The program performs the measurement of the output signals from transducers (encoders, limit switches and proximity sensors), compute the control signal based on the controller strategy, and apply it in analogue form (0 to +/-10V) to the four quadrant converter to drive the motor.

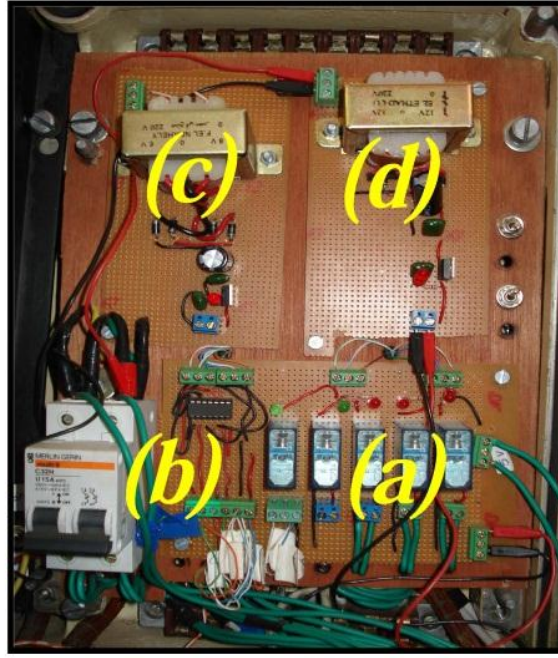


Fig. 6 Interface circuit

III. System Identification

Nowadays the model based controller design is widely used. The first step in designing the controller is to model the plant and controller. System identification is the process of creating models of a wide range of dynamic process from input-output signals without knowledge of the actual system physics. The aim of system identification can be defined as “to find a model with adjustable parameters and then to adjust them so that the predicted output matches the measured output” [13].

A. Steps of System Identification [14]

- 1) **Experiment Setup and Data Collection:** An experiment is prepared to test the practical system with input signals (step/impulse, random binary signal, pseudo-random binary signals, multi-sine) and the related output data are collected.
- 2) **Data pre-processing:** Pre-processing means preparing collected data for system identification. Many issues indicates need for pre-processing such as missing or faulty values, offsets and drift in signal levels, nonlinearities in data, transforming data from one type into another, and splitting the collected data into estimation (identification) data and validation data.
- 3) **Model Structure Selection:** Models of system identification can be classified into two main categories, parametric and non-parametric models. Non-parametric models consist of data tables or curves and are not represented by a compact mathematical formula with adjustable parameters. Parametric models have well defined mathematical structures; these structures are fit to the input-output data by adjusting the model parameters. There are several parametric models structures such as *AR*, *ARX*, *AMAX*, *BJ*, *OE*, and *SS*. The parametric model structures differ by how many of polynomials are included in the general-linear polynomial model given by equation (1).

$$A(s)Y(s) = \frac{B(s)}{F(s)}U(s) + \frac{C(s)}{D(s)}E(s) \quad (1)$$

where, $A(s)$, $B(s)$, $C(s)$, and $D(s)$ are polynomials in the frequency domain, $Y(S)$ and $U(S)$ are the input and the output of the system. $E(s)$ is the input noise to the system.

- 4) **Parametric Estimation:** Parametric estimation method uses numerical search to find the parameter values that correspond to the best agreement between simulated and measured output Model validation.
- 5) **Model Validation:** The validity of the identified model is checked by testing the model using an actual input data. The percentage of fitting and residuals between actual output data and the identified model output data are calculated.

