PROPORTIONAL-INTEGRAL (PID) CONTROLLER DESIGN USING GENETIC ALGORITHM (GA)

Asst. Lect. Emad A. Hussein & Lect. Musa H. Waly
Electrical Engineering Department
Al-Mustansiriya University
Iraq-Baghdad

ABSTRACT

The selection of the three coefficients of proportional-integral (PID) controllers (Ki, Kp, and Kd) is basically a search problem in a three-dimensional space. This is so because points in the search space correspond to different selections of a PID controller's three parameters. By choosing different points of parameter space, we can produce, for example, different step responses for a step input. A PID controller can be determined by moving in this search space on trial-and-error basis. The main problem in the selection of the three coefficients is that these coefficients do not readily translate into the desired performance and robustness characteristics that the control system designer has in mind. Several rules and methods using root locus and performance indices. The first design uses the Integral of Time multiplied by Absolute Error (ITAE) performance index. Hence we select the three PID coefficients (Ki, Kp, and Kd) to minimize the ITAE performance index, which produces a good transient response to a step input.

Our paper uses the Genetic Algorithm (GA). In this method the selection of the three PID coefficients depends on the minimization of the Mean Squared Error (MSE), which will produce an excellent transient response to a step input.

Key words: PID Controller, ITAE, and Genetic Algorithm GA.

تصميم مسيطر تناسبي _ تكاملي (ي أي دي) باستخدام خوارزمية جينية

م.م موسى هادي والي م.م. عماد احمد حسين
الجامعه المستنصريه/ قسم الهندسه الكهربائيه

الخلاصة

إنّ إختيارَ **المعاملات الثلاثة** (Ki, Kp, Kd) الخاصة بالمسيطرات التفاضلية-التكاملية (PID) **أساساً** هي **مشكلة بحثٍ في** فضاء **ثلاثي الأبعاد**. وهذه المشكلة هي بسبب **تُقابلُ النقاطُ في فضاءِ البحثَ** لاختيارات **مختلفة** لمعاملات المسيطر الثلاثَ. بإخْتيار نقاط **مختلفة مِنْ فضاءِ المعاملاتِ** يُمْكِنُ أنْ نُنتجَ، على سبيل المثال، **لاستجابة نظام لمدخل دالة** خطوية. يمكن للمسيطر التفاضلي-**التكاملي** أنْ يُقرّرَ بالإنتقال في فضاءِ البحثِ هذا **بالاعتماد على قاعدةِ التجربة والخطأ**. إنّ المشكلة الرئيسية في إختيارِ المعاملاتِ الثلاثة بأنّ **المعاملاتِ لا تُترجمُ** الأداءِ المطلوبِ بسهولة وكذلك فهي لا تعبرعن منظومة السيطرة التي هي في عقل المصمم. **عِدّة قواعد وطرق اَستعملت** اعتمدت في عملها على نظريتين (root locus و Performance indices). اَستعملُ التصميمُ الأولُ **معامل مكاملة مضاعفة زمن الخطأ المطلق** (ITAE). لذا فبحسب هذه النظرية سيتم اختيار المعاملات الثلاث لتقليل معامل (ITAE) **مما** يعطي استجابة جيدة لمدخل الدالة الخطوية.

يَستعملُ تصميمُنا الخوارزمية الوراثية (GA). في هذه الطريقةِ، يَعتمدُ إختيارُ معاملاتِ المسيطر الثلاث على تحقيق حدّ أدنى **لمعدل مربع الخطأ** (MSE) **مما** سَيُنتجُ عنه استجابة مثلى لمدخل الدالة الخطوية.

1. DESIGN OF CONTROLLERS (A PREVIEW)

Many industrial processes are controlled using proportional-integral controllers. the popularity of PID controllers can be attributed arty to their forename in a wide rang of operating conditions and partly to their functional which allows engineers to operate them in simple , straightforward manner . To implement such a controller, three parameters must be determined for the given process: proportional gain, integral gain, and derivative gain.

The feedback control system shown in **Fig.(1)** will be our tested system that the laplace term (Gp(s)) represent the plant transfer function, (Gc(s)) the controller transfer function and (H(s)) the feedback transfer function. The input and output laplace term will be R(s) and Y(s) respectively. So we can classify the controllers to four types [1][2]:

1.1- Proportional Term Controllers (P-Controllers)

The P-controller is a pure gain (no dynamics) of value Kp (i.e. Gc(s) = Kp) thus we see that the gain Kp that we have been varying to generate the desired response, that is , the system characteristic equation is given by:

$$1+K_pG_p(s)H(s)=0$$

This controller is used in situations in which satisfactory transient and steady–state responses can be obtained simply by setting a gain in the system, with no dynamic compensation required.

1.2- Proportional plus Integral term Controllers (PI-Controllers):

This controller increases the system by 1 and is used to improve the steady–state response. The transfer function of the PI controller can be expressed as

$$G_c(s) = k_p + \frac{K_i}{s} \tag{1}$$

The controller has a pole at the origin and at zero at Ki /Kp . since the pole is nearer to the origin than is the zero , the controller is phase – lag , and the controller adds a negative angle to the criterion of the root locus ,Hence this controller is used to improve the steady-state response of the system , as stated earlier . The open–loop function can be expressed as

$$G_c(s)Gp(s)H(s) = (k_p + \frac{k_i}{s})G_p(s)H(s) \tag{2}$$

And we see that are only two independent parameters to be determined in the design process. We can arbitrarily set K=1 without affecting the generality of the design, the problem is then to determine Kp and Ki to meet certain steady-state design criteria.

1.3- Proportional plus Derivative term Controllers (PD-Controllers):

The transfer function of the PD controller is

$$G_c(s) = K_p + K_d s = K_d(s + \frac{K_p}{K_d}) \tag{3}$$

Thus the PD controller introduces a single zero at s = - Kp / Kd , and it is seen that this controller adds a positive angle to the criterion of the root locus, therefore , the PD controller is a type of phase –lead controller and improves the system transient response . Since only a single zero is introduced. The open–loop function can be expressed as:

$$G_c(s)Gp(s)H(s) = (K_p + K_d s)G_p(s)H(s) \tag{4}$$

1.4- Proportional plus Integral plus Derivative term Controllers (PID-Controllers)

The design of proportional–plus–integral-plus-derivative (PID) controller is introduced in this section. The PID controller is probably the most commonly used controller in feedback control system. With r(t) the controller input and y(t) the output (according to **Fig.(1)**) , the PID controller is defined as :

$$Y(s) = (K_p + \frac{K_i}{s} + K_d s)\,R(s) \tag{5}$$

or

$$G_c(s) = \frac{Y(s)}{R(s)} = K_p + \frac{K_i}{s} + K_d s$$

$$y(t) = K_p r(t) + K_i \int r(t)dt + K_d \frac{dr(t)}{dt}$$

A block diagram representation of this controller is given in **Fig. (2a)**, and the transfer function are shown in **Fig. (2b)** .Quite often it is not necessary to implement all three terms to meet the design specification for a particular control system.

