*Nigerian Journal of Engineering*
*Faculty of Engineering*
*Ahmadu Bello University*
*Samaru - Zaria, Nigeria*

*Vol. 25, No. 1, Sept. 2018*
*ISSN: 0794 - 4756*

# OPTIMIZATION OF PID CONTROLLER PARAMETERS FOR DEEP SPACE ANTENNA POSITIONING SYSTEM USING GENETIC ALGORITHM

**E. E. Idenyi[1], I. M. Dagwa[2] and M. O. Afolayan[1*]**

**[1]Mechanical Engineering Department, Ahmadu Bello University, Zaria, Nigeria.**
**[2] Mechanical Engineering Department, University of Abuja, Nigeria.**
***Corresponding Author email: tunde_afolayan@yahoo.com**

## ABSTRACT

*Proportional-Integral-Derivative (PID) controller's parameters for deep space antenna positioning system were optimized using Genetic Algorithm (GA). The use of GA resulted in the optimum controller parameters being selected for the system every time. Matlab/ Simulink environment was used in determining the optimum value for its parameters. Simulation result showed that the performance of the optimized PID Controller gave a response values of 2.2412sec rise time, 2.9861sec settling time and 0% overshoot and undershoot and is comparably better than the conventionally, Zeigler-Nichols method, tuned controller response values of 0.8568sec rise time, 9.2289sec settling time, 66.3812% overshoot and 23.1264% undershoot at an amplifier gain value of 100 for both method. Results for different amplifier gain values also show that the system response at an amplifier gain of 250 produced the best response in terms of rise time, settling time and overshoot but has a problem of distorted response in its transient state characteristics.*

***Keywords:*** *Deep Space Antenna, Genetic Algorithm (GA), Proportional-Integral-Derivative (PID) Controller, Optimization, Tuning.*

## INTRODUCTION

Antennas are electrical devices which convert electric power into radio waves, and vice versa. Deep space antenna communicates with spacecraft by sending commands (uplink) and receiving information (downlink) from it (Gawronski, 2008). An antenna tracking (the act or process of following the trail) a satellite must keep the satellite well within its beam-width in order not to lose track (Nise, 2006). In order to ensure this due to Earth's rotation, the antenna shown in Figure 1 is continuously positioned with the aid of a controller and a drive mechanism. This implies that suitable and efficient positioning of antenna structure will enhance signal clarity, wider coverage area and satisfactory reception of radiated signal (Agubor *et al.*, 2013).

The antenna dish rotates with respect to the horizontal axis while the whole structure rotates on a circular track with respect to the vertical axis. The position of antenna is controlled by using gears and feedback potentiometer. Antenna positioning is also controlled by using some controllers (Chisti *et al.*, 2014). A controller aims at minimizing the error between a measured process variable of the controlled system and a reference, by calculating the error and generating a correction signal to the system from the error (Pillai *et al.*, 2013, Prasanna *et al.*, 2016 and Surya *et al.*, 2014).

A deep space antenna positioning system is shown in figure 2. The purpose of this system is to have the azimuth angle output of the antenna, $\Theta_o (t)$, following the input angle of the potentiometer, $\Theta_i (t)$. The input command is an angular displacement. The potentiometer converts the angular displacement into a voltage. Similarly, the output angular displacement is converted to a voltage by the potentiometer in the feedback path. After that, differential amplifier checks how much the obtained signal is different from the given signal and also find the error. The signal and power amplifiers boost the difference between the input and output voltages. This amplified actuating signal drives the system (Nise, 2006; Okumus *et al.*, 2012). The system normally operates to drive the error to zero. When the input and output match, the error will be zero, and the motor will not turn. Thus, the motor is driven only when the output and the input do not match. The greater the difference between the input and the output, the larger the motor input voltage, and the faster the motor will turn (Nise, 2006). The motor used is a fixed field DC servo motor (Okumus *et al.*, 2012). For getting better response, several controllers like Proportional-Integral-Derivative (PID) Controller, Linear Quadratic Regulatory (LQR) Controller, Fuzzy Logic Controller (FLC) *etc.* have been proposed and used (Astrom, *et al.*, 1995; Franklin *et al.*, 2002; Kiam *et al.*, 2005; Ogata, 2007; Pillai, *et al.*, 2013). Other approach uses Axiomatic Design methodology, "which focuses on the mapping between customer needs into instantiation" (Joseph *et al.*, 2017).