B. Application of System Identification to the System

The experiment is designed as shown in figure (7) where the input signal is a control voltage from 0V to +/-10V and the output signal is the speed of rotation in *r.p.m.*

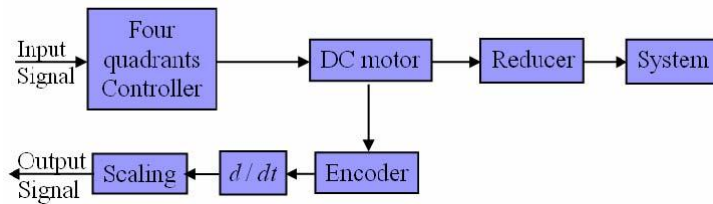


Fig. 7 Scheme of experiment for system identification

The system is tested by step input with different amplitudes 1V, 2V, 3V, 4V and 5V, the output speed response of the system for each input is shown in figure (8).

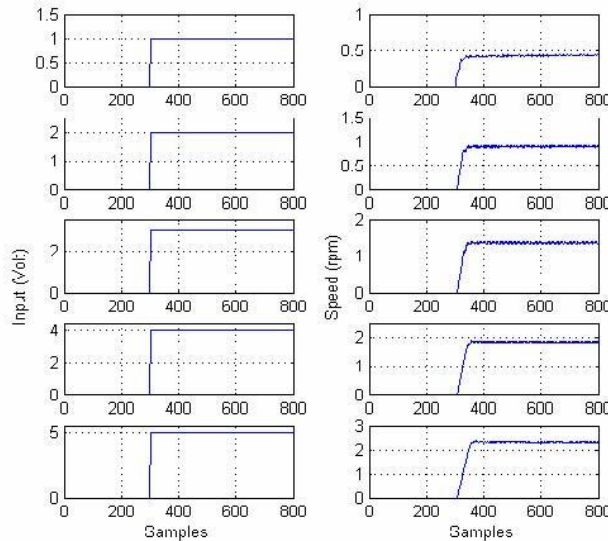


Fig. 8 The step response of the system under study

The ARX model is used to model the practical system because this model describes both the system dynamics and noise properties using the same set of poles. Since the system under study is of order tow, a first and second order ARX model is tested. The first and second order ARX models give 97.02 % and 96.94 % percentage of fitting between the test and validation data respectively. So there was no significant loss of accuracy in reducing the order of the model from 2 to 1. The transfer function of the first order ARX model is given by equation (2) which is the relation between the output speed and the input control voltage.

$$G_P(s) = \frac{2.5956 e^{-0.07s}}{(1+0.1656 s)} \quad (2)$$

The transfer function describes the output position to the input control voltage is given by equation (3).

$$G_P(s) = \frac{2.5956 e^{-0.07s}}{s(1+0.1656 s)} \quad (3)$$

IV. Tuning of PID Controller with GA

A. PID Controller

The classical PID control system can be described as shown in figure (9) and the typical PID control law in its standard form is given by equation (4).

$$u(t) = K_p e(t) + K_d \frac{de(t)}{dt} + K_i \int_0^t e(\tau) d\tau \quad (4)$$

where, $e(t)$ is the system error (difference between the reference input and the system output) K_p is the proportional gain, K_d is the derivative gain, and K_i is the integral gain.

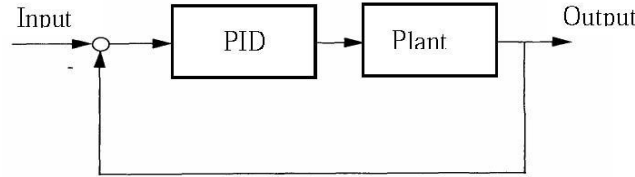


Fig. 9 Block diagram of classical PID control system

B. Genetic Algorithm

Genetic algorithm (GA) is a simple, powerful, general-purpose, derivative-free, stochastic global optimization method inspired by the laws of natural selection and genetics. GA algorithm is derivative-free, which means that it does not need functional derivative information to search for a set of parameters that minimize a given fitness function (the function which measures the quality of a particular solution) [15]. There are three main steps for genetic algorithm; these steps can be summarized as follows [16]:

- 1) **Random initialization of population:** An initial population is created randomly or heuristically. In general, there are n individuals (points in the search space) in the population and n is an even number.
- 2) **Fitness evaluation of each individual:** In this step, all the individuals of the initially created population are encoded using a one of the encoding schemes (binary string or real number coding). Then the fitness value of each individual is evaluated by means of a fitness function (the function which measures how close each individual meet the required specifications).
- 3) **New population generation:** The overall goal of this step is to obtain a new population of individuals which have high fitness values and is performed basically in three stages, reproduction, crossover and mutation.
 - **Reproduction stage:** The individuals of the initial population are selected into the new population, according to a probabilistic rule, which favours those individuals with higher fitness values.
 - **Crossover stage:** The crossover operator takes two of the selected parent individual and swaps part of them at a randomly selected location. This provides mechanism for the individuals to mix and match their desirable quantities in forming offspring. The crossover is applied with a crossover probability P_c .

- Mutation stage: The mutation process introduces further changes to an individual (new genetic material is introduced into a population, but this new material does not originate from the parents and is not introduced by crossover). The most common mutation technique is to change a randomly chosen bit in the bit string (for binary coding) of the individual to be mutated. The mutation is implemented with a probability of P_m .

C. PID Controller Tuning Procedures Using GA

Tuning procedures of PID controller gains using GA can be summarized in the following points [17, 18]:

- 1) The range of parameters K_p , K_i and K_d should be investigated from the stability point of view. This means applying Routh-stability test or any other stability check criteria into the system to be controlled with the PID controller and find the range of controller parameters in which the system is stable.
- 2) Let the GA generate the initial population with n individuals by random way. Each individual in the population is a concatenated binary bit string (in the case of binary coding) represents K_p , K_i and K_d .
- 3) For each individual in the population, the closed-loop control system (the candidate controller and model of the plant) is simulated, and the fitness function is calculated. There are many fitness functions used with the tuning of PID controller such as *IAE*, *ITAE*, *ISE*, and *ITSE*. In this paper, a transient response based fitness function is used. This fitness function is called weighted goal attainment function *WGAF* and is defined by the following equation (5) [19].

$$f = \frac{1}{C_1(t_r - t_{rd})^2 + C_2(M_P - M_{Pd})^2 + C_3(t_s - t_{sd})^2 + C_4(e_{ss} - e_{ssd})^2} \quad (5)$$

where, C_1 : C_4 are positive constants (weighting factor), their values are chosen according to prioritizing their importance, (t_{rd}) is the desired rise time, (M_{Pd}) is the desired maximum overshoot, (t_{sd}) is the desired settling time, and (e_{ssd}) is the desired steady state error.

- 4) The new generation is achieved by applying the three stages reproduction, crossover and mutation after choosing the method of reproduction, probability of crossover operator and mutation rate as discussed in the previous section.
- 5) The fourth and fifth steps can be repeated many times (number of generation) until there is no change in the fitness value of best individual.

V. Simulation Results

The block diagram of the closed loop position control system of the identified model with PID controller is shown in figure (10), where, θ_{ref} is the desired angle and θ is the actual angle.

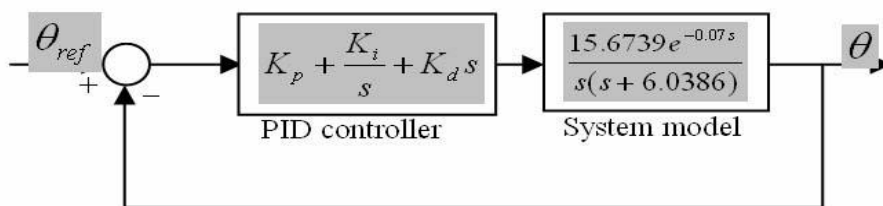


Fig. 10 Block diagram of the closed loop system with PID controller

GA is used off-line to search for the optimum PID controller to satisfy the required transient response given by table (1). By trials the GA parameters is chosen as given by table (2) and the lower and upper bounds of the PID controller gains were given by table (3). The optimum PID controller gains are given by table (4).