2- PERFORMANCE OF A SECOND – ORDER SYSTEM:

Let us consider a single – loop second –order system and determine its response step input a closed – loop feedback control system is shown in **Fig. (2)** The output is:

$$Y(s) = \frac{G(s)}{1+G(s)}R(s) = \frac{K}{s^2 + ps + K}R(s) \tag{6}$$

Utilizing the generalized, we may rewrite this equation as:

$$Y(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{7}$$

With a unit step input, we obtain

$$Y((s) = \frac{\omega_n}{s(s \ + 2\xi\omega_n + \omega_n)} \tag{8}$$

For which the transient output, as obtained from the laplace transform:

$$y(t) = 1 - \frac{1}{\beta} e^{-\xi\omega_n S} \sin(\omega_n \beta t + \theta) \tag{9}$$

Where $\beta = \sqrt{1-\xi^2}, \theta = \cos^{-1}\xi,\ and \quad 0 < \xi < 1$

Then transient response of second order system for various values of the damping ratio (ζ) is shown in **Fig (3a)**, the closed-loop roots approach the imaginary axis, and the response becomes oscillatory. The response as a function of time is also shown in **Fig (3b)** for step input.

The laplace transform of the unit impulse, where R(s) =1, implies

$$Y(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{10}$$

Which is T(s) = y(s) /R(s), the transfer function of the closed-loop system. The response for an impulse function input is then

$$y(t) = \frac{\omega_n}{\beta} e^{-\xi\omega_n S} \sin \omega_n \beta t \tag{11}$$

Which is simply the derivative of the response to a step input. The impulse response second–order system for several values of the damping ratio. The designer is able to select several alternative performance measures from the response of the system for either a step or impulse input [1][3][4].

Standard performance measures are usually defined in terms of the step response system as shown in **Fig. (4)**. The swiftness of the response is measured by the rise time T_r and the peak time T_p. For under damped system with an overshoot, the 0-100% rise time is a useful index. If the system is over damped, then the peak time is not defined, and the 10 – 90 % rise time, T_{r1} is normally used. The similarity with which the actual response matches the step input is measured by the percent overshoot and settling time T_s. The percent overshoot, ***PO.***, is define as:

$$P.O. = \frac{M_{pt} - fv}{fv} * 100\% \tag{12}$$

for a unit step input , where M_{pt} is the peak value of the time response , and f_v is value of the response normally f_v is the magnitude of the input , but many system have a final value significantly different from the desired input magnitude , for the system with a unit step represented by Eq (12) ,we have f_v = 1.

The setting time, T_s, is defined as the time required for the system to settle within a certain percentage σ of the input amplitude, this band of ± σ is shown in **Fig (4)** for the second order system with closed-loop damping constant $\zeta\omega_n$, with a response described by Equ.(11) we seek to determine the time, T_s, for which the response remains within 2 % of the final value. This occurs approximately when

$$e^{-\xi\omega_n Ts} < 0.02$$

or

$$\xi\omega_n Ts \cong 4$$

Therefore we have

$$T_S = 4\tau = \frac{4}{\xi\omega_n} \tag{13}$$

Hence we will define the setting time as four time constants (that is , $\tau = 1/\zeta\omega_n$) of the dominant roots of the characteristic equation . The steady–state error of the system may be measured on the step response of the system as shown in **Fig (4)**. The transient response of the system may be described in terms of two factors:

1- The swiftness of response, as represented by the rise time and the peak time.

2- The closeness of the response to the desired response, as represented by the overshoot and settling time.

As nature would have it, these are contradictory requirements, and a compromise must be obtained. The explicit relation for M_{pt} and T_p as a function of ζ is:

$$M_{Pt} = 1 + e^{-\xi\pi/\sqrt{1-\xi^2}}$$

Therefore the percent overshoot is

$$PO. = 100e^{-\xi\pi/\sqrt{1-\xi^2}}$$

or

$$\zeta = \frac{Ln(100/PO.)}{\sqrt{\pi^2 + (Ln(100/PO.))^2}} \tag{14}$$

3- PERFORMANCE INDICES:

Increasing emphasis on the mathematical formulation and measurement of control performance can be found in the recent references on automatic control. Modern theory assumes that systems engineer can specify quantitatively the required system performance, then a performance index can be calculated or measured and used to the system's performance.

Whether the aim is to improve the design of a system or to design a control system performance index must be chosen and measured. A system is considered an optimum control system when the system parameter adjusted so that the index reaches an extremism value, commonly a minimum value performance index, to be useful, must be a number that is always positive or zero, then best system is defined as the system that minimizes this index.

A suitable performance index is the integral of the square of the error, ISE, which is defined as:

$$ISE = \int_0^T e^2(t)dt \tag{15}$$

The upper limit T is a finite time chosen somewhat arbitrarily so that the integral approaches a steady – state value. It is usually convenient to choose T as the setting time, Ts.

Another readily instrumented performance criterion is the integral of the knitted of the error IAE which is written as

$$IAE = \int_0^T |e(t)|dt \tag{16}$$

This index is particularly useful for computer studies.

To reduce the contribution of the large initial error to the value of the integral as well as to emphasize errors occurring later in the response, the following index has been proposed:

$$ITAE = \int_0^T t\left|e(t)\right|dt \tag{17}$$

The performance index ITAE provides the best selectivity of the performance index that is, the minimum value of the integral is readily discernible as the system parameters are varied.
Table(1) illustrates the optimum coefficients of the T(s) based on the ITAE criterion for a step input [2][5].

4. GENETIC ALGORITHM (GA) METHODOLOGY

A genetic algorithm (GA) is a search technique used in computing to find exact or approximate solutions to optimization and search problems. Genetic algorithms are categorized as global search heuristics. Genetic algorithms are a particular class of evolutionary algorithms (also known as evolutionary computation) that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover (also called recombination).
Genetic algorithms are implemented as a computer simulation in which a population of abstract representations (called chromosomes or the genotype of the genome) of candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem evolves toward better solutions. Traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (recombined and possibly randomly mutated) to form a new population. The new population is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population. If the algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached.
Genetic algorithms find application in bioinformatics, phylogenetics, computational science, engineering, economics, chemistry, manufacturing, mathematics, physics and other fields.
A typical genetic algorithm requires two things to be defined:
1- a genetic representation of the solution domain,
2- a fitness function to evaluate the solution domain.
A standard representation of the solution is as an array of bits. Arrays of other types and structures can be used in essentially the same way. The main property that makes these genetic representations convenient is that their parts are easily aligned due to their fixed size that facilitates simple crossover operation. Variable length representations may also be used, but crossover implementation is more complex in this case. Tree-like representations are explored in Genetic programming and graph-form representations are explored in Evolutionary programming.

The fitness function is defined over the genetic representation and measures the quality of the represented solution. The fitness function is always problem dependent. For instance, in the knapsack problem we want to maximize the total value of objects that we can put in a knapsack of some fixed capacity. A representation of a solution might be an array of bits, where each bit represents a different object, and the value of the bit (0 or 1) represents whether or not the object is in the knapsack. Not every such representation is valid, as the size of objects may exceed the capacity of the knapsack. The fitness of the solution is the sum of values of all objects in the knapsack if the representation is valid, or 0 otherwise. In some problems, it is hard or even impossible to define the fitness expression; in these cases, interactive genetic algorithms are used.

Once we have the genetic representation and the fitness function defined, GA proceeds to initialize a population of solutions randomly, then improve it through repetitive application of mutation, crossover, inversion and selection operators [6][7][8].

4.1- Initialization

Initially many individual solutions are randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Traditionally, the population is generated randomly, covering the entire range of possible solutions (the search space). Occasionally, the solutions may be "seeded" in areas where optimal solutions are likely to be found.

4.2-Selection

During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Certain selection methods rate the fitness of each solution and preferentially select the best solutions. Other methods rate only a random sample of the population, as this process may be very time-consuming.