Figure 1.0: A deep space antenna (Gawronski, 2008)

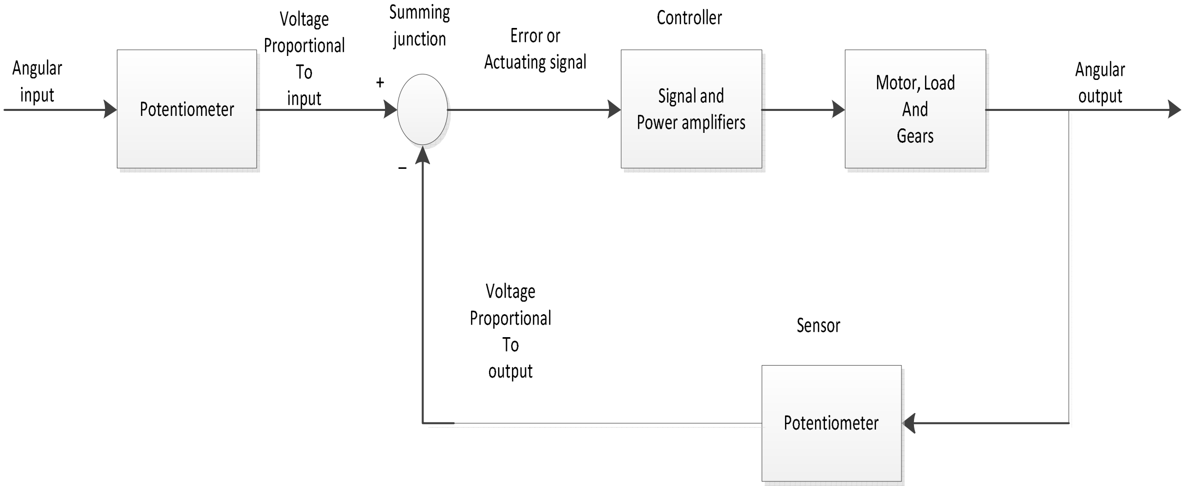

Figure 2: Antenna azimuth position control system block diagram (Nise, 2006).

**Genetic Algorithm (GA)**

The key issue for PID controllers is the accurate and efficient tuning of its parameters. In practice, controlled systems usually have some features, such as nonlinearity, time-variability, and time delay, which make controller parameter tuning more complex. Moreover, in some cases, system parameters and even system structure can vary with time and environment. As a result, the traditional PID parameter tuning methods such as Zeigler-Nichols method are not suitable for these difficult scenarios. Therefore, with the aid of Genetic Algorithms (GAs), Artificial Neural Networks and Fuzzy Logic, many researchers have recently proposed various alternative and intelligent PID controllers (Zhang *et al.*, 2009). Genetic Algorithm is a stochastic search and optimization method that mimics the process of natural evolution (Pillai *et al.*, 2013). The advantage of GA over other popular and efficient optimization algorithm such as Artificial Neural Networks and Fuzzy Logic is it's high convergence (execution) speed (Zhang *et al.*, 2009). The convergence criterion of a genetic algorithm is a user-specified conditions, for example, the maximum number of generations or when the string fitness value exceeds a certain threshold (Ibrahim, 2005).

The Proportional-Integral-Derivative (PID) Controller is widely used in most industrial processes due to their simplicity of operation, ease of design, inexpensive maintenance, low cost, and effectiveness for most linear systems, however, the problem with them is that, they are often poorly tuned. Conventional technique like Zeigler-Nichols method does not give an optimized value for PID controller parameters (Pillai *et al.*, 2013).

In this work, we aim to optimize the PID controller parameters for the terrestrial antenna positioning system using Genetic Algorithm (GA).