Table 1 Desired Response for the System

M_p	$t_r (s)$	$t_s (s)$	$e_{ss} \%$
$\leq 1\%$	0.8	1.5	$\leq 1\%$

Table 2 GA Parameters

GA Property	Value / Method
Members of each Individual	three (K_p, K_i, K_d)
Population Size	100
Coding Method	real number
Fitness Function	WGAF
Max. No. of Generations	50
Selection Method	roulette wheel
Crossover Method	arithmetic
Crossover Rate	60 %
Mutation Method	uniform mutation
Mutation Rate	1 %

Table 3 Range of the PID Controller Parameters

Parameter	Minimum	Maximum
K_p	0	3
K_d	0	1
K_i	0	0.1

Table 4 The Optimum PID Controller Parameters

PID Controller gains	Value
K_p	0.9663
K_d	0.9226
K_i	0

The transient response of the system with the optimum PID controller is shown in figure (11). The transient response parameters are given in table (5).

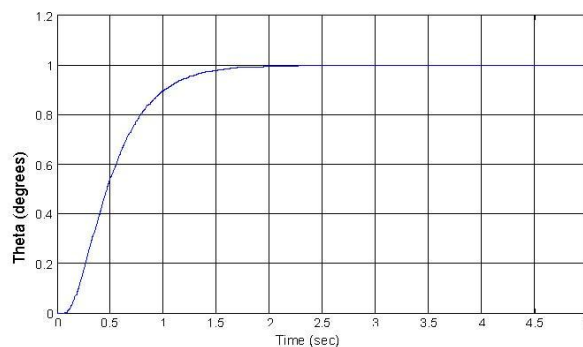


Fig. 11 Response of the system with optimum PID controller

Table 5 Response of the system with Optimum PID Controller

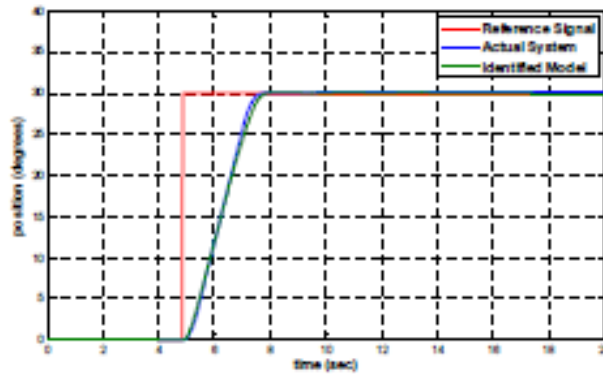
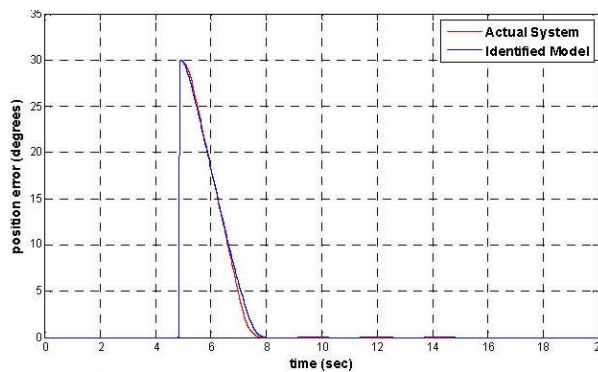
M_p	t_r (s)	t_s (s)	e_{ss} %
0	0.8092	1.4999	0

VI. Experimental Results

The optimum PID controller tuned off-line is applied on-line to the practical system. The experimental results of the practical system are compared with the simulation results of the system model using the optimum PID controller. The practical system and its identified model is tested using two control strategies, point-to-point control (picking up a target at different step angles) and tracking control (racks a sine wave input with different frequency).

A. Point-to-Point Position Control

- 1) **Picking up a Target at “30°”**: Figures (12) and (13) show the position response and position error of the practical system and its model to pick up the target respectively. Table (6) shows the transient response parameters of the practical system and its model to pick up the target.