Most functions are stochastic and designed so that a small proportion of less fit solutions are selected. This helps keep the diversity of the population large, preventing premature convergence on poor solutions. Popular and well-studied selection methods include roulette wheel selection and tournament selection.

4.3- Reproduction

The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover (also called recombination), and/or mutation.

For each new solution to be produced, a pair of "parent" solutions is selected for breeding from the pool selected previously. By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents". New parents are selected for each child, and the process continues until a new population of solutions of appropriate size is generated.

These processes ultimately result in the next generation population of chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best organisms from the first generation are selected for breeding, along with a small proportion of less fit solutions, for reasons already mentioned above.

4.4-Termination

This generational process is repeated until a termination condition has been reached. Common terminating conditions are:

- A solution is found that satisfies minimum criteria
- Fixed number of generations reached
- Allocated budget (computation time/money) reached
- The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results
- Manual inspection
- Combinations of the above

5-SIMPLE GENERATIONAL GENETIC ALGORITHM PSEUDOCODE

1. Choose initial population
2. Evaluate the fitness of each individual in the population
3. Repeat until termination: (time limit or sufficient fitness achieved)

i Select best-ranking individuals to reproduce ii Breed new generation through crossover and/or mutation (genetic operations) and give birth to offspring

iii Evaluate the individual fitnesses of the offspring

iv Replace worst ranked part of population with offspring

6- PROBLEM DOMAINS

Whenever the ITAE performance index gets stuck at a local minimum or the convergence rate is relatively slow depending on the damping ratio value (ζ), we start the genetic search by finding the optimal PID coefficient (***K***K_i, K_p, and K_d) with the best fitness i.e. the smallest (MSE) as the survivor. In other words, the PID coefficient will be the GA chromosomes in which a fitness value is assigned to each chromosome. The fitness value for the j-th chromosome is inversely proportional to the mean squared error $\overline{\varepsilon_j^2}$, where ($\overline{\varepsilon_j^2}$) is given by

$$\overline{\varepsilon_j^2} = \frac{1}{R}\sum_{k=1}^{R}[d_j(k) - y_j(k)]^2 \qquad (18)$$

Where R is the window size over which the errors will be accumulated; d(k) is the original response; y(k) is the estimated output associated with the j-th estimated chromosome. The survived chromosome will be the optimized PID coefficient that gets us a minimum MSE (or optimal solution).

7- EXAMPLES AND SIMULATION RESULTS

In this section, The feedback control system shown in **Fig.(5)** will be tested with different plants represented by the plant transfer function (G_p). The optimum controller coefficients (K_i, K_p, and K_d) for the controller transfer function (G_c) will be obtained using ITAE criteria and Genetic algorithm (GA). These optimum coefficients will obtained according to a step input and a required settling time (T_s) and percentage overshoot (***PO.***).

Where:

$$G_c = \frac{K_d s^2 + K_p s + K_i}{s} \qquad (19)$$

and the closed-loop transfer function:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{G_c G_p(s)}{1 + G_c G_p(s)} \qquad (20)$$

7.1- First Example: Robust control of temperature

Consider a temperature controller with a plant transfer function

$$G_p(s) = \frac{1}{(s+1)^2} \qquad (21)$$

We desire to obtain an optimum controller coefficients (K_i, K_p, and K_d) using ITAE performance and Genetic algorithm (GA) for a step input, settling time (T_s)of (**0.6**) and a percentage overshoot (***PO.***) of (**32%**).

7.1.1- ITAE Optimization criteria

Using the plant and controller transfer functions ((G_p) and (G_c) represented by equ.(21) and equ.(19) respectively) we have the closed-loop transfer function according to equ.(20) :

$$T(s) = \frac{Y(s)}{R(s)} = \frac{K_d s^2 + K_p s + K_i}{s^3 + (2 + K_d)s^2 + (1 + K_p)s + K_i} \quad (22)$$

The optimum coefficients of the characteristic equation for ITAE are obtained from table (1) as

$$(s^3 + 1.75\omega_n s^2 + 2.15\omega_n^2 s + \omega_n^3) \quad (23)$$

We need to select ω_n in order to meet the settling time and percentage overshoot requirement. Since

$$\zeta = \frac{Ln(100/PO.)}{\sqrt{\pi^2 + (Ln(100/PO.))^2}} \quad (24)$$

and

$$\omega_n = \frac{4}{\zeta T_s} \quad (25)$$

so we obtain ζ = 0.34 and ω_n = 19.552. The value of ω_n will be substituted in equ.(23) to get the characteristic equation:

$$(s^3 + 34.216s^2 + 812.903s + 7474.352) \quad (26)$$

By matching equ.(26) with the characteristic equation (denominator) of equ.(22) we obtain

2 + K_d = 34.216,

1 + K_p = 812.903, and

K_i = 7474.352

So: K_d = 32.216, K_p = 811.903, and K_i = 7474.352

The overall transfer function according to ITAE criteria will be:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{32.216s^2 + 811.9s + 7474.352}{s^3 + 34.216s^2 + 812.9s + 7474.352} \quad (27)$$

Fig.(6) shows the original step response and **Fig.(7)** shows the ITAE step response according to the transfer function obtained in equ.(27). The Mean Squared Error (MSE) according to ITAE optimization equal to: 0.1005

7.1.2- GA Optimization method:

The Mean Squared Error (MSE) between the original step response and the obtained one, dedicated previously in equ.(18), is selected to be minimized by the use of GA in order to obtain optimum PID coefficients that give minimum MSE. The specifications that were used in GA tuning are:

Number of generation =100.
Population size =20.
Crossover probability =0.8 (Simple crossover).
Mutation probability =0.05 (Uniform mutation).
Fig.(8) shows the GA step response. The obtained PID coefficients according to GA optimization method are: K_d = 32.0283, K_p = 824.6774, and K_i = 7479.1691
That gives a minimum mean squared error equal to: 0.0347
The overall transfer function due to GA optimization is:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{32.0283s^2 + 824.6774s + 7479.1691}{s^3 + 34.02683s^2 + 825.6774s + 7479.1691} \quad (28)$$

7.2- SECOND EXAMPLE: CASSETTE TAPE STORAGE DEVICE [2][3]

A cassette tape storage device has been designed for mass-storage. It is necessary to control accurately the velocity of the tape. The speed control of the tape drive is represented by the system shown in **Fig. (5)** and its (G_p) is:

$$G_p(s) = \frac{10}{s^2 + 40s + 400} \quad (29)$$

The desired settling time (T_s) is (**0.8**) and a percentage overshoot (***PO.***) is (**30%**).