## METHODS

### The work flow

MATLAB Genetic Algorithm Toolbox is used to optimize and simulate the system. The work flow for the GA implementation is as shown in Figure 3. These steps are briefly described as follows;

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

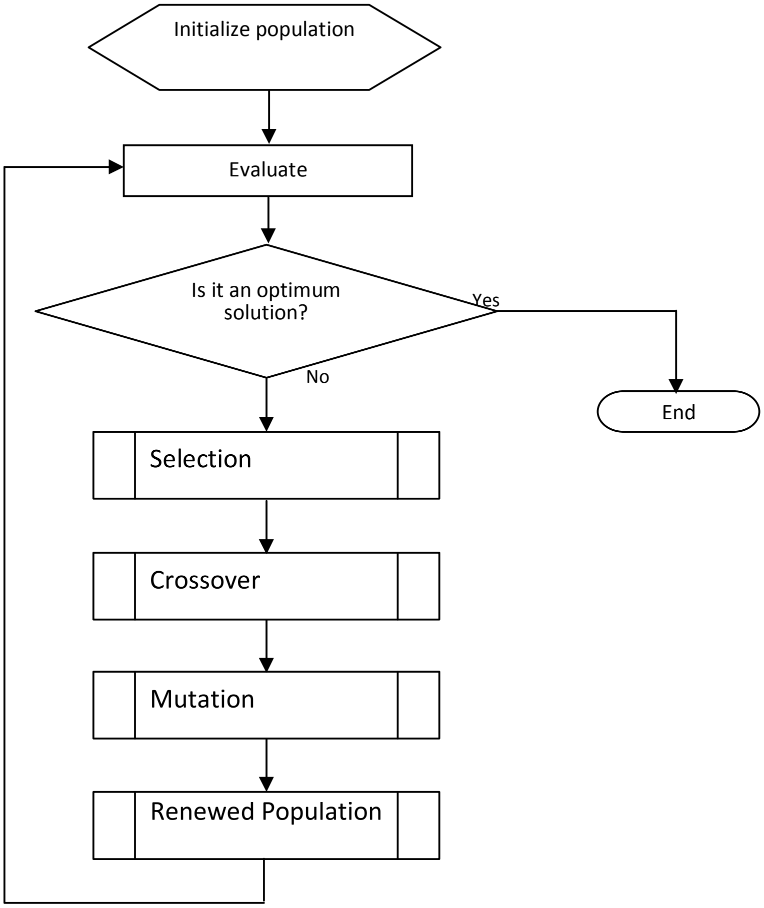

Figure 3: Work flow of Genetic Algorithm (GA)

The codes for the implementation of the GA for this research based on the Genetic Algorithm Optimization Toolbox (GAOT) is given as shown in the appendix:

**Simulation Procedure**

The simulation of the system was done using MATLAB/Simulink environment. The procedures followed are:

1. The GA optimized PID Controller is initialized with a population size of 80 chromosomes and the responses were analysed for different values of amplifier gain, K.
2. The value of K can be found for a stable system by utilizing the Routh-Herwitz criterion. According to this criterion, a system will give stable response if the value of gain K is in the range 0-262 (Chisti *et al.*, 2014). Utilizing this criterion, different amplifier values at interval of 50 was selected for this work.
3. The objective function for use in this research is to find a PID controller that gives the smallest overshoot, fastest rise time and quickest settling time.
4. A probability of 70% (0.7) cross over operation was selected for this work as this gave the best result. Reason being that 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 (Ibrahim, 2005).
5. A mutation probability of 0.2% is selected for this work. Iteration is done 100 times reason being that optimal performance is obtained at this value.

The GA parameters chosen are indicated in Table 1.

Table 1: GA parameters

| S/No. | Parameter | Value |
|---|---|---|
| 1 | Population | 80 |
| 2 | Iteration | 100 |
| 3 | Crossover | 0.7 |
| 4 | Mutation | 0.2 |

## RESULTS AND DISCUSSION

Simulation result for the deep space antenna control system using Matlab/Simulink is shown in table 1, 2 and figures 3 to 7. The GA parameters, plant parameters and the system response values and curves are also shown for the different amplifier's gain values. By utilizing the Routh-Herwitz criterion, different amplifier gains K in the range 50-250 were selected for this work at interval of 50.

