Abstract— A DC servo motor system is one of the widely used variable speed drives in industrial production, process automation and building technology worldwide. They are intended and designed to be used in motion control applications which require high accuracy positioning, quick reversing and exceptional performance. It is a widely used device for many applications, starting from simple industrial tasks to precise manufacturing applications such as in robotics. It uses a PID controller in order to obtain an error feedback signal, and thus produces a stable angular speed. However, PID controllers produce various problems such as poor and unstable responses during the frequent presence of disturbance and noise. This paper aims to implement a parametric second-order model of a servo motor via raw data extracted from real experiments, an optimization technique named Genetic algorithms (GA) was used for identifying of the model parameters, also to design an efficient PI controller that provides a fast response, less rise time and sufficient settling time. The parameters of the designed PI controller were also selected based on using a GA optimization technique with three popular common performance criteria, namely, integral square error (ISE), integral absolute error (IAE), and integrated time absolute error (ITAE). Moreover, the performance of the optimized PI controller was testified and evaluated when the servo motor is operating under varying inertia loads. The experimental results show that the GA PI controller with the ITAE criteria is more proficient in enhancing the angular speed stability and also in meeting the operational demands compared with other traditional objective functions.

Keywords: servo motor, Ziegler Nichols (ZN), PI controllers, parameter identification, Genetic algorithm, Optimization techniques.

Index Terms—first term, second term, third term, fourth term, fifth term.

I. INTRODUCTION

Servo motors have been widely used in a variety of industrial applications such as in robotics that require accurate positioning as well as speed control [1].

In servo motors, signals sent from the feedback controller are used to control the position and the speed of the motor output. PID controller is one of the most conventional controllers preferred for speed control of motors [2], they are still applied in the majority of real world servo systems [3] due to their robustness and simple structure.

There are many tuning techniques for PID such as Ziegler Nichols (ZN) [4] which works properly for many simple processes, but can’t offer good tuning for non-linear systems which causes surge and big overshoots [5], another drawback of ZN method is the dependency on a priori knowledge for the parameters of the DC motor. Therefore, in order to cope with this problem, a procedure of optimization for more efficient control design has to be used [6].

There are many stochastic optimization techniques introduced by many researchers which are based on the concepts natural biological evolution, such as Genetic Algorithm (GA) [2, 7-12], Particle Swarm Optimization (PSO) [13-15] and Artificial Neural Networks (ANN) [16-18] and ANT-Colony Optimisation [19, 20]. PSO was first introduced by Kennedy and Ebenhart [21], the idea was inspired from the behaviour of organisms, such as fish schooling, the PSO then enhanced by Holland from the university of Michigan in 1970s which the evaluation rules DC were introduced for the first time to cope with the optimization problems. After that, the PSO was enhanced and developed to solve non-linear problems that contain a number of variables as multiplies [21, 22].

The GA algorithms have been successfully used for solving complex optimization problems, they provide practical solutions in the calculation of different controller parameters due to their quick convergence and high accuracy [23].

Problem description: During the design of an optimized PID for electric motors, it has been assumed that their parameters never vary during the operation, but in most of practical applications, the mechanical load parameters of motors such as inertia and friction may vary because of coupling or decoupling inertial parameters, and also due to the change in load. It has been notified that the ratio of no load to full load friction is 1 to 15, and the variation in
the moment of inertia [24-26] has been increased by more than 10 times because of coupling/decoupling inertia elements for speed and positioning control applications [27].

In this paper, the design of GA PID controller based on DC servo drive will be introduced, the servo motor (namely CE110) was first modelled based on its real data which stored and extracted via software called CE2000, the GA algorithm was implemented and applied with three fitness functions for both identifying of motor parameters and finding the optimum values for the PID controller that achieve the operational requirements of the servo. Moreover, a number of experiments have been carried out when the inertial load of the motor is variant in order to examine the efficiency of the designed controller to work under unprepared environment, the performance of GA PID controller was investigated under three different load conditions for motor that are based on the inertia size (small, medium and large). Also, a comparison between the performance of GA PID controller with different fitness criteria (ISE, IAE and ITAE) are applied.