7.2.1- ITAE Optimization criteria:

Using the plant and controller transfer functions ((G_p) and (G_c) represented by equ.(29) and equ.(18) respectively) we have the closed-loop transfer function according to equ.(19) :

$$T(s) = \frac{Y(s)}{R(s)} = \frac{10K_d s^2 + 10K_p s + 10K_i}{s^3 + (40 + 10K_d)s^2 + (400 + 10K_p)s + 10K_i} \quad (30)$$

Repeating steps modeled in equ.(22), (23) and equ.(24) we fined that ζ = 0.358 and ω_n = 13.97. The value of ω_n will be substituted in equ.(22) to get the characteristic equation:

$$(s^3 + 24.447s^2 + 419.595s + 2726.397) \quad (31)$$

By matching equ.(30) with the characteristic equation (denominator) of equ.(31) we obtain:
40 + 10K_d = 24.447,
400 + 10K_p = 419.595, and
10K_i = 2726.397
So:
K_d = -1.5553, K_p = 1.972, and K_i = 272.63
The overall transfer function according to ITAE criteria will be:

$$T(s) = \frac{Y(s)}{R(s)} = \frac{-15.553s^2 + 19.72s + 2726.397}{s^3 + 24.447s^2 + 419.595s + 2726.397} \quad (32)$$

Fig.(9) shows the original step response and Fig.(10) shows the ITAE step response according to the transfer function obtained in equ.(32). The Mean Squared Error (MSE) according to ITAE optimization equal to: 0.3497

7.2.2- GA Optimization method

The specifications that were used in GA tuning are:
Number of generation =100.
Population size =20.
Crossover probability =0.8 (Simple crossover).
Mutation probability =0.05 (Uniform mutation).
Fig.(11) shows the GA step response. The obtained PID coefficients according to GA optimization method are: K_d = -1.0166, K_p = 1.8710, and K_i = 273.8123
That gives a minimum mean squared error equal to: 0.0849
The overall transfer function due to GA optimization is:

$$T(s)=\frac{Y(s)}{R(s)}=\frac{-10.1662s^2+18.707s+2738.1232}{s^3+29.8338s^2+418.707s+2738.1232} \quad (33)$$

8- COMPARISON AND CONCLUSIONS

Fig.(12) and Fig.(13) shows a comparison for the obtained step response (original, ITAE and GA response) in the same figure for the first and second example respectively. One can see from Fig.(12) that there is a similarity between the original response (the sold line) and both the ITAE response (the dash-dot line) and the GA response (the dashed line).
Also we can observe that there is a great convergence between the original response (the sold line) and GA response (the dashed line) rather than the ITAE response (the dash-dot line).
Finally we can conclude from this comparison that the GA optimization method is suitable in some type of plants but in other types the ITAE optimization criteria may be more effective than the GA method.
Generally GA optimization method is a very effective, flexible, tenuous and suitable for many complex plants rather than other methods.

REFERENCES

[1] R. C. Dorf, R. H. Bishop, “Modern Control Systems” Addison-Wesley Longman, Inc, (1998).

[2] Chi-Tsong Chen, “Analog & Digital Control System Design” State University of New York at stony Brook 2005.

[3] Kilian, “Modern Control Technology, Components and systems” 2nd Edition 2006.

[4] “Control System Toolbox for Use With MATLAB”, User’s Guide, Mathworks, Inc., Version 4.2, 4th printing, 1999.

[5] L. C. Jain & R. K. Jain “Hybrid Intelligent Engineering systems”, World Scientific Publishing, Co. Pte. Ltd., 1997.

[6] K. –F. Man, K.-S.Tang, S. Kwong & W. A. Halang., “ Genetic Algorithm: Concepts and Design” City Univrsity of Hong Kong, Tachee Aveneue, Kowloon, maito: email: %20k. nan @ ieee. Org. (2000).

[7] D. E. Goldberg, “Genetic Algorithms in search, optimization, and Machine Learning,” Addison-Wesley Publishing Company, Inc, U.S. (1989).

[8] Coley, D. A. "An Introduction to Genetic Algorithms for Scientists and Engineers". World Scientific Publishing, 1999.

Table (1): The optimum coefficients of T(s) based on the ITAE criterion for a step input.

$$s+\omega_n$$
$$s^2+1.4\omega_n s+\omega_n^2$$
$$s^3+1.75\omega_n s^2+2.15\omega_n^2 s+\omega_n^3$$
$$s^4+2.1\omega_n s^3+3.4\omega_n^2 s^2+2.7\omega_n^3 s+\omega_n^4$$
$$s^5+2.8\omega_n s^4+5.0\omega_n^2 s^3+5.5\omega_n^3 s^2+3.4\omega_n^4 s+\omega_n^5$$
$$s^6+3.25\omega_n s^5+6.60\omega_n^2 s^4+8.60\omega_n^3 s^3+7.45\omega_n^4 s^2+3.95\omega_n^5 s+\omega_n^6$$

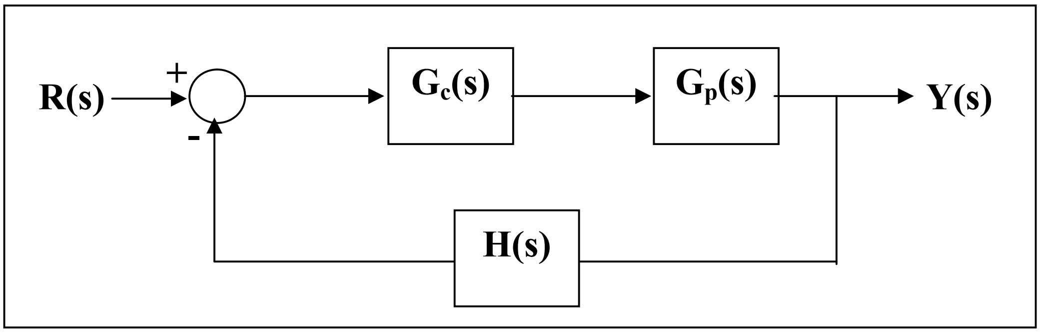

Fig.(1): Feedback control system with a controller, plant and feedback transfer function.