Table 2 and Figures 4 to 8, shows that the response of the system at the amplifier gain value of 50 gives the best transient characteristics although with the longest settling time. The system response at amplifier gain of 100 gives a faster settling time and a slower rise time as compared to that at gain of 50. This implies that the system the gain of 100 settles faster and rises slower. However, there is a negligible problem of peaking in its transient response characteristics. At amplifier gain of 150, the system settles faster than that at gains of 50 and 100 and rises faster than that at gain of 100 and slower than that at gain of 50. This implies that it's settling time is smaller and rise time is smaller than that at 100 and higher than that at 50. However, there is a significant distortion in its transient response characteristics - its response pattern is not smooth. The response of the system at gain value of 200 settles faster and rises faster than that at 50, 100 and 150. This implies that it has the smallest settling time and rise time as compared to that at gain value of 50, 100 and 150. However, there is much distortion as it rise. Finally, the response of the system at gain value of 250 settles and rises fastest as compared to the previous gain values. This implies that it has the least settling time and rise time. However, it can be seen from the response graph that it has the biggest distortion while rising. This problem is associated with the increase in the amplifier gain values. The higher the gain value, the bigger the distortion. The overall responses have zero overshoots and undershoot.

Table 2: System response parameter

| | **Parameter** | **Plant Parameter (Amplifier gain)** | | | | |
|---|---|---|---|---|---|---|
| | | **50** | **100** | **150** | **200** | **250** |
| 1 | PID Controller gains $K_P$<br>$K_I$<br>$K_D$ | 0.2862,<br>1.0000<br>0.3063 | 0.0930,<br>1.0000<br>0.2905 | 0.0211,<br>1.0000<br>0.2735 | 0.0374,<br>1.0000<br>0.2073 | -0.1679,<br>1.0000<br>0.0750 |
| 2 | Best cost | 1.4255 | 1.1503 | 0.98395 | 0.87362 | 0.80041 |
| 3 | Rise time | 2.1064sec | 2.2412sec | 2.0182sec | 1.8619sec | 1.6919sec |
| 4 | Settling time | 3.2339sec | 2.9861sec | 2.6281sec | 2.4181sec | 2.1563sec |
| 5 | Overshoot | 0% | 0% | 0% | 0% | 0% |
| 6 | Undershoot | 0% | 0% | 0% | 0% | 0% |
| 7 | Peak | 1.0000sec | 1.0000sec | 1.0000sec | 1.0000sec | 1.0000sec |
| 8 | Peak Time | 4.4407sec | 3.8740sec | 3.3936sec | 3.2930sec | 2.7042sec |