**Fig. 12 Position response of picking up a target at 30°****Fig. 13 Position error of picking up a target at 30°****Table 6 Transient response parameters of picking up a target at 30°**

Parameter	Actual S t	Identified M d l
$M_p\%$	0.0547	0
$T_s(sec)$	2.6154	2.7823
$T_r(sec)$	1.7702	1.9726
$e_{ss}\%$	0.05	0

2) *Picking up a Target at Multi Step Angles:* Figures (14) and (15) show the position response and position error of the practical system and its model to pick up the target.

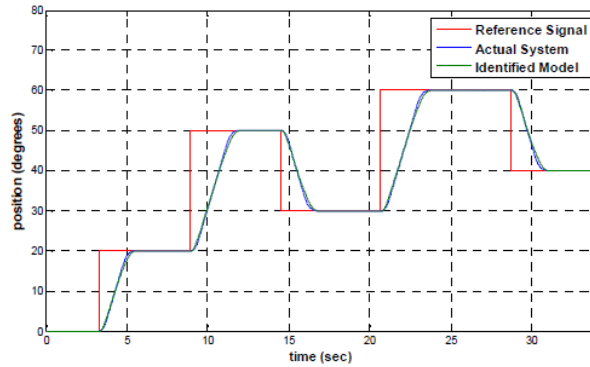


Fig. 14 Position response of picking up a target at multi-step angles

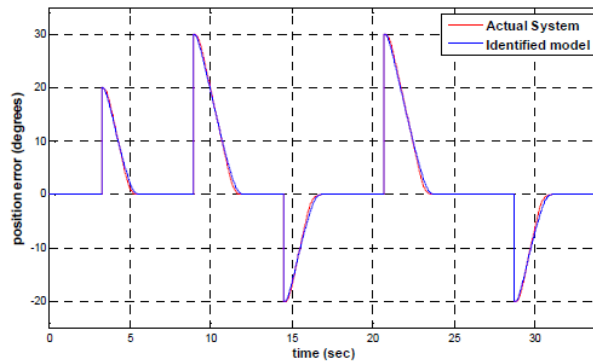


Fig.15 Position error of picking up a target at multi-step angles

B. Position Tracking Control

1) *Tracking of Target1:* The position of *target1* is assumed to vary as a sine wave with amplitude of 20° and frequency of 0.1 Hz . Figures (16) and (17) illustrate the response of system and its model to track the target.

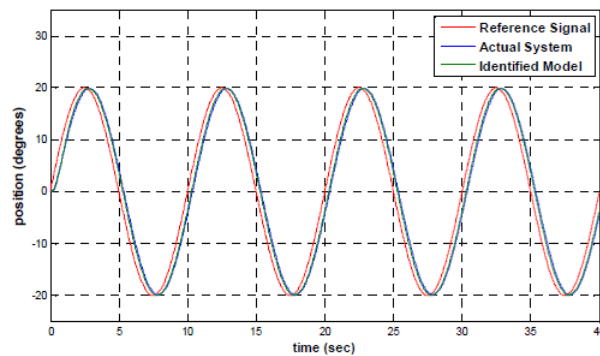


Fig. 16 Position tracking of *target 1*

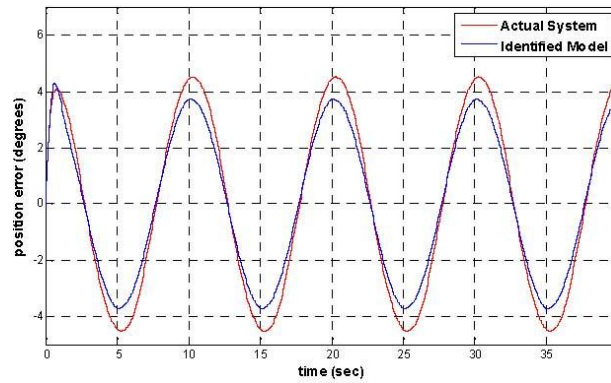


Fig.17 Position error of target 1

2) *Tracking of Target2*: The position of *target2* is assumed to vary as a sine wave with amplitude of 20° and frequency of 0.05 Hz . Figures (18) and (19) illustrate the response of the system and its identified model to track target.