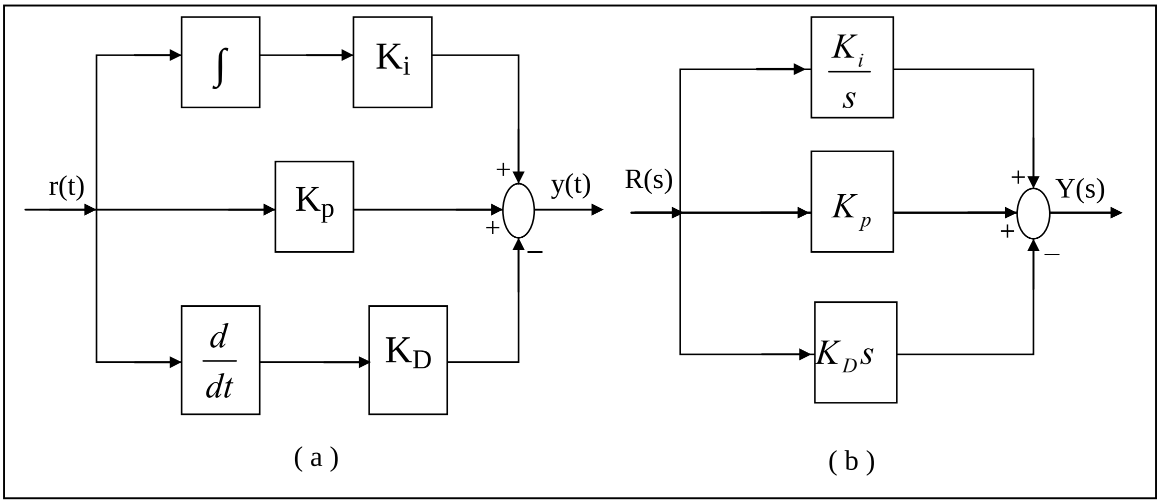

Fig.(2): PID Controller

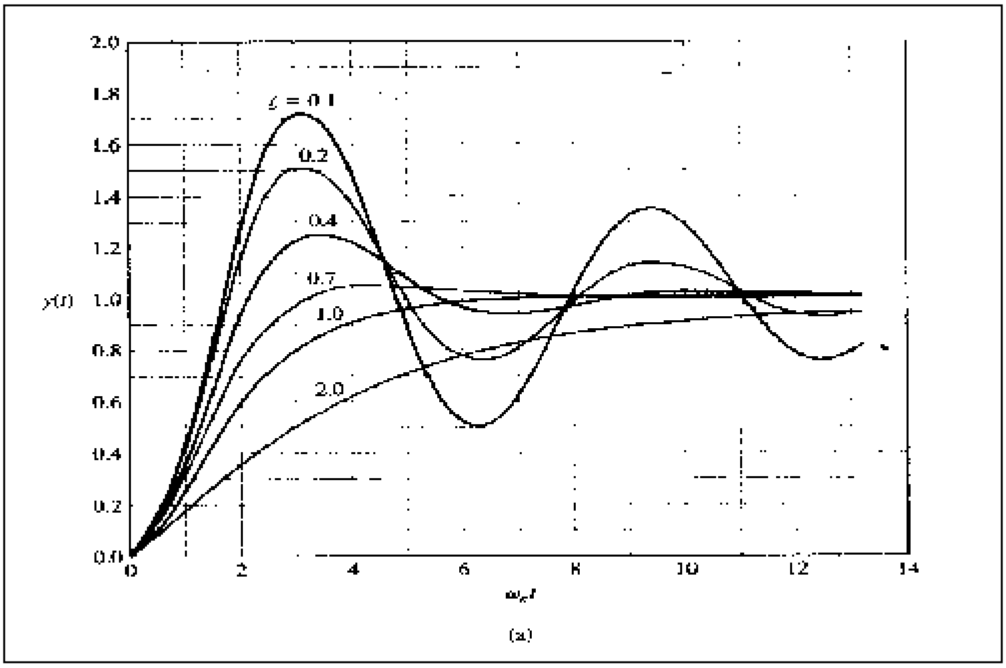

Fig. (3a): Transient response of a second order system for various values of the damping ratio (ζ)

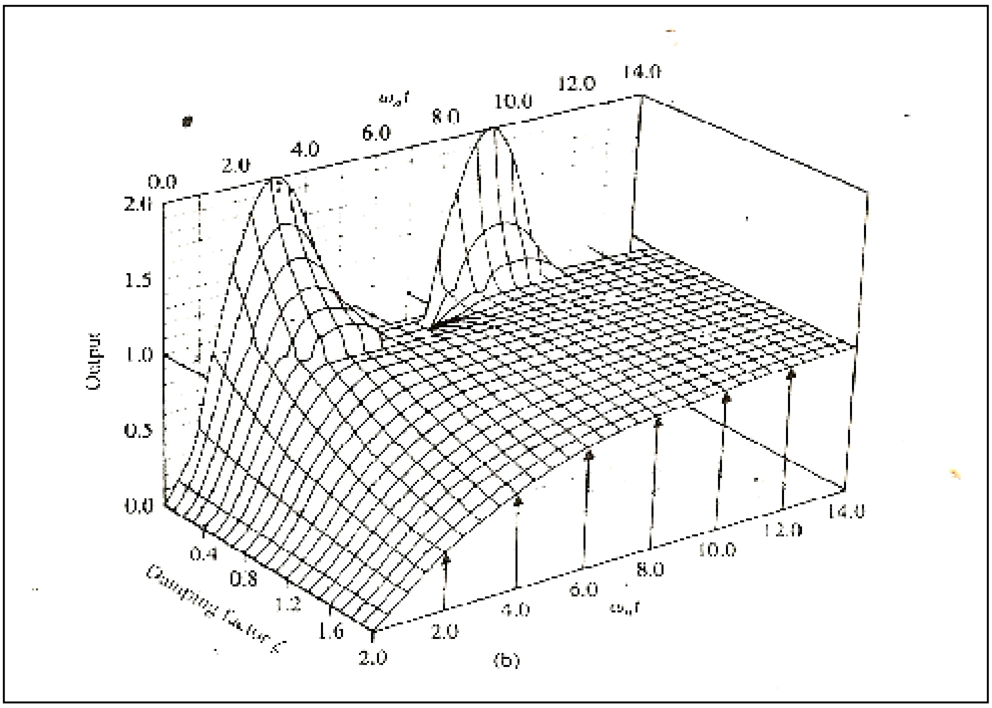

Fig. (3b): Transient response of a second order system as a function of time

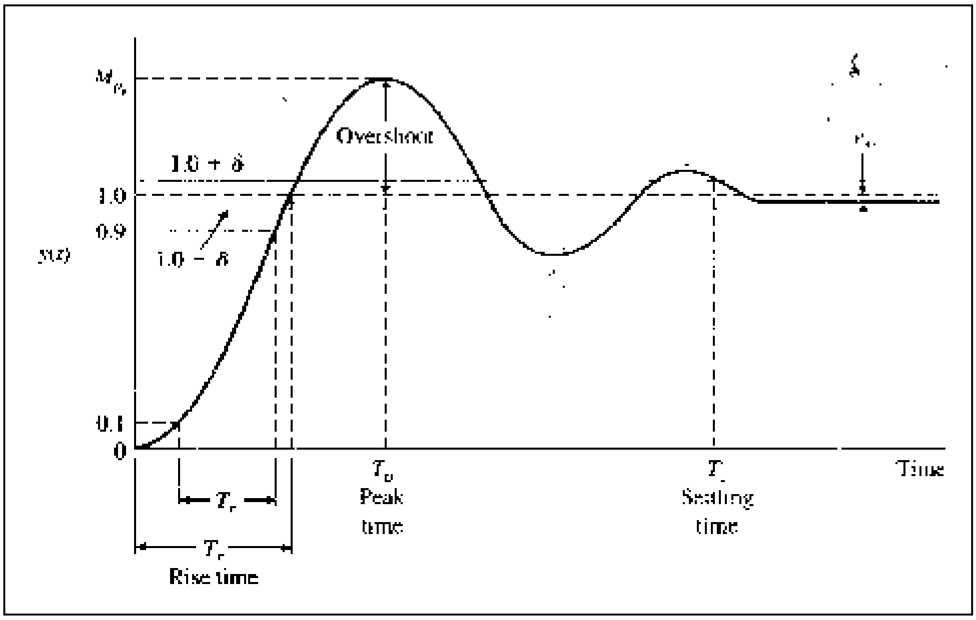

Fig. (4): Step response of a control system.

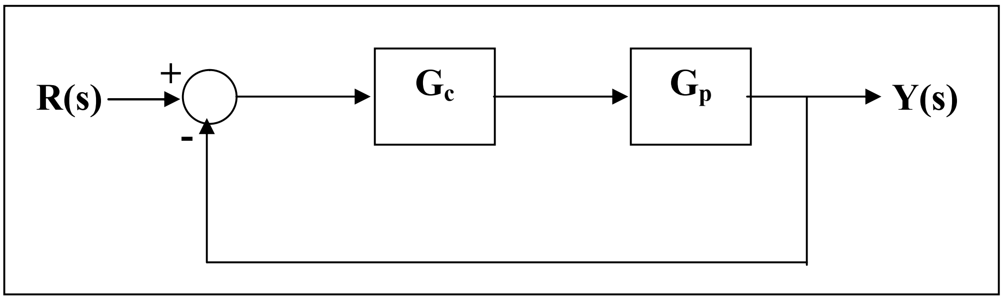

Fig.(5): Feedback control system with a controller and a plant transfer function.