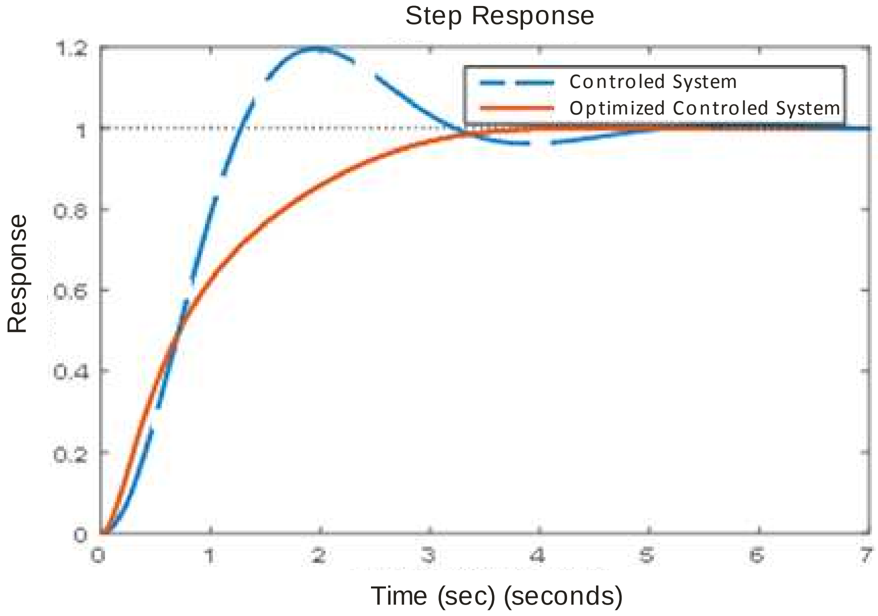

Figure 4: Step Response at gain K = 50

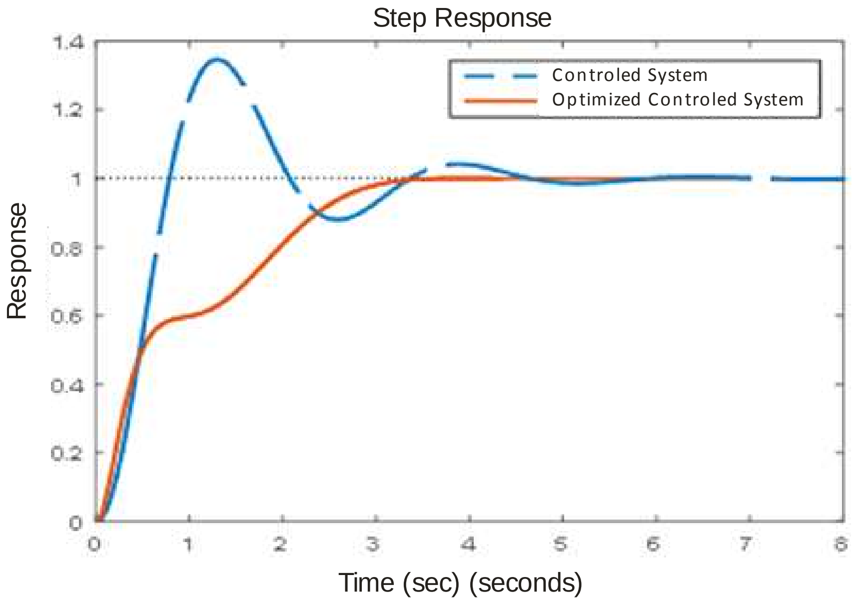

Figure 5: Step Response at gain K = 100

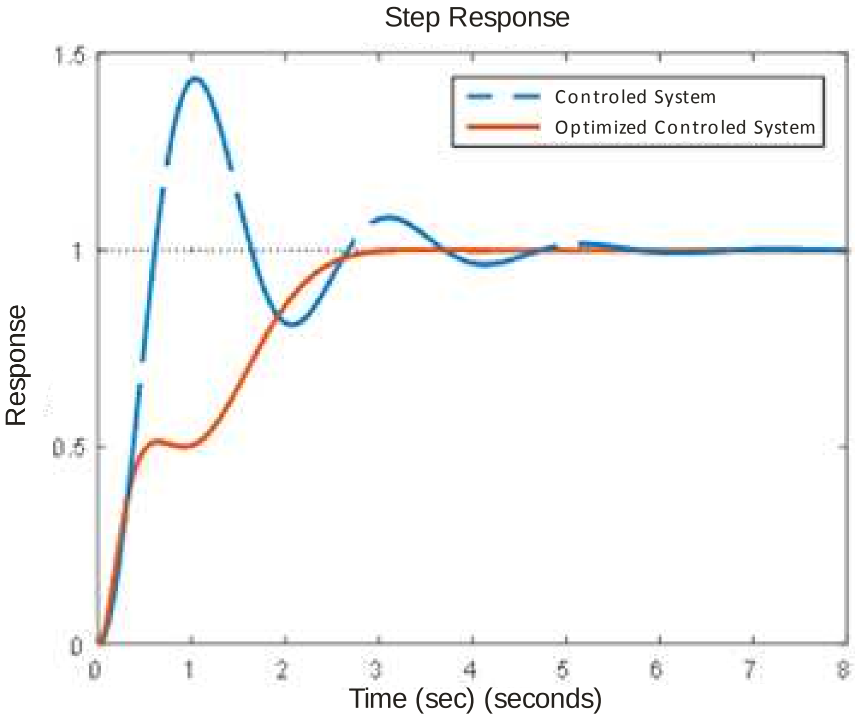

Figure 6: Step Response at gain K = 150

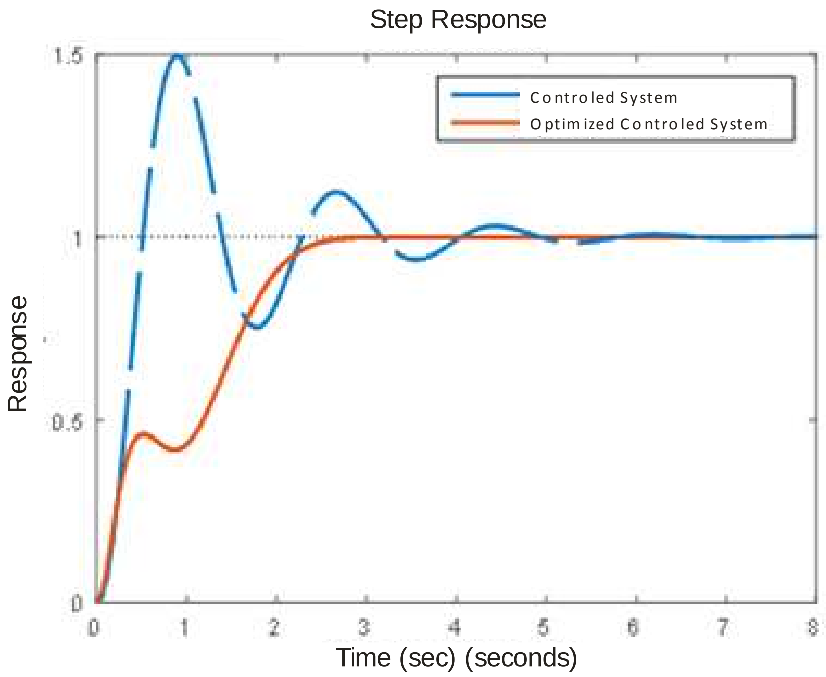

Figure 7: Step Response at gain K = 200

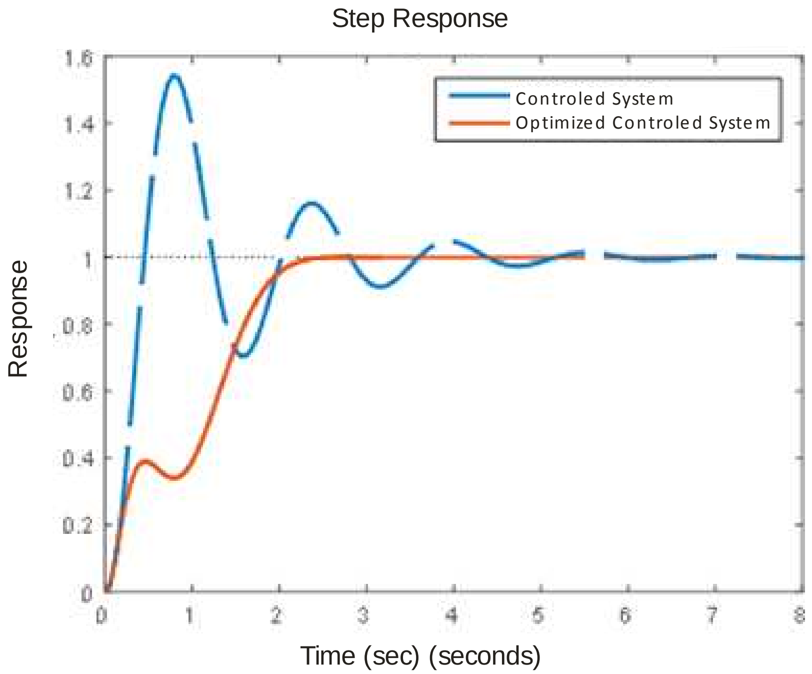

Figure 8: Step Response at gain K = 250

Table 3: Comparison between Ziegler-Nichols method and the G.A. method of tuning at amplifier gain of 100

| S/N | Parameter | | Zeigler-Nichols Method (Chishti *et al.* (2014) | GA Method |
|---|---|---|---|---|
| 1 | PID Controller Gain | $K_P$<br>$K_I$<br>$K_D$ | 16.000<br>2.000<br>5.000 | 0.0930<br>1.0000<br>0.2905 |
| 2 | Rise time | | 0.8568sec | 2.2412sec |
| 3 | Settling time | | 9.2289sec | 2.9861sec |
| 4 | Overshoot | | 66.3812% | 0% |
| 5 | Undershoot | | 23.2614% | 0% |
| 6 | Peak | | 1.6638sec | 1.0000sec |
| 7 | Peak Time | | 0.1689sec | 3.8740sec |

Chishti *et al.* (2014) uses Zeigler-Nichols Method to tune PID with a system response at amplifier gain of 100. The response is tabulated (Table 3) with that of this work on GA tunning method at 100 amplifier gain value. It is obvious that the GA tunning method is better than that of Zeigler-Nichols Method in terms of settling time, overshoot and undershoot, peal and peak time.

