# GENETIC ALGORITHMS BASED OPTIMIZATION DESIGN OF TUNNING A PID CONTROLLER

**Reath .J. Tulafeh**
**Mechanical Engineering -Technical Collage / Almusiab**

**Abstract:**

The main aim of this paper is to tune a PID controller to enhance the out put response of fifth order modal with sensor in feedback control system . This controller tuned by genetic algorithms, because the fifth order system is notoriously difficult to control optimally using conventional PID controller. Genetic Algorithms (GA) are effective at finding high performance areas in large domains and the ideal choice to tune the PID controller can be achieved. Genetic Algorithms were examined in detail, it was decided to create an objective function which evaluates the optimum PID gains based on the controlled systems overall error. GA’s outperformed standard tuning practices, e.g. Ziegler Nichols, at designing of PID controllers, in the tests carried out. Experimentally, it can be determined that the Integral Square Error (ISE) performance criterion produces the most effective PID controllers compared with other performance criterion.

**Key Words:** PID controller , intelligent control technique, genetic algorithms, AVR, fifth order plant

# تمثيل أنظمة السيطرة ذات الدرجة الخماسية باستخدام برنامج الماتلاب وتحسين ادائها باستخدام المسيطر الثلاثي المنغم بالخوارزميات الجينية

**م.م. رياض جاسم طليفح**
**هندسة ميكانيكية- الكلية التقنية- المسيب- قسم هندسة تقنيات المضخات**

**الخلاصة :**

الهدف الرئيسي في هذا البحث هو انشاء مسيطر ثلاثي (PID)منغم بطرق ذكية لتحسين اداء انظمه من الدرجة الخماسية تحوي على متحسس في الاشارة الراجعه من الاخراج ، هذا المسيطر تم تنغيمه باستخدام الخوارزميات الجينية GA . حيث ان يصعب السيطره على هكذا نوع من الانظمة عليها باستخدام المسيطر الثلاثي المنغم بالطرق التقليدية مثل طريقة (Ziegler Nichols) الخوارزميات الجينية ذات تاثير كبير في ايجاد المتغيرات للمسيطر الثلاثي وقد اختبرت الطريقة واستخدم عامل (ISE) لايجاد الاستجابة الجيده .تم الحصول على نتائج جيده ذات معامل خط قليل أي استجابه جيده وقورنت النتائج مع الطرق السابقة للتنغيم واوضحت بان هذه الطريقه الذكيه بالتنغيم جيده جدا .

## Introduction:

PID controller is a one of the earliest industrial controllers. It has many advantages: Its cost is economic, simple easy to be tuned and robust. This controller has been proven to be remarkably effective in regulating a wide range of processes [Astrom and Hagglund 1988 ]. More than 90% of industrial controllers are still implemented based around PID algorithms, particularly at lowest levels. With its three-term functionality covering treatment to both transient and steady-state responses, proportional- integral-derivative (PID) control offers the simplest and yet most efficient solution to many real-world control problems. PID controller is a convenient fractional order structure that has been employed for control purposes [Ang 2005]. An FOPID is characterized by 3 parameters: the proportional gain, the integrating gain, the derivative gain, the integrating order and the derivative order. There are a number of efficient search algorithms that have their origins in the field of biological evolutionary and are known as evolutionary computation , the field of evolutionary computation is comprised mainly of Genetic algorithms ,genetic programming and evolutionary strategies [Ang 2005]

## PID Controller and Optimization Algorithms

### PID Controller

Over 90% of the controllers in operation are PID controllers. Despite the development of more advanced control strategies, the majority of industrial control systems still use PID controllers because they are standard industrial components. **Figure (1)** shows the control system with PID controller. The three parameters that must be determined (some times, must be optimized) for the given process[Kuo and Golnaraghi2003 ], to give the desirable output responses for the plant are: proportional gain, integral gain and derivative gain. The transfer function of the PID controller looks like the following:

$$C(s) = K_p + K_i / s + K_d s = (K_d s^2 + K_p s + K_i) /s \qquad \ldots\ldots(1)$$