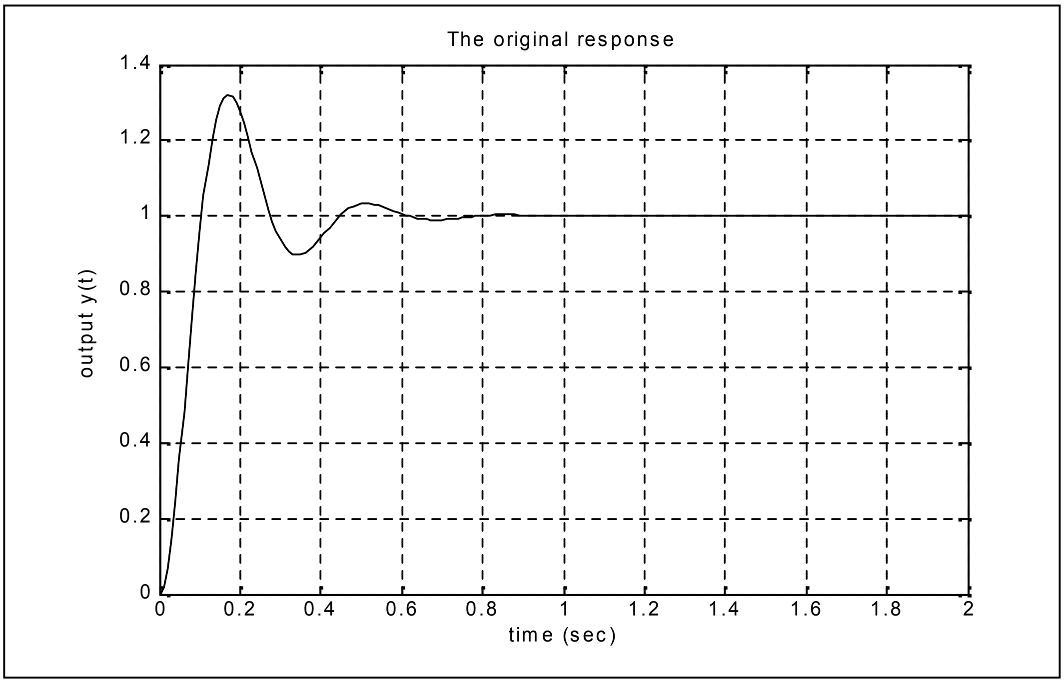

Fig. (6): The original step response

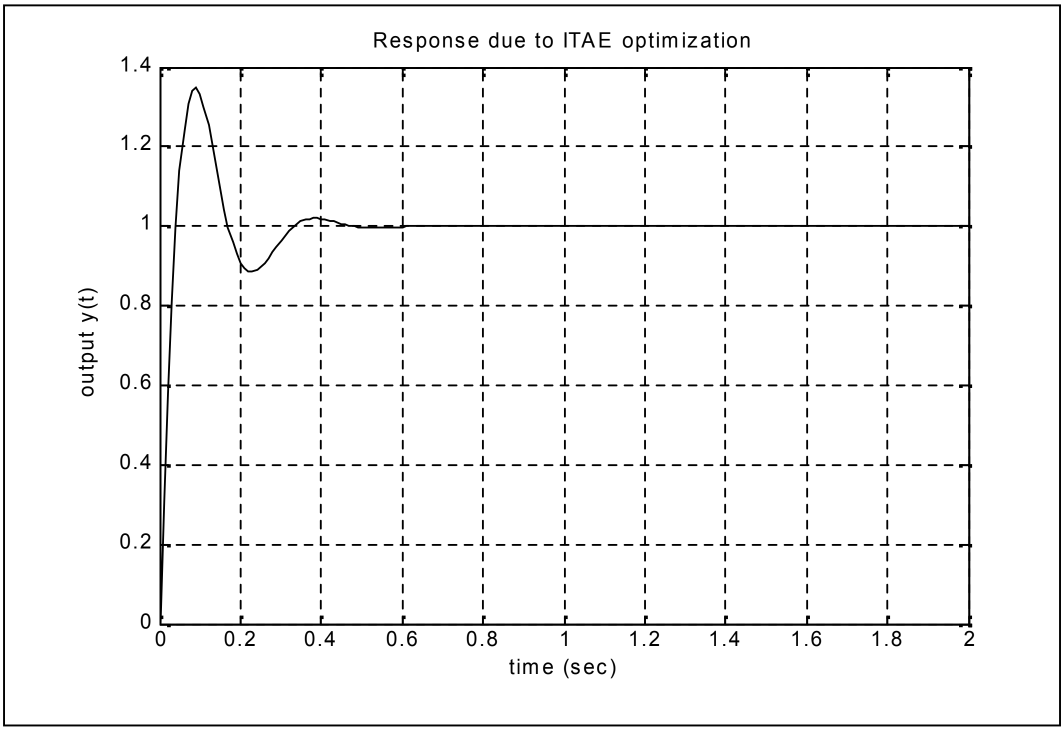

Fig. (7): The ITAE step response

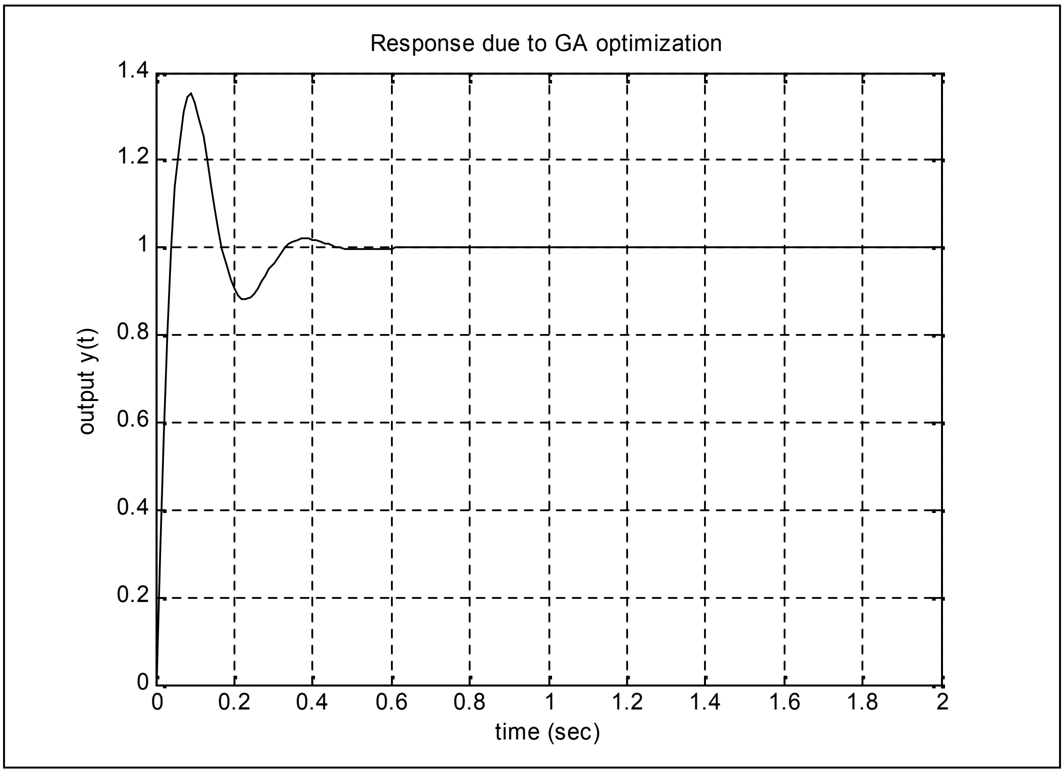

Fig. (8): The GA response