In this paper, a scheduling PID tuning parameters using Genetic Algorithm (GA) optimization strategy for a DC servo motor speed control is introduced. The organization of this paper is given as follows. Section II shortly describes the basics of PID controllers and the concept of the Genetic optimization algorithms, it also highlights on the three convergence criteria used along with the algorithm. Section III describes the architecture of the servo motor system and the speed control theory that the servo relies on. It also describes the mathematical modelling of the servo motor. Section IV presents the experimental results and shows the efficiency of the proposed GA technique in identifying of model parameters and in finding the optimal PID elements, while Section V concludes the paper.

II. BACKGROUND

A. PID controller

PID controllers have been widely used in industrial applications due to their robustness and simple structure, they are preferred for speed and positioning control of motors because of the easiness to integrate with software and hardware. A PID controller can be defined as a control mechanism that utilises a combination of three terms which are used for controlling the state of the system, within this strategy, the output of the system is fed back into the control loop in order to make corrections to the behaviour of the system, and this cyclic has influence on series of events the form the control (as shown Figure 1). A PID controller determines the error value which is the difference between the measured variable and the required set point. The controller then attempts to reduce the error value by modifying the process during the use of controlled variables. [28]

The PID combines of three constant terms which are: proportional term (P), integral term (I), and derivative term (D). Proportional gain relies on current error, integral gain is dependent on the accumulation of past error, and derivative gain is an estimation of the future (upcoming) error based on the present rate of change [29]. The controller can provide a particular control action in order to meet a demand of a certain process by tuning weight parameters related with the three controller elements. The response of the PID controller can be understood and evaluated based on its response to errors and to the degree of overshoots that is over or under the required set point.

The PID controller can be mathematically represented by the following equation:

\[ u(t) = k_p e(t) + k_i \int_0^t e(\tau)d\tau + k_d \frac{de(t)}{dt} \]  (1)

Where: \( u(t) \) is the output of the controller.

\( k_p \), \( k_i \), and \( k_d \) represent proportional, integral and derivative gains, respectively.

e is the error of the system, and \( \tau \) is the instantaneous time.

![Figure 1. Notional PID Controller.](image)

The increment in the value of proportional controller will lead to increase of the response speed for the system and may drive the system to the instability. The value of the integral gain will keep increasing until the value of stability error become zero. However, in the case of using the derivative gain within the control system, it will cause rising in the sensitivity of the PID controller towards quick and sudden changes in the system.

B. Genetic algorithm

The Genetic Algorithm (GA) can be simply defined as a metaheuristic algorithm that relies on a random search method with using of a parametric coded. the GA has been preferred by many researchers for solving the complex control problems that cannot be easily solved by the analytical approaches, especially when the search space is very big. [30]

The principle of GA is based on the living organism, which the good of them will remain existent while the other will vanish or die. A child character is a mixture of parents characters, it can survive if successfully adapts with new conditions. However, any child has a probability of disappearance if contained worse character from parents. The strategy of GA for solving optimisation and search problems is based on previous solutions and one new solution. The previous solutions are obtained by parents, while the new solution is obtained by the new child via operators like mutation and crossover. One of the good points for the GAs is that the new solution is also searched within the previous good solutions. The last step is the selection of the optimum solution among best
solutions. It is worth mentioning that the performance of GA is influenced by the parameters of mutation and crossover rates. Figure 3 shows the flow diagram of GA.

C. Tuning of the controller via GA approach

In this research, the GA approach will be applied for obtaining the optimum PID gains that guarantee the best control performance and meets the performance demands of the DC servo motor. Table 4 shows the performance requirements for the servo motor which are very small peak shoots and minimum settling time and rise time. The detailed description of PI optimisation via GA approach will be discussed within section B (Control optimization).

The block diagram for the servo motor system can be illustrated in Figure 2, the system combines of the controller, the servo model and the optimisation technique.

Where \( w^* \) is the speed reference

\( w \) is the actual speed of the servo motor (the output)

![Figure 2. Structure of GA Technique of the PI Controller.](image)

Figure 3. Flowchart of GA.