□ $K_p$ = Proportional gain
□ $K_i$ = Integral gain
□ $K_d$ = Derivative gain

The error signal (e) will be sent to the PID controller, and the controller computes both the derivative and the integral of this error signal. The signal (u) just past the controller is given as:

$$u= K_p.e + K_i \int_0^t e.dt + K_d\, de/dt \qquad \ldots\ldots\ldots (2)$$

This signal will be sent to the plant, and the new output (y) will be obtained. This new output (y) will be sent back to the sensor again to find the new error signal (e). The controller takes this new error signal and computes its derivative and its integral again. This process goes on and on [Jack2004]. **Table (1),** illustrates Effect of PID controllers parameters on transient response. These correlations may not be exactly accurate, because $K_p$, $K_i$, and $K_d$ are dependent of each other. In fact, changing one of these variables can change the effect of the other two. For this reason, the table should be used as a reference when determining the values for $K_i$, $K_p$ and $K_d$.

The steady state response can be improved by the addition of an integral term, and the speed of response can be increased by the addition of a derivative term. A common PID control transfer function in equation (2-1) can be re-written as:

$$C(s) = \frac{U(s)}{E(s)} = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) \quad \text{......... (3)}$$

Where the controller gains

$$K_i = \frac{K_p}{T_i} \quad \text{........ (4)}$$

$$K_d = K_p T_d \quad \text{......... (5)}$$

**PID Controller Tuning**

Many methods to determine the PID parameters have been developed during the last four decades. All these methods tried to optimize some performance criteria that mostly constitute a nonlinear optimization problem, in which many local minimums may be present [Becerra 2003 ].
The most popular applications of the PID controllers in power system control are in the control circuits of power generation plants (Synchronous Generator), either in Load Frequency Control (LFC) as a power system stabilizer to control the load angle variation and stability of the power system, or as an auxiliary regulating controller inserting in the Excitation Control System together with the Automatic Voltage Regulator to control and enhance the terminal voltage transient stability response [Jones and Moura 1995 ].

All the previous or conventional tuning methods for the PID parameters (gains) are time consuming and some times manually done. Even if real time expert systems are used to automate the tuning process, also these methods utilize only a few available information's about the dynamic behavior of the system since the power system is a dynamic system, so they often yield no proper tuning [Jones and Moura 1995 ].

One of the must popular conventional tuning methods for the PID controller is the modified Ziegler Nichols [Ziegler and Nichols*1942*]. This method is very simple and practical. The system is placed under proportional control and taken to the limit of stability by increasing the gain until permanent oscillations are achieved. The gain at which this occurs is called the ultimate gain (*Ku*), and the period of this oscillation is known as the ultimate period (*Tu*). With these two parameters, the controller parameters *Kp, Ti* and *Td* can be calculated as shown in **Table (2).** The other popular conventional tuning methods for PID controller are the classical tuning method which depends on the trial and error,. The main part of the objectives of this research is to design and tune a powerful controller that based on the Genetic Algorithm optimization technique**.** In other words, to optimize the parameters of the PID controller using Genetic Algorithm instead of conventional tuning method that mentioned above. The PID parameters are obtained by minimizing one of the most popular error criteria which is the ISE criteria, where the minimization is performed using Genetic Algorithm (GA) [Ziegler and Nichols*1942*]. The design of adaptive controllers to improve the performance of the power system has been a topic of research for a long time in the electrical power system field, because, the Parameters in this system are changing with time, slowly due to environmental effects or rapidly due to faults. Thus it is necessary to update the controller parameters with system changes, so the need for the intelligent adaptive controller being very essential. The world directed toward using the intelligent optimization tuning methods. One of the best robust method and have been enjoying increasing popularity in the field of numerical optimization in recent year is the Genetic Algorithm which will be discussed in details in the next section.