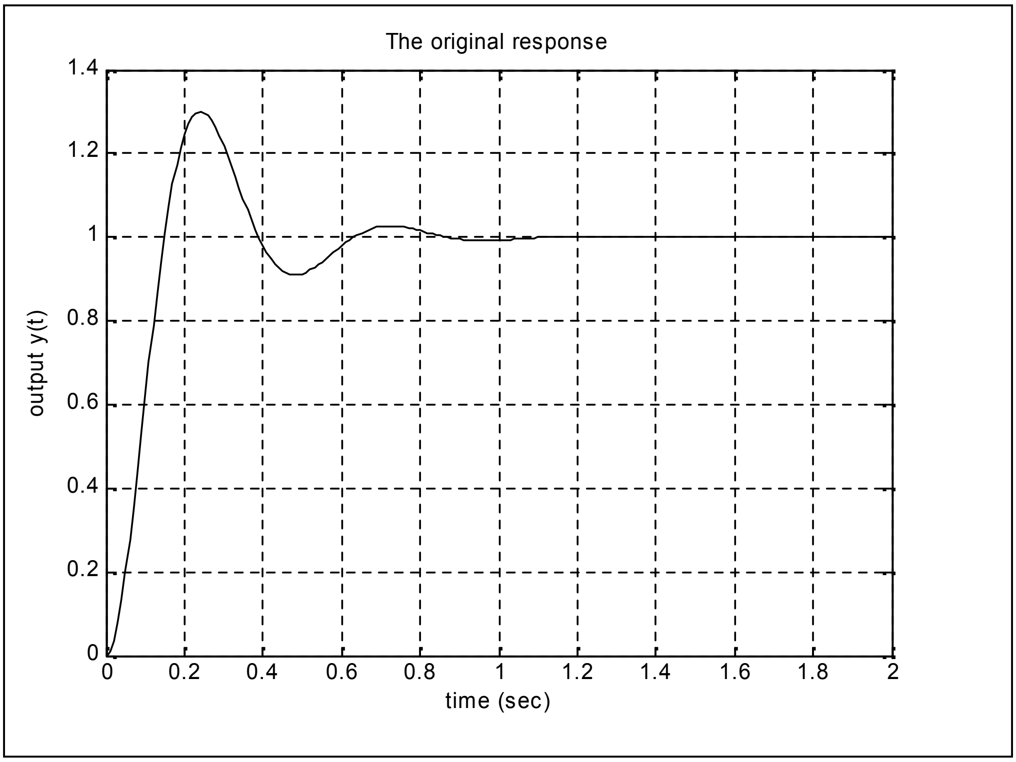

Fig. (9): The original step response

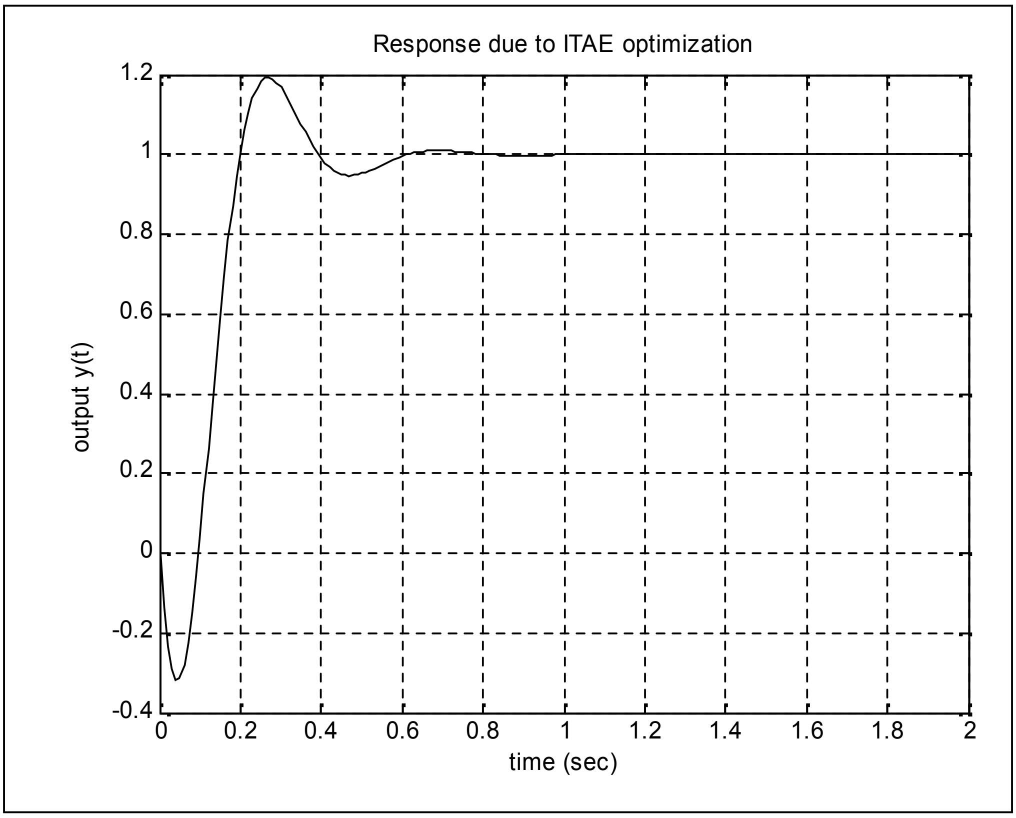

Fig. (10): The ITAE step response

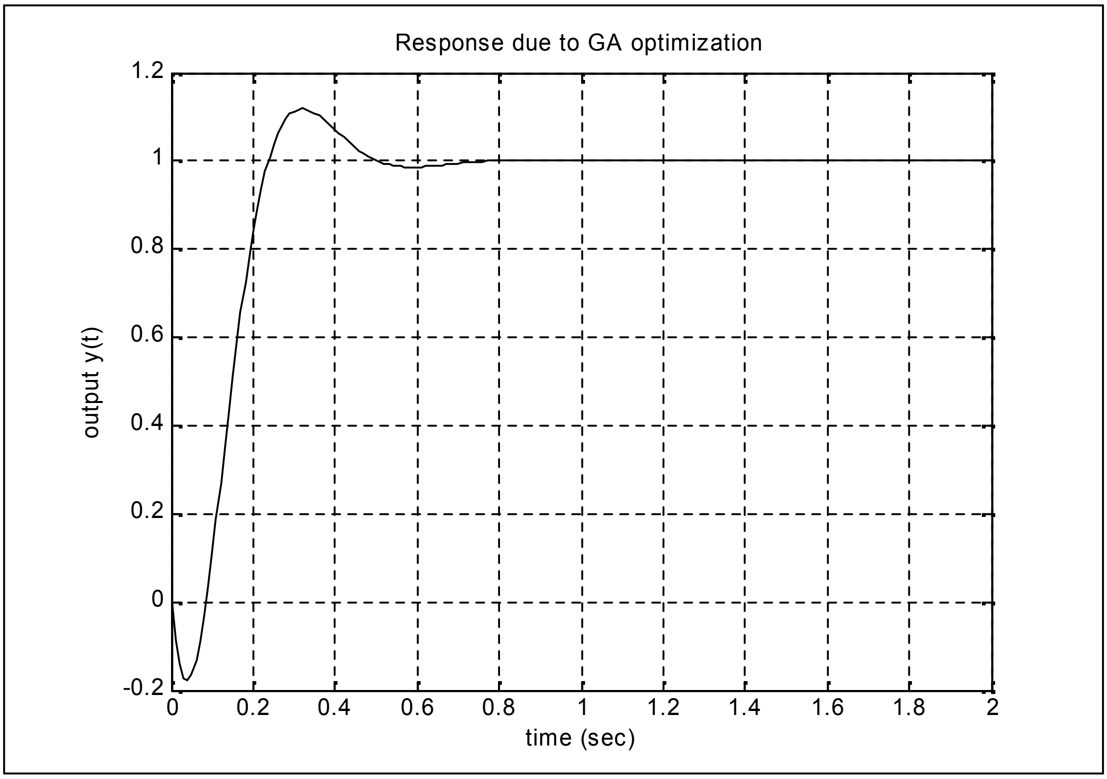

Fig. (11): The GA response