D. Performance criteria of the GA

In the design methodology of a PID controller, one of the most significant performance criteria is the difference (error) between the output of the plant and the set point signal. Using this error criterion as the fitness function of the optimization algorithm causes a small overshoot but with a long settling time. Fitness functions generally depend on error equations. There are four formulas represent the error criteria which are: Integral Square Error (ISE), Integral Absolute Error (IAE), Integral Time Absolute Error (ITAE) and Integrated of Time weight Square Error (ITSE), these formulas represent the most commonly used fitness functions that use for evaluating the performances of GA [31, 32].

\[
ISE = \int_{0}^{\infty} (r(t) - y(t))^2 \, dt = \int_{0}^{\infty} (e(t))^2 \, dt \tag{2}
\]

\[
IAE = \int_{0}^{\infty} \left| r(t) - y(t) \right| \, dt = \int_{0}^{\infty} \left| e(t) \right| \, dt \tag{3}
\]

\[
ITAE = \int_{0}^{\infty} t \cdot \left| r(t) - y(t) \right| \, dt = \int_{0}^{\infty} t \cdot \left| e(t) \right| \, dt \tag{4}
\]

\[
ITSE = \int_{0}^{\infty} t \cdot (r(t) - y(t))^2 \, dt = \int_{0}^{\infty} t \cdot (e(t))^2 \, dt \tag{5}
\]

In the frequency domain, each of the aforementioned criteria has advantages and drawbacks. For instance, the drawback of the IAE and ISE criteria is that its reduction can cause a response with a small peak overshoot but the settling time will be big due to the nature of ISE which weights all errors equally independent of time. Despite the ITSE criterion can cope with drawback of ISE criterion, the derivation processes of these analytical formulas are not simple and time-consuming [33]. However, Fitness functions are not actually restricted to the aforementioned formulas. Engineers can use custom fitness functions based on the target design and control system. The total performance (convergence speed and the accuracy of the optimization) of evolutionary
algorithms is dependent on the fitness function adopted to observe the optimization search [30], therefore, it is important to select a proper fitness function that can provide satisfying optimization results.

In this paper, the GA optimization will be used for identifying of the modeled motor parameters and also for tuning of PID controllers, the GA will be used with three fitness functions which are ISE, IAE and ITAE, and their performance will be examined when the servo motor is operated with varying inertial loads.

III. EXPERIMENTAL SETUP

The DC servo motor system combines of four main parts (as shown in Figure 4), which are: a high-performance low-inertia DC servo motor (CE110), Interface circuit (CE122), Controller and PC and software (CE2000). The working principle of the used servo system can be described as follows: when a command signal is produced from the interface of the user (CE122), it goes into the positioning controller of the servo. Then the positioning controller saves the information about multi-tasks. The controller has been programmed via CE2000 software for activating the motor (i.e. varying the speed). A low power-level signal is taken and amplified via the servo control, bringing the power up to suitable levels to produce movement of the servo motor.

In this paper, the digital interface (CE122) with the designed PID controller will be used for controlling the speed of the motor within specific limits which rely on the value of the supplied voltage (supplied voltage levels for the CE110 motor are restricted between -10 v and 10 v).

A. Speed servo control methodology

The fundamental concepts of servo motion control have not varied significantly in the last 50 years. The basic reasons for using servo systems in contrast to open loop systems include the need to improve transient response times, reduce the steady state errors and reduce the sensitivity to load parameters.

Features of the servo motor used in this paper: A rotation detector (encoder) is mounted on the motor and feeds the rotation position/speed of the motor shaft back to the driver. The driver calculates the error of the analogue voltage (speed command) from the controller and the feedback signal (speed) and controls the motor rotation so the error becomes zero. The applied type of motion control method is named “Speed Control by Analogue voltage” which the analogue voltage is input to control the speed. [34]

B. Servo motor mathematical model

The model of the system is calculated by relating the torque supplied via the motor ($\tau_m$) to that required for driving the generator of the load, the flywheel and frictional losses. This can be represented by the following expression:

$$\tau_m = \text{Load Torque} + \text{Frictional Torque} + \text{Inertial Torque}$$

The model is calculated by relating the torque supplied via the motor to that required for driving the generator of the load, the flywheel and frictional losses. This can be represented by the following expression:

$$\tau_m = k_{m1} + k_{m2} + k_{m3}$$

Where:

- $k_{m1}$ is the torque supplied to the motor
- $k_{m2}$ is the friction coefficient of rotating components
- $k_{m3}$ is the gain constant for the load/generator
- $I$ is the inertia of flywheel
- $v_l$ is the load voltage from the generator

The motor electrical circuit can be represented by the equation

$$v(t) = R I + L \frac{di}{dt} + v_{bemf}$$

Where: $v(t)$ is the voltage of the motor input,
- $R$ is the armature resistance of the motor
- $L$ is the armature inductance
- $i$ is the armature current

and $v_{bemf}$ represents the back electromotive force in the DC servo motor.