## CONCLUSIONS

From the results, it can be concluded that the best response is obtained at an amplifier gain of 250 in terms of transient state and steady state characteristics of the system response as it rises and settles faster than that at 50, 100, 150 and 200. Generally, the system responds faster with increase in amplifier gain values and this increase is responsible for the peaking in its transient state characteristics.

We can also conclude from this work, that the response of the system with Genetic Algorithm tuned PID controller is better than the system response with conventionally tuned PID Controller i.e. Zeigler-Nichols method tuned PID Controller in terms of the transient response, steady state response and stability.

## RECOMMENDATIONS

We are recommending that the gain K used for further work should be of smaller interval so as to get the best value in terms of transient response and stability as well as minimizing the peaking problem associated with higher gain value. It is further recommended that this tunning method be implemented on a real system.

## REFERENCES

Agubor, C., Ndinechi, M. and Opara, R. (2013). A Practical Guide to Site Selection for Communication Antennas and thier Suport Structures in Nigeria. Academic Research International, 4(1), 391.

Astrom, K. J., Hagglund, T. (1995). PID Controllers: Theory, Design and Tuning. 2nd Edition. Society of America, Research Triangle Park.

Chishti, A. R., Bukhari, S. F.-U.-R., Khaliq, H. S., Khan, M. H. and Bukhari, S. Z. H. (2014). Radio Telescope Antenna Azimuth Position Control System Design and Analysis IN MATLAB/SIMULINK using PID and LQR Controller. Universitatea Tehnică Gheorghe Asachi" din Iaşi Tomul LX (LXIV), Fasc. 3-4, 2014 Secţia.

Franklin, G., Powell, F., Emami-Naeini, A. (2002). Feedback Control of Dynamic Systems. 4th Ed., Addison-Wesley Publishing Company.

Gawronski, W. (2008). Modeling and Control of Antennas and Telescopes. Springer Science and Business Media, New York.

Ibrahim, S. M. (2005). The PID Controller Design using Genetic Algorithm. University of Southern Queensland.

Joseph, T. F., Vladimir, O., Slawomir, K. and Adrian, B. (2017). Low-cost Antenna Positioning System Designed with Axiomatic Design. MATEC Web of Conferences 127, 01015 (2017). DOI: 10.1051/matecconf/201712701015. pp 1-7.

Kiam Heong Ang, Gregory Chong and Yun Li (2005). "PID control system analysis, design and technology", IEEE Transactions on Control Systems Technology, vol 13, no 4, pp. 555-576.

Nise, N. S. (2006). Textbook on Control System Engineering. 6th Edition, John Wiley and Sons.

Ogata, K. (2007). Modern Control Engineering. 3rd Edition. Pearson Education, Inc., publishing as Prentice Hall, One Lake Street, Upper Saddle River, New Jersey 07458.

Okumus, H. I., Sahin, E., Akyazi, O. (2012). Antenna Azimuth Position Control with Classical PID and Fuzzy Logic Controllers. IEEE Transaction on Education.

Pillai, R. P., Jadhav, S. P. and Patil, M. D. (2013). Tuning of PID controllers using advanced genetic algorithm. International Journal of Advanced Computer Science and Applications (IJACSA), 1(1), 6.

Prasanna, S., Mauli, S., Arun, Pai., Gayatri, A. (2016). Automatic Antenna Positioning System. International Journal for Scientific Research and Development| Vol. 4, Issue 03, 2016 | ISSN (online): 2321-0613. pg 372-374.