**Error Criteria in PID Controller:**

In order to select the best controller, we define a cost function. The cost function mainly derives on how the controller reacts to a given disturbance. There are many types of cost functions (optimization criteria) and some times it called performance indexes. In fact, we can define infinitive criterions. The most popular are [Kuo and Golnaraghi2003]:

Integral of absolute value of Error $\mathbf{IAE} = \int_0^\infty e(t)\, dt$ ……( 6)

Integral of Error Squared $\mathbf{ISE} = \int_0^\infty e(t)^2 dt$ ……..(7)

Integral Time Weighted Error $\mathbf{ITAE} = \int_0^\infty t\, e(t)\, dt$ ……....(8)

For the above functions we simulate the results so as to optimize the controller. PI (D) control is perhaps the most basic form of feedback. It is very effective and can be applied to a wide range of problems. This type of control has become one of the most important ways, for the scientific and the industrial control users to work together especially in power system control, in prescience words, in Synchronous Machine Excitation Control system [Goodwin etal.2000].

**Genetic Algorithms**

Genetic Algorithms (GA's) are a stochastic global search method that mimics the process of natural evolution. The genetic algorithm starts with no knowledge of the correct solution and depends entirely on responses from its environment and evolution operators (i.e. reproduction, crossover and mutation) to arrive at the best solution. By starting at several independent points and searching in parallel, the algorithm avoids local minima and converging to sub optimal. A genetic algorithm is typically initialized with a random population consisting of between 20-100 individuals. This population (mating pool) is usually represented by a real-valued number or a binary string called a chromosome. For illustrative purposes, the rest of this section represents each chromosome as a binary string. How well an individual performs a task is measured is assessed by the objective function. The objective function assigns each individual a corresponding number called its fitness. The fitness of each chromosome is assessed and a survival of the fittest strategy is applied. [. Holland1992 ]. In this project, the magnitude of the error will be used to assess the fitness of each chromosome. There are three main stages of a genetic algorithm, these are known as *reproduction, crossover* and *mutation*.

**Reproduction**

During the reproduction phase the fitness value of each chromosome is assessed. This value is used in the selection process to provide bias towards fitter individuals. Just like in natural evolution, a fit chromosome has a higher probability of being selected for reproduction. An example of a common selection technique is the '*RWl'* selection method, Each individual in the population is allocated a section of a roulette wheel; the size of the section is proportional to the fitness of the individual as

shown in **Figure 2**. A pointer is spun and the individual to whom it points is selected. This continues until the selection criterion has been met. The probability of an individual being selected is thus related to its fitness, ensuring that fitter individuals are more likely to leave offspring.
Multiple copies of the same string may be selected for reproduction and the fitter strings should begin to dominate [Passino and Yorkovich 1998]. There are a number of other selection methods available and it is up to the user to select the appropriate one for each process. All selection methods are based on the same principal i.e. giving fitter chromosomes a larger probability of selection solutions [Holland1992].
Four common methods for selection are:
1. Roulette Wheel selection
2. Stochastic Universal sampling
3. Normalised geometric selection
4. Tournament selection

**Crossover**

Once the selection process is complete, the crossover algorithm is initiated. The crossover operations swaps certain parts of the two selected strings in a bid to capture the good parts of old chromosomes and create better new ones. Genetic operators manipulate the characters of a chromosome directly, using the assumption that certain individual's gene codes, on average, produce fitter individuals. The crossover probability indicates how often crossover is performed. A probability of 0% means that the 'offspring' will be exact replicas of their 'parents' and a probability of 100% means that each generation will be composed of entirely new offspring. The simplest crossover technique is the Single Point Crossover. There are two stages involved in single point crossover [Holland1992].:
1. Members of the newly reproduced strings in the mating pool are 'mated' (paired) at random.
2. Each pair of strings undergoes a crossover as follows: An integer $k$ is Randomly selected between one and the length of the string less one, [$1,L-1$]. Swapping all the characters between positions k+1 and L inclusively creates two new strings.