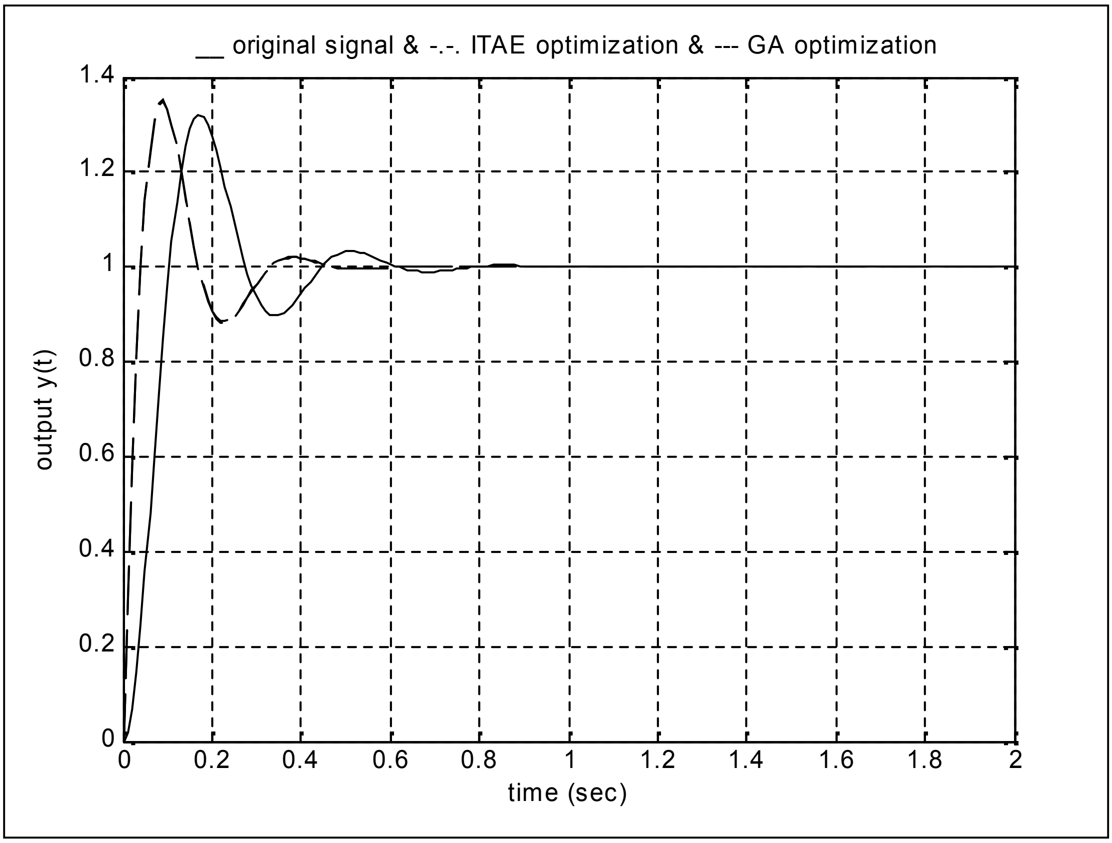

Fig. (12): A comparison for the three responses for the first example

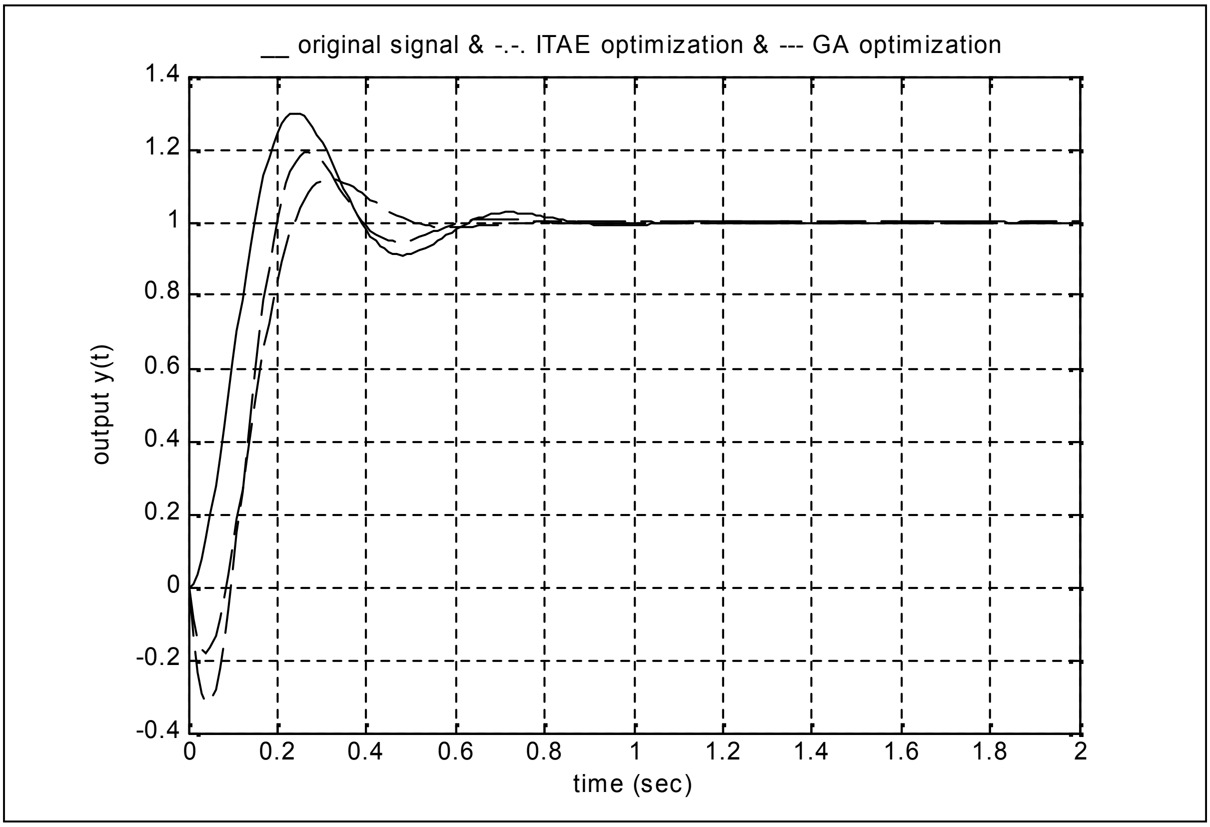

Fig. (13): A comparison for the three responses for the second example