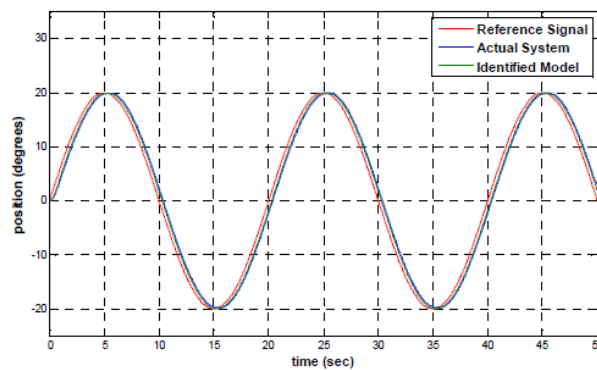


Fig. 18 Position tracking of target 2

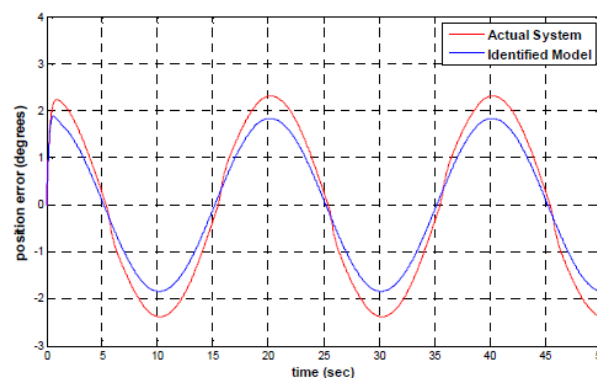


Fig. 19 Position error of target 2

3) *Tracking of Target3*: The position of *target3* is assumed to vary as a sine wave with amplitude of 60° and frequency of 0.00625 Hz . Figures (20) and (21) illustrate the response of the system and its identified model to track target.

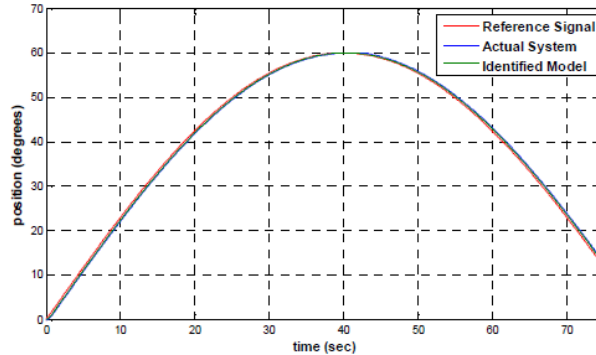


Fig. 20 Position tracking of target 3

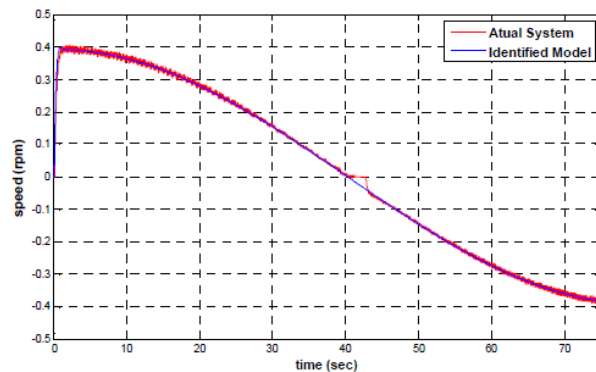


Fig. 21 Position error of target 3

The results show good controller behaviour under both set-point and tracking control. The difference between the results of actual system and its identified model is negligibly small.

VII. Conclusion

Application of GA-PID controller to control the position of a fixed arm manipulator system has been tested in this paper. Although the genetic algorithm solve the tuning problem by a simple way that does not need an experienced control system designer, the proposed controller has proved to be effective under both set-point and tracking control.