The back electromotive force (bemf) and the motor torque can be represented with the motor constant $k_m$ in the following equations.

$$v_{bemf} = k_m$$

Where $k_m$ is the motor constant

$$\tau_m = k_m$$

By substituting equation (6) with equation (9)

$$k_{m1} = k_{m2} + k_{m3}$$

By taking the Laplace transform for equation (10) and rearrange it

$$i(s) = \frac{k_{m1}}{k_{m2} + k_{m3}} v_l(s) + \frac{k_{m4}}{k_{m3}} w(s)$$

By substituting equation (8) with equation (7)

$$v(t) = R I + L \frac{di}{dt} + k_m w$$

By taking the Laplace transform for equation (12)

$$v(s) = (R + Ls) \times i(s) + k_m w(s)$$

By substituting equation (11) with equation (13)

$$v(s) - \frac{R + Ls}{k_m} v_l(s) = \frac{R + Ls}{k_m} w(s)$$

By substituting equation (11) with equation (13)

$$v(s) - \frac{R + Ls}{k_m} v_l(s) = \frac{R + Ls}{k_m} w(s)$$

Because of that the servo trainer (CE110) has a voltage amplifier, so the gain of this amplifier ($k_v$) should be
considered when the gain of the servo motor is calculated. Therefore:

\[ w(s) = \frac{k_mk_v}{(R+Ls)(b+ls)+k_m} v(s) - \frac{(R+Ls)k_i}{(R+Ls)(b+ls)+k_m} v_i(s) \]  \hspace{1cm} (16) 

Where \( k_v \) is the gain of the voltage amplifier and has a known value which is 2.5 (as shown in Table 1). The inertia of the flywheel can also be calculated by the following equation:

\[ I = \frac{MR^2}{2} \]  \hspace{1cm} (17)

Where \( M \) is the mass of the flywheel thickness, and \( R \) is the radius of the shaft.

By assuming that the servo control system has only an inertial load and there is no loads applied from the generator, in this situation \( v_i(s) = 0 \) and the equation (16) can be rewritten as follows:

\[ w(s) = \frac{k_mk_v}{br + k_m b^2} v(s) \]  \hspace{1cm} (18)

The equation (18) represents the mathematical model of the servo motor in a second order form, therefore, by comparing this equation with a classic second order form that contain a steady state gain \( K \) and two time constants \( (T_1, T_2) \), the unmeasured values of real motor parameters such as the friction coefficient of rotating components \( b \) and motor constant \( (k_m) \) can be then determined from the model parameters identified via the GA algorithm.

\[ w(s) = \frac{K}{T_1T_2S^2 + (T_1 + T_2)S + 1} v(s) \]  \hspace{1cm} (19)

where: \[ K = \frac{k_mk_v}{A} \] \( T_1T_2 = \frac{L}{A} \), \( (T_1 + T_2) = \frac{(R+b)l}{A} \) and \( A = br + k_m b^2 \).

In this paper, the servo model will be built and applied based on the equation (19) and the stability of the servo motor will be examined under varying inertial loads with the use of GA PID controller. Also, an optimization technique based on GA approach will be introduced and its efficiency will be then testified when different performance criteria are used.

### IV. RESULTS AND DISCUSSION

In this paper, the experimental work can be divided into two main stages which are: i) the identification process and ii) the control process. The first step is to connect the components of the servo motor system as previously shown in Figure 4, the inertial load of the servo was also set by selecting the thickness of the flywheel (mass) through adding or removing inertia discs, the servo motor was supplied with a voltage of 10 V and run under three different inertial loads. The motor data including the input voltage and the output speed was online recorded and saved via software (CE2000), the later extracts the speed information by using a sensor (tachometer) connected to the motor shaft. Moreover, the saved data will be then used for modeling of the servo and for tuning the PI elements through Matlab toolbox. It is worth mentioning that the generator was not connected to the motor during the experiments because the influence of generator loads (external loads) was out of the scope of this paper, thus the only concentration will be on varying inertial loads (inner loads).