More complex crossover techniques exist in the form of Multi-point and Uniform Crossover Algorithms. Multi-point crossover is an extension of the single point crossover algorithm and operates on the principle that the parts of a chromosome that contribute most to its fitness might not be adjacent. There are three main stages involved in a Multi-point crossover.
1. Members of the newly reproduced strings in the mating pool are 'mated' (paired) at random.
2. Multiple positions are selected randomly with no duplicates and sorted into ascending order.
3. The bits between successive crossover points are exchanged to produce new offspring.

In uniform crossover, a random mask of ones and zeros of the same length as the parent strings is used in a procedure as follows.

**Mutation**

Using *selection* and *crossover* on their own will generate a large amount of different strings. However there are two main problems:
1. Depending on the initial population chosen, there may not be enough diversity in the initial strings to ensure the GA searches the entire problem space.
2. The GA may converge on sub-optimum strings due to a bad choice of initial population.

These problems may be overcome by the introduction of a mutation operator into the GA. Mutation is the occasional random alteration of a value of a string position. It is considered a background operator in the genetic algorithm

The probability of mutation is normally low because a high mutation rate would destroy fit strings and degenerate the genetic algorithm into a random search. Mutation probability values of around 0.1% or 0.01% are common, these values represent the probability that a certain string will be selected for mutation i.e. for a probability of 0.1%; one string in one thousand will be selected for mutation. Once a string is selected for mutation, a randomly chosen element of the string is changed or 'mutated'[Holland1992]. The steps involved in creating and implementing a genetic algorithm are as follows

1. Generate an initial, random population of individuals for a fixed size.
2. Evaluate their fitness.
3. Select the fittest members of the population.
4. Reproduce using a probabilistic method (e.g., roulette wheel).
5. Implement crossover operation on the reproduced chromosomes
(choosing probabilistically both the crossover site and the 'mates').
6. Execute mutation operation with low probability.
7. Repeat step 2 until a predefined convergence criterion is met.

The convergence criterion of a genetic algorithm is a user-specified condition e.g. the maximum number of generations or when the string fitness value exceeds a certain threshold.[ Holland1992].:

.

**Elitism**

With crossover and mutation taking place, there is a high risk that the optimum solution could be lost as there is no guarantee that these operators willpreserve the fittest string. To counteract this, elitist models are often used. In an elitist model, the best individual from a population is saved before any of these operations take place. After the new population is formed and evaluated, it is examined to see if this best structure has been preserved. If not, the saved copy is reinserted back into the population. The GA then continues on as normal

**Genetic Algorithms versus Traditional Methods**

Genetic algorithms are substantially different to the more traditional search and optimization techniques. The five main differences are [Holland1992].:

1. Genetic algorithms search a population of points in parallel, not from a single point.
2. Genetic algorithms do not require derivative information or other auxiliary knowledge; only the objective function and corresponding fitness levels influence the direction of the search.
3. Genetic algorithms use probabilistic transition rules, not deterministic rules.
4. Genetic algorithms work on an encoding of a parameter set not the parameter set itself (except where real-valued individuals are used).
5. Genetic algorithms may provide a number of potential solutions to a given problem and the choice of the final is left up to the user.

## Simulation Result for Tuning PID Controller

In this paper a system was chosen and a PID controller was designed for it using conventional methods and modern method to evaluate the output response of the system and the results of the two techniques were compared. The system chosen was: the fifth Order plant with sensor and PID Controller is shown in **Figure( 4 )**