Practical selection of the ranges for crossover and mutation probabilities and the stopping criteria are well known. However this is not the case for population size. Several population sizes were tried until the variations in the final solution can be neglected. This technique is time consuming but very successful.

System Identification technique was introduced in this work to estimate the transfer function of the system from the input-output test data. From the results which show a comparison between the practical system and its identified model, it can be found that the error between them is very small and can be neglected.

References

- [1] A.Visioli, "Tuning of PID Controllers with Fuzzy Logic", IEE Proceeding in Control Theory Applications, Vol. 148, Issues 1, January 2001.
- [2] Seng TL, Khalid MB, Yusof R. "Tuning of a neuro-fuzzy controller by genetic algorithm". IEEE Transaction on Systems Man, and Cybernetics, Volume 29 Issue 2, pp. 226–236 1999.

- [3] K. J. Astrom and T. Hagglund, “*PID Controller: Theory, Design and Tuning*”, Instrument Society of America, Research Triangle Park, NC, Second edition, 1995.
- [4] Markus Olsson, “*Simulation Comparison of Auto-Tuning Methods for PID Control*”, Master of Science, Linköpings University, Linköping, 2008.
- [5] J. G. Ziegler, N. B. Nichols, “*Optimum Settings for Automatic Controllers*”, Transactions of ASME, Vol. 64, pp. 759-768, 1942.
- [6] P. Cominos and N. Munro, “*PID Controllers: Recent Tuning Methods and Design to Specification*”, IEE Proceeding on Control Theory Applications, Vol. 149, Issues 1, January 2002.
- [7] Kristiansson B, Lennartson B. “*Robust and optimal tuning of PI and PID controllers*”. IEE Proceeding in Control Theory Applications, 46–53, 2002.
- [8] Zwe-Lee Gaing, “A Particle Swarm Optimization Approach for Optimum Design of PID Controller in AVR System”, IEEE Transactions on Energy Conversion, Vol. 19, Issues 2, 2004.
- [9] Chih-Cheng Kao, Chin-Wen Chuang, and Rong-Fong Fung, “The Self-tuning PID Control in a Slider–crank Mechanism System by Applying Particle Swarm Optimization Approach”, Mechatronics, Vol. 16, pp. 513–522, 2006.
- [10] A. H. Jones and P. B. Oliveira, “*Genetic Auto-Tuning of PID controllers*”, IEE International Conference on Genetic Algorithms in Engineering Systems: innovations and Applications, 1995.
- [11] Instillation and Operating Instructions for Regenerative Drive, Model KBRG-255 (5HP), Full Wave, Four Quadrant, KB Electronics Inc., 1999.
- [12] National Instrument Application Note 084, “*Using Quadrature Encoders with E-series DAQ Boards*”, National Instrument Corporation, 1996.
- [13] Lennart Ljung, “*System Identification: Theory for the User*”, Second Edition, Prentice-hall International Inc., 1999.
- [14] Lennart Ljung, “*System Identification Toolbox 7: User’s Guide*”, MathWorks Inc., 2007.
- [15] D. A. Coley, “An Introduction to Genetic Algorithms for Scientists and Engineers”, World Scientific, London, 2000.
- [16] L. Davies, “Genetic Algorithm Handbook”, Van Nostrand, New York, 1995.
- [17] Neenu Thomas, P. Poongodi, “*Position Control of DC Motor Using Genetic Algorithm based PID Controller*”, Proceeding of the World Congress on Engineering, London, U.K., Vol. II, July, 2009.
- [18] S.M.Giriraj, R. Jain, N.Anantharaman, “*Genetic Algorithm based PID Controller Tuning for a Model Bioreactor*”, Indian Institute of Chemical Engineers, Vol. 50 No. 3, PP. 214-226, September 2008.
- [19] Yasser S. El-koteshy, “*Modern Control Strategies of Electric machines*”, Master of Science, Military Technical College, Cairo, Egypt, 2009.