#### A. Model parametric identification

In this section, we describe how to identify the model parameters which are the gain constant \( k \) and two time constants \( (T_1, T_2) \) via the proposed GA technique. The first step is to define the size of inertial load, then the servo motor will be run at the highest voltage value \( (10 \text{ V}) \) in open loop mode. The next step is to build the mathematical model by using Matlab/Simulink toolbox, the input and output information extracted from the real servo via CE2000 will be imported to the model. The GA algorithm was simulated via the optimization tool in MATLAB, and both of the fitness function and initial population were first created and the search bounds were also specified in order to minimize the execution time for each iteration. Each individual was evaluated and rated based on the value of the fitness function, smaller values for the fitness function indicate to the best fit for the individual, whereas larger values for the fitness function show less fit. The best fit individual will be then used as parent and passed to the crossover and mutation processes for producing new population. Moreover, new children for new population are created from parents of the mutation and crossover process. The process is iterated till obtaining a stratifying output. The flowchart of the GA process can be illustrated in Figure 3. Table 2

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values and units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass of the flywheel thickness ((M))</td>
<td>0.684 kg, 1.368 kg, 2.052 kg</td>
</tr>
<tr>
<td>Radius of the shaft discs ((R))</td>
<td>4.5 mm</td>
</tr>
<tr>
<td>Rotor Inductance ((L))</td>
<td>5 mH</td>
</tr>
<tr>
<td>Rotor Resistance ((R_i))</td>
<td>7.8 (\Omega)</td>
</tr>
<tr>
<td>Voltage gain ((K_v))</td>
<td>2.5</td>
</tr>
<tr>
<td>Motor torque constant</td>
<td>9.0 Ncm/Amp</td>
</tr>
<tr>
<td>Motor amplifier gain</td>
<td>2.5</td>
</tr>
<tr>
<td>No load current</td>
<td>65 mA</td>
</tr>
<tr>
<td>No load speed</td>
<td>2300 rpm</td>
</tr>
</tbody>
</table>

#### Table 1: Parameters of the CE110 Motor.
shows the measured parameters via GA approach, it can be notified that the values of gain constant (k) are quite similar during the three load cases. Also, the three performance criteria used along with the GA algorithm provide similar gain results. Moreover, it can also be notified that there is a proportional relationship between the value of the two time constants (T1,T2) and the size of inertial load, the increment in the size of inertial load led to an increment in the values of the dominator of the model (T1*T2 and T1+T2). The performance of the three fitness functions was quite close in the three load cases, except for the performance of ISE in the third case (High inertia load), the model parameters via ISE are quite different from those calculated by IAE and ITAE under the same loading condition. The justification behind this odd result that the ISE criterion failed to provide accurate identification for the model parameters at maximum inertial loads. The experiment has been iterated five times when the ISE fitness function is applied at the high load condition, the results showed that there was no correlation between the determined model parameters via ISE (T1*T2 and T1+T2) during the five iterations.

In order to evaluate the performance of the GA algorithm with ISE, IAE and ITAE criteria through the identification stage, time domain responses of both modeled and real servo are measured, the responses of the real system under varying inertial loads can be illustrated in Figure 6. Table 3 shows that the results obtained by the optimized model are close to those measured from the real system, the correlation percentage was quite high and indicates to the identity between the time domain performance for the model and the real servo motor. Moreover, it can be observed from the table that when the ISE was applied as a fitness function for GA in the three loading cases, the correlation coefficient was nearly constant, hence it’s performance is more consistent than the other types in coping with load variations.

It can also be observed from the obtained outcomes that each criteria has some individual advantages. For instance, ITAE was the best criteria in minimizing of rise time when small loads are applied, while ISE and IAE were the best for medium and high load cases, respectively. Moreover, the smallest settling time was obtained by ITAE in the minimum load case, while the ISE and ITAE provide better results in the medium and large loads. It can also be notified that there is a proportional relationship between the increment in the size of inertial load and the increment in the values of rise time and settling time. In the next section, the GA with the three error criteria will be used to optimize the PID parameters in order to obtain a stable speed for the servo motor and also to keep time domain specifications such as settling time within the desired range (as shown in Table 4).