### Conventional Methods (Ziegler-Nichols Designed PID Controller)

In this section the obtained simulation model for the fifth order plant will be examined by using a conventional PID controller tuned through using a Ziegler – Nichols method , which is classical tuning method as explained below. The optimized PID controllers parameters values using conventional tuning method are also represented. All Figures with a simulation time of 1 second coincide with the X – axis, while the terminal response in each figure in Y – axis .The Ziegler-Nichols (Z-N) methods rely on open-loop step response or closed-loop frequency response tests [J. G. Ziegler and N. B. Nichols, Optimum Setting for Automatic Controllers, *1942.*]. A P, PI or PID controller is tuned according to a **Table (3)** based on the process response. The PID tuning is calculated from the Z-N tuning table found in literature such as Åström and Hägglund (1995). **Table (3)** shows the tuning rules for a PID controller. P and PI controllers have separate tuning rules. For a discrete-time PID controller the table should be revised to take the sampling time into account. There is, however, no discrete Z-N tuning table, so **Table (3)** is used in lack of other. A discrete-time controller approximates a continuous-time controller at small sampling times, so this tuning rule is only used for the shortest sampling times. The Z-N method is designed for rejecting load disturbances. For reference step changes it performs worse and gives for simple systems a damped oscillating response. The tuning is usually bad for higher order systems and can only be considered as a simple first aid tuning. For comparison without controller response is shown in **Figure (5)** also the error signal is shown in **Figure (6).** When the PID controller tuned by conventional method used the response is enhanced and the good time parameter is obtained as shown in **Figure( 7)** and **Figure (8)**

### Intelligent Method (Genetic PID Controller)

Genetic algorithm will be applied to the area of PID optimization in both an off-line and on-line tuning environment. Tuning a system off-line means that the PID parameters of the controller are updated when the system has been taken off-line. The PID values are updated using the systems input and output data after the system has been placed.

These updated PID values are used in place of the old PID values and the system is brought back on-line. This process continues until the optimum PID coefficients of the system have been obtained. Tuning a system on-line means that the PID parameters of a controller are updated while the system is on-line. The systems input and output data are collected at regular intervals and used to update the PID values. The new PID values are then used in place of the old ones until a newer updated set of PID values have been deduced at the next time interval. The optimizations of the controller's PID parameters using genetic algorithms will be compared to that of a standard method for designing PID controllers (Ziegler-Nicholas tuning method). The Genetic Algorithms are used here to find the optimum PID values for the $5^{th}$ order system model. The important steps of GA-PID controller are explained below:

The parameters types for the used genetic algorithms are ([Holland1992]):

- Type of Selection = Roulette Wheel Selection.

The Selection is a method for increasing the number of solution candidates having high fitness values. There are several types of selection such as (Roulette Wheel Selection (RWS), and Rank selection).

- Type of Crossover = Single point Crossover.

The GA is not only filling the population with the best of the first generation by choosing better solutions, but there are also created new solutions from the original database which can be done by crossover operator. There are several types of crossover operator, some of these are (Single, Two, multi Point Crossover and Arithmetic Crossover).

- Type of mutation = Uniform mutation

Mutation operator is another degree of freedom in search procedure, which is frequently used in design of GA. The need for mutation is to keep diversity in the population. There are many ways of accomplishing mutation such as binary, Uniform, Non-Uniform and Boundary mutation (K. M. Passino and S. Yorkovich, ***Fuzzy Control***, 1998.].

- Fitness function =1/(ISE+0.00001)

The fitness (objective function) is chosen depending on the problem in hand such that the individuals having high fitness values are the good solution candidates for the optimization. Therefore, selection of the next generation will dependent on the fitness measure . The Optimization criterion which is applied in the present work is Integral of square error (ISE).

- Probability of Crossover = 0.95
- Probability of Mutation = 0.01
- Max. NO. of Generation = 500
- Population size = 200

**Figure (9)** show the output response when GPID controller. **Figure (10)** shows the error response is

**-Transfer Function for PID controller:**

$$T.F = \frac{0.01312\, s^2 + 0.23015\, s + .29113}{s} \qquad (9)$$

- Optimized Genetic for PID :

**Kp = 0.13, Ki = 1.02, Kd = 0.001**

-Transfer Function of Fifth Order Model with sensor modal as shown in **Figure (4 )** and Genetic - PID Controller is(K. M. Passino and S. Yorkovich,1998)s

Transfer function =

$$\frac{3.995e\text{-}009\, s^7 + 6.553e\text{-}006\, s^6 + 0.001944\, s^5 + 0.185\, s^4 + 4.357\, s^3 + 30.34\, s^2 + 54.81\, s + 29.11}{1.152e\text{-}008\, s^7 + 1.429e\text{-}005\, s^6 + 0.002247\, s^5 + 0.1146\, s^4 + 4.013\, s^3 + 31.89\, s^2 + 54.35\, s + 29.11} \qquad (10)$$

## Conclusions

From the simulation result ,the conventional PID tuning methods are time consuming and some times depends on the human experts, but the GAs are the best choices and used here as optimizing techniques for PID parameters optimization and the obtained results of the ISEs are comparable.