Surya, D. C., Pankaj, R.., Arvind, K., Irshad, A. (2014). Microcontroller Based Wireless Automatic Antenna Positioning System. International Journal of Electronics, Electrical and Computational System IJEECS. ISSN 2348-117X. Volume 3, Issue 6. August 2014 pp 12-26.

Zhang, J., Zhuang, J. and Du, H. (2009). Self-organizing genetic algorithm based tuning of PID controllers. Information Sciences, 179(7), 1007-101.
.

**APPENDIX**

```
function gapidtuning
clc;
clear all;
close all;

%% Defined the Problem parameters and transfer function
s=tf('s'); %Make 's' a transfer function symbol
K=input('Please Provide the value of the gain parameter
(K):= ');
%input('Please Define the Transfer Function of the Plant:=
');
G=6.63*K/(s^3+101.71*s^2+171*s+6.63*K);

%%please defined the plant transfer function
CostFunction=@(x) pid_obj(x,G);
                        % Objective Function Handle for the GA
nVar=3;                 % Number of Variables
VarSize=[1 nVar];       % Size of Variables Matrix
VarMin=-1;              % Lower Bound of Variables
VarMax= 1;              % Upper Bound of Variables
VarRange=[VarMin VarMax];   % Variation Range of
Variables

%% GA Parameters
MaxIt=100;                          % Maximum Number
of Iterations
nPop=input('Please Provide the population Size of Genes:=
');%50;% Population Size
pCrossover=input('Please provide the Percentage of Cross
Over Operator:= ');%0.7;% Crossover Percentage
nCrossover=round(pCrossover*nPop/2)*2;      % Number
of Parents (Offsprings)
pMutation=input('Please Provide the Mutant percentage:=
');%0.2;% Mutation Percentage
nMutation=round(pMutation*nPop);        % Number of
Mutants

%% Initialization
% Empty Structure to Hold Individuals Data
ini_ind.Position=[];
ini_ind.Cost=[];
ini_ind.Out=[];
% Create Population Matrix
pop=repmat(ini_ind,nPop,1);
% Initialize P ositions
for i=1:nPop
   pop(i).Position=unifrnd(VarMin,VarMax,VarSize);
   [pop(i).Cost, pop(i).Out]=CostFunction(pop(i).Position);
end
% Sort Population
pop=SortPop(pop);
% Store Best Solution
BestSol=pop(1);
% Vector to Hold Best Cost Values
BestCost=zeros(MaxIt,1);

%% GA Main Loop
for it=1:MaxIt
   % Crossover
   popc=repmat(ini_ind,nCrossover/2,2);
   for k=1:nCrossover/2
      i1=randi([1 nPop]);
      i2=randi([1 nPop]);
      p1=pop(i1);
      p2=pop(i2);
      [popc(k,1).Position,
popc(k,2).Position]=Crossover(p1.Position,p2.Position,VarR
ange);
      [popc(k,1).Cost,
popc(k,1).Out]=CostFunction(popc(k,1).Position);
      [popc(k,2).Cost,
popc(k,2).Out]=CostFunction(popc(k,2).Position);
   end
   popc=popc(:);
   % Mutation
   popm=repmat(ini_ind,nMutation,1);
   for k=1:nMutation
      i=randi([1 nPop]);
      p=pop(i);
      popm(k).Position=Mutate(p.Position,VarRange);
      [popm(k).Cost,
popm(k).Out]=CostFunction(popm(k).Position);
   end
   % Merge Population
   pop=[pop
       popc
       popm];
   % Sort Population
   pop=SortPop(pop);
   % Delete Extra Individuals
   pop=pop(1:nPop);
   % Update Best Solution
   BestSol=pop(1);
   % Store Best Cost
   BestCost(it)=BestSol.Cost;
   % Show Iteration Information
   disp(['TJ_Itr ' num2str(it) ': Best Cost = '
num2str(BestCost(it))]);

end
   % Plot Step Response
   figure(1);
   step(G);
   hold on
   step(BestSol.Out.T);
   legend('Controled System','Optimized Controled System')
   xlabel('Time (sec)')
   ylabel('Responce')
   stepinfo(BestSol.Out.T)
%% Plots
figure;
plot(BestCost);
xlabel('Time (sec)')
ylabel('Cost Function')
```