<table>
<thead>
<tr>
<th>Size of inertial load</th>
<th>Type of Fitness function</th>
<th>The obtained model parameters</th>
<th>Fitness of solution</th>
<th>Performance index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>ISE</td>
<td>0.9757, 0.1724, 0.3217</td>
<td>3.0914, 0.0033</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>0.9769, 0.1831, 0.3090</td>
<td>2.6325, 0.0041</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>0.9539, 0.2253, 0.2500</td>
<td>0.2883, 0.0444</td>
<td></td>
</tr>
<tr>
<td>Medium</td>
<td>ISE</td>
<td>0.9673, 0.1829, 0.8093</td>
<td>2.3761, 0.0046</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>0.9662, 0.1882, 0.7979</td>
<td>2.1483, 0.0052</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>0.9664, 0.3234, 0.6345</td>
<td>0.7607, 0.0223</td>
<td></td>
</tr>
<tr>
<td>Large</td>
<td>ISE</td>
<td>0.9625, 0.5725, 0.8388</td>
<td>0.3903, 0.0442</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>0.9626, 0.2065, 1.2908</td>
<td>1.4981, 0.0081</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>0.963, 0.2082, 1.289</td>
<td>1.5105, 0.0081</td>
<td></td>
</tr>
</tbody>
</table>
The function provides the fastest model which converges to the fitness value of 1.315 after 84 generations. The experimental outcomes show that the model converges to the fitness function value of 0.4655 after 89 generations when the inertial load is applied. The results clearly indicate that with the IAE criteria, the fastest model was obtained.

In the case of a medium inertial load is applied, the optimization model converges to fitness value 0.6621 after 29 generations. Moreover, it can be observed from Figure 7 that the ITAE provides the fastest optimization model that converges to the fitness value compared with the other two criteria. The highest fitness value was 2.563 and obtained by the ISE.

In general, the ISE and IAE provide similar performance when small loads are used, they both require the same number of generations to reach the fastest optimal solution. Moreover, the ISE again provides the fastest model when the inertial loads are increased from small to medium. Surprisingly, the ITAE successfully provides the fastest optimization model at the maximum inertial load, it only required for 29 generations to reach the final fitness value, this number of generations is almost half of the number required for the ISE and about a third for the ITAE in order to obtain the fitness value. It can be observed from the experimental results in the three load cases, that the fastest optimization model doesn’t converge to the highest fitness value.

When the high inertia load is applied, the optimization model with ISE criteria converges to fitness function value of 2.562, from generation 77. In case of IAE, the model converges to value, 0.6792, after generation 49. The model with ITAE converges to fitness value 0.6621 after 29 generations. Moreover, it can be observed from Figure 7 that the ITAE provides the fastest optimization model that converges to the fitness value compared with the other two criteria. The highest fitness value was 2.563 and obtained by the ISE.

As Figure 7 shows, in the case of a small inertia load is applied, and the ISE performance criteria was defined as a GA fitness function, the optimization model converges to fitness function value of 0.3325, from generation 36. In case the IAE criteria is used, the model converges to value, 0.3799, after generation 36. The model converges to fitness value of 3.728, after generation 47, when the ITAE is applied. The results clearly indicate that with the use of the IAE criteria, the fastest model was obtained. However, both of ISE and IAE reached their final fitness value at the same number of generations which is 36. However, the highest fitness value was produced when the ITAE is applied.

In the case of a medium inertial load is applied, the model converges to fitness value 0.4209, after generation 70, when ISE criteria is used, and the model converges to fitness value 0.4655 after 89 generations when the function is IAE. In case the ITAE is used, the model converges to the fitness function value of 1.315 after 84 generations. The experimental outcomes show that ISE provides the fastest model which converges to the fitness value before the IAE and ITAE criteria. However, the highest fitness value was again obtained by the ITAE function.
B. Control optimization

In this part, the GA technique will be used for determining the PID controller elements that yield a desired output response and meet the time domain performance criteria that was previously defined in Table 4, this criteria includes the peak overshoot, rise time and settling time. The obtained results from the previous identification process were used to implement the servo model with GA PI controller (as illustrated in Figure 8). The GA optimization was simulated via Matlab optimization toolbox which the fitness function was first defined and created prior its insertion into the optimization tool. Also, the parameters of the unknown output of the fitness function were defined as $K_c$, $K_i$ and $K_d$.