-The genetic algorithm is really powerful search optimization algorithm, and when number of generation is increase the best result is obtain but the computation time for the execution program is increased also.

-The overall enhancements results of the performance Indexes or the optimization criteria (ISEs), which being a measures for the terminals transient voltage stability enhancements through all the works

## References:

K. J. Astrom and J. Hagglund, “Automatic Tuning of PID Controllers”, Instrument Society of America, 1988

K. H. Ang, G. Chong, and Y. Li, "PID Control System Analysis, Design, and Technology" IEEE Transactions on Control Systems Technology , Vol. 13, No. 4, July, 2005.

B.C. Kuo and F. Golnaraghi, **Automatic Control Systems,** John Willy &Sons, Inc, 8th edition, USA, 2003.

[4] H. Jack, **Dynamic System Modeling and Control,** Hugh Jack Press, USA, 2004.

V. M. Becerra, **Process Control Simulating using Simulink**, Sept. 2000, Updated for Matlab 6.5 – June 2003.

A.H. Jones and P. B. De Moura Oliveira, **"Genetic Auto-Tuning of PID Controller"**, Genetic Algorithms in Engineering Systems: Innovations and Applications, 12-14 September 1995, Conference Publication No. 414, IEE, pp. 141-145, 1995.

J. G. Ziegler and N. B. Nichols, **"Optimum Setting for Automatic Controllers",** Trans. ASME, Vol. 65, pp.433-444, 1942.

G. C. Goodwin, S. F. Graebe and M. E. Salgado, **Control System Design,** Santa Maria Press, Valparaiso, Chile, 2000.

J. Holland, **Adaptation in Natural and Artificial Systems,** 2nd edition, MIT Press / Bradford Book Edition, Cambridge, Massachusetts, 1992.

K. M. Passino and S. Yorkovich, **"Fuzzy Control"**, Addison – Wesley Longman, Inc., 1998.

A.A. Albakry " Optimize parameters controller of synchronous machine to enhance transient t response " , Department of Electrical Engineering, University of Technology Iraq, Thesis, pp. 90-95,2007

Table (1): Effects of PID controllers parameters on transient response

| CL RESPONSE | Rise Time | Overshot | Settling Times | S.S Error |
|---|---|---|---|---|
| **$K_P$** | Decrease | Increase | Small change | Decrease |
| **$K_i$** | Decrease | Increase | Increase | Eliminate |
| **$K_d$** | Small Change | Decrease | Decrease | Small Change |

Table (2): Modified Ziegler-Nichols PID tuning

| | *Kp* | *Ti* | *Td* |
|---|---|---|---|
| **P** | 0.33 *Ku* | 0 | 0 |
| **PI** | 0.33 *Ku* | 2 *Tu* | 0 |
| **PID** | 0.20 *Ku* | 0.8 *Tu* | 0.2 *Tu* |

Table (3): Ziegler-Nichols tuning table for PID controller.

| Method/Parameter | $K_p$ | $T_i$ | $T_d$ |
|---|---|---|---|
| Z-N step response | $1.2\frac{K_1}{K_2}\frac{T_2}{T_1}$ | $2T_1$ | $0.5T_1$ |
| Z-N frequency response | $0.6K_u$ | $0.5T_u$ | $0.125T_u$ |

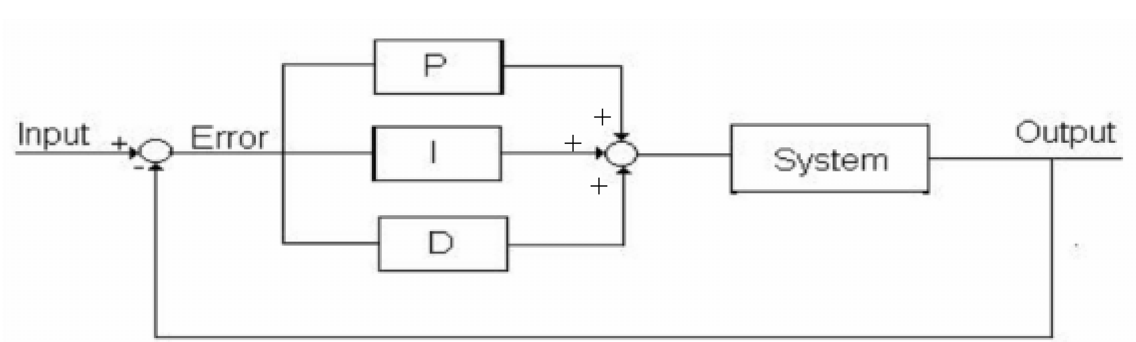


Figure 1 : Closed Loop Control system with PID controller

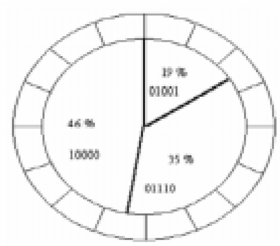


Figure 2. Depiction of RW selection

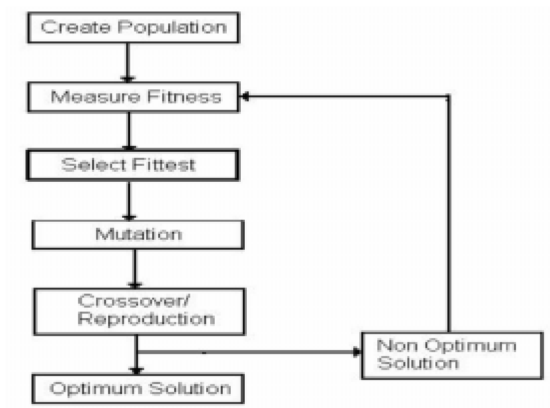


Figure 3 : Genetic Algorithms block diagram representation[9]