In order to reduce simulation time, both the size of population and selected bounds have to be kept small, but the downside will be low accurate outcome from the GA. Therefore, the balance between the accuracy and time cost is very important and was taken into our consideration during this work. The search range for the gain values of the PID controller will be restricted in order to minimize the computational cost of the GA optimization process. $k_p$ values are specified in the range of 1 to 20, $k_i$ in the range of 1 to 10 and $k_d$ in the range of 0 to 5. It is also worth mentioning that the range of requirement can be increased by widening the number of generations which will lead to an increment in the number of generations required for converging to optimal value. In this work, the number of generations is 100 and the population size is 20, and initial bound values are specified in the range [0.1, 10; 0.1, 10] respectively.

The performance of GA technique for tuning PI controllers can be illustrated in Table 5 and Figure 9, the table shows that there is no derivative gain values calculated by the GA algorithm despite of its existence within the structure of the controller in the simulation, this means that the PI controller is more appropriate than PID controllers to achieve the required performance. Table 5 shows that the GA based PI controller with the ITAE has provided superior performance compared with the other criteria in terms of all demanded specifications (rise time, settling time and percentage of overshoot), it also produces the best performance in all load cases. Moreover, the parameters of the PI controller were almost the same despite of the variations in the inertial load. Moreover, the GA based PI controller with the ISE fitness function have the slowest settling time compared with other functions in the three loading cases. Also, the percentage of overshoot was significantly increased along with increment in the loads, the controller with ISE function produces satisfactory results in terms of rise time. On the other hand, the PI controller with IAE produce acceptable results when the applied loads are small and medium, but its performance deteriorated at the maximum inertial load, it produces slow settling time and the highest percentage of overshoot compared with other fitness functions. Table 5 also shows that the optimization time for GA with three fitness functions was quite the same (around 34 sec), it is important to be reduced in the case of using the GA technique in applications that require on-line optimization of control parameters.

![Simulink model of Servo Motor with PID parameters](image)

Table 5: GA Performance.

<table>
<thead>
<tr>
<th>Size of inertial load</th>
<th>GA Criteria</th>
<th>PI Parameter</th>
<th>Performance of controller</th>
<th>Overall optimization time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$K_p$</td>
<td>$K_i$</td>
<td>$t_r$ (s)</td>
</tr>
<tr>
<td>Small</td>
<td>ISE</td>
<td>1.0</td>
<td>1.0</td>
<td>1.8</td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>4.4</td>
<td>2.1</td>
<td>1.8</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>4.3</td>
<td>2.0</td>
<td>1.9</td>
</tr>
<tr>
<td>Medium</td>
<td>ISE</td>
<td>3.9</td>
<td>2.4</td>
<td>1.8</td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>4.4</td>
<td>2.1</td>
<td>1.8</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>4.3</td>
<td>2.0</td>
<td>1.9</td>
</tr>
<tr>
<td>Large</td>
<td>ISE</td>
<td>3.9</td>
<td>2.4</td>
<td>1.8</td>
</tr>
<tr>
<td></td>
<td>IAE</td>
<td>5.8</td>
<td>2.5</td>
<td>1.3</td>
</tr>
<tr>
<td></td>
<td>ITAE</td>
<td>2.0</td>
<td>1.9</td>
<td>2.5</td>
</tr>
</tbody>
</table>
introduced the best performance at high loads. The experimental results also showed that during the PI tuning process, the GA with ITAE provided the best performance in terms of meeting all required time domain specifications and in finding good gain values for the PI controller compared with other functions, the simulation results confirmed that the GA with ITAE produced the best outcomes under varying inner loads. For future work, due to the high computational cost for tuning PI controller with GA implemented on and commanded from a host PC, the GA algorithm will be further implemented in a separate hardware card (a smart FPGA board) in order to accelerate the optimization process and also to evaluate its performance in real-time applications.

![Figure 9. Time Response of the Servo Speed with ISE, IAE and ITAE Approaches.](image)

**V. CONCLUSION**

In this paper, a design of PI controller via Genetic algorithm (GA) was presented, the GA was used with three common fitness functions for identifying of model parameters and for finding the optimal PI values that ensure a stable speed for the DC motor. The DC motor, namely CE110 was mathematically modeled prior to the parametric identification process, the GA was used with three fitness functions which are ISE, IAE, and ITAE. The results showed that the GA with both ISE and IAE provide similar performance during the identification process, they both introduced the fastest optimization solutions compared to the ITAE when the inner loads are small and medium. However, the GA with ITAE

**REFERENCES**


Hamza Alzarok and Ayoub H. Mushalh/Tuning of a Speed Control System for DC Servo Motor Using Genetic Algorithm