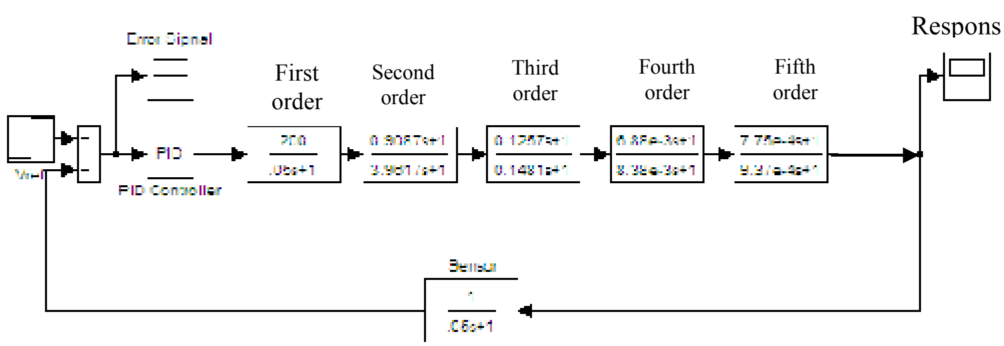


Figure 4: Fifth Order plant with sensor and PID Controller block diagram

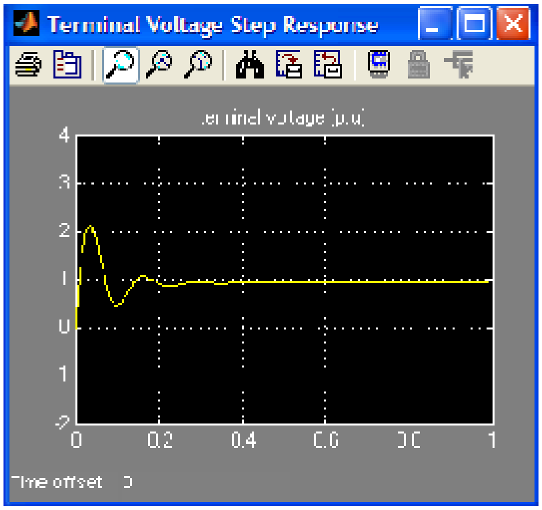


Figure 8: Terminal Voltage Step Response for Fourth Order Model With PID Controller

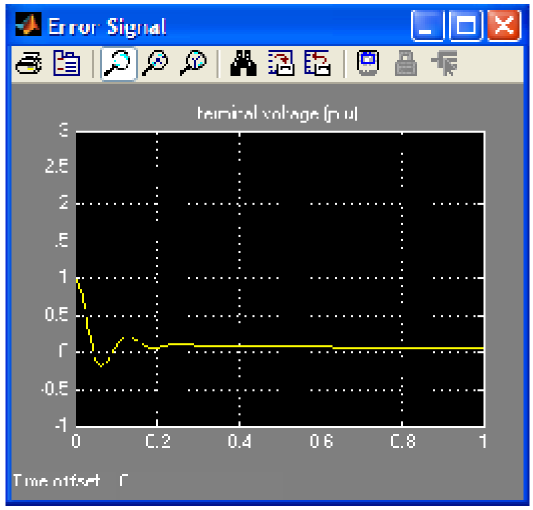


Figure 9: Error Signal of Terminal Voltage Step Response for Fourth Order Model with PID controller

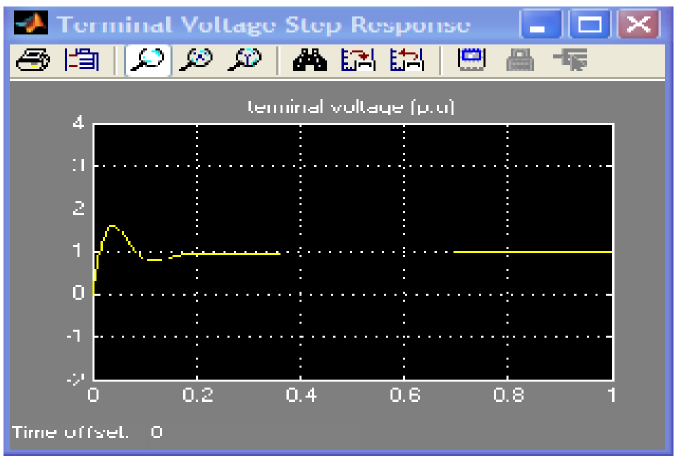


Figure 9 : Terminal Voltage Step Response for 5th Order Model with GPID

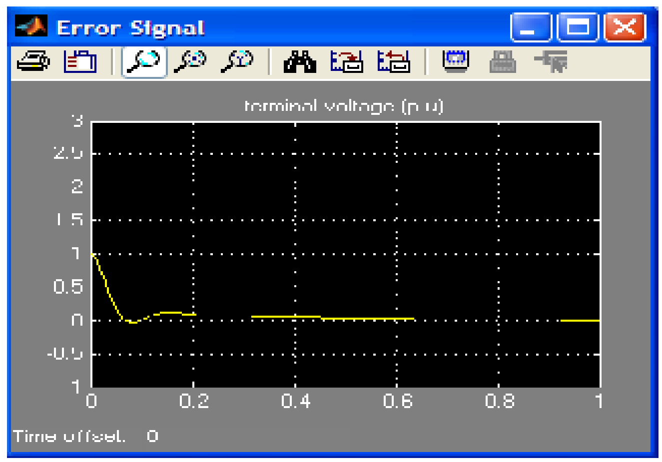


Figure 10 : Error Signal for Terminal Step Response of 5$^{th}$ Order Model with GPID