# Implementation of Genetic Algorithm for Parameter Tuning of PID Controller in Three Phase Induction Motor Speed Control

Dedid Cahya Happyanto[1] and Ardik Wijayanto[1]

*Abstract*—**Induction motor at low speeds has a tough speed setting sets the width of the range. This study tested by giving the load motor disorders to describe the condition. The method used for vector control system so that the resulting performance is good at setting the motor speed and torque. This method is used in setting the Proportional Integral Derivative (PID) Tuning parameter settings based on Genetic Algorithm (GA) to provide a dynamic response to changes in speed and load torque on the motor, so we get smoothness at any speed change and braking and maximum torque motors. Optimization function is required to obtain a new PID parameter values each input value changes or load disturbances, in terms of the initial determination of these parameters using Ziegler-Nichols method based on frequency response. Tests were performed at a speed of approximately 1800 rpm value rise time of about 10 seconds after generation added, at a rate of 1800 rpm rise time value of the average remains around 9 seconds, but slightly reduced the oscillations in the response, and the speed of approximately 1700 rpm rise time value of the average is 9 seconds. The test results show that GA-based PID controller has a good response in approximately 0.85% overshoot at the motor speed change and braking.**

*Keywords*—**PID controller, Genetic Algorithms, Three Phase Induction Motor**

## I. INTRODUCTION

The induction motor has more benefits because it is robust and relatively cheap. It is also mostly used in electric drive for the constant speed as propulsion electric car, driving in industrial control and no need regular maintenance[1]. However, it also has weakness in complicated speed control. The advance technology in electric drive makes this complication easy to solve.

Drive system used in the industry today many use induction motors. At the first, Electric car was designed by a DC motor drive, and then develop it using a permanent magnet synchronous motor drive. In the next decade an electric car with a three-phase induction motor drive for an induction motor has many advantages, namely a smaller physique, robust and free maintenance [2]. System control was setted in the speed of an induction motor using an intelligent control. The mechanism was setting the control parameters based on determined of vector control [3].

Induction motor has a multivariable nonlinear coupled structure of which its speed is difficulty in control. However, DC motor has a decoupled structure that is easier to control of speed. The control operation of AC motor driver generally needs complicated algorithm implemented in an accurately real time signal process. By using the advance technology of power electronics and electric control, the task for complicated control can be implemented. To running the system, the induction motor should be made linear by Field Oriented Control method. In the development of Field Oriented Control, the method is known having a characteristics that is similar to the DC motor of which the current magnetic field and anchor current are mutually upright[4]. Furthermore, in application, the method of Indirect Field Oriented Control (IFOC) is used to driving an induction motor using GA-based PID controller. In this case to control of speed an induction motor. Genetic Algorithm (GA) combined with conventional PID controller to implement the decoupled system in induction motors [5]. The continuing with this research on fault tolerant control system to achiecve high performance induction motor drive as a drive on industrial control system. In this study discussed how the IFOC applied to an induction motor drive system on indutrial system driving [6] combined with GA-based PID controller.

[1]Dedid Cahya Happyanto and Ardik Wijayanto are with Department of Electrical, Politeknik Elektronika Negeri Surabaya, Surabaya, 60111, Indonesia, E-mail: dedid@eepis-its.edu; ardik@eepis-its.edu.

## II. METHODS

### A. Electric Driving System

#### 1) System Configuration

However, AC motors are nonlinear multivariable structure which coupled, thus setting the pace is more complicated, it is opposite to the DC motor which is a structure that decoupled by setting a more modest pace. Performance control of AC motors driving generally requires a complex control algorithm, implemented with a real time signal processing is fast. Evolution each discipline have contributed to the overall improvement in the technology of electric driving. Therefore, the development of power electronics technology and electric steering is very rapid, and advances in power electronics, microelectronics and microcomputer has allowed the use of supersophisticated control tasks can be realized. Because it must be a linear induction motor by operating through the method of Field Oriented Control (FOC).

Development of methods for the FOC is a method of operating the induction motor which has characteristics alike a DC motors, so that the flow field and flow perpendicular to the anchor. FOC method is a technique to improve vector control methods in the angle between the axis of the fictitious pole stator and rotor. Drive system is used to control the speed and braking of induction motors using a six step SVPWM inverter.

If the condition of the road is uneven, then the electric car should be made braking and motor speed must be reduced by changing the settings on the induction motor controller input. This should be done with changes in turning PID parameters system on the controller, as shown in figure 1.

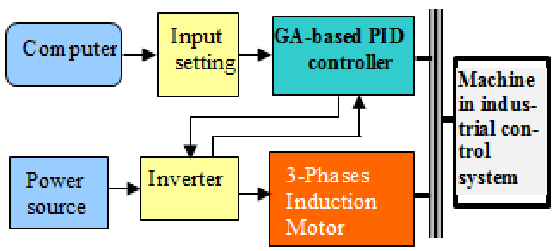



Figure 1. Configuration of electric drive system

*2) Inverter*

Inverters are used to convert the DC voltage source into an AC source, a voltage is generated which can be a constant or a variable value. A voltage source inverter is called an inverter (voltage source inverter) if a constant output voltage while the current source inverter (current source inverter) if a constant output current and a DC inverter relationship variables (variables linked DC inverter) if the output voltage can be controlled or controlled larger and smaller than the input voltage. Inverter is used to control the three-phase induction motor at the end of this project is a three-phase inverter SYSDRIVE 3G3MV Omron type. The configuration of the inverter as shown in figure 2.

*B. The Speed Controller*

*1) Three Phase Induction Motor Model*

A motor works on the basis of induction process in the rotor part. When the current flows in the rotor coil, it creates induction that is caused by the differences between the rotor rotation and the field of stator, created by the static coils.

Electromagnetic torque ($T_e$) is the function of stator current and rotor current, such as:

$$T_e = pM\,(i_{dr}\, i_{qs} - i_{qr}\, i_{ds}) \quad (1)$$

notes :

- M = coupled induction ( H)
- $i_{ds}$ = Stator currenton axes d (A)
- $i_{dr}$ = Rotor current on axes d (A)
- $i_{qs}$ = Stator currenton axes q (A)
- $i_{qr}$ = Rotor current on axes q (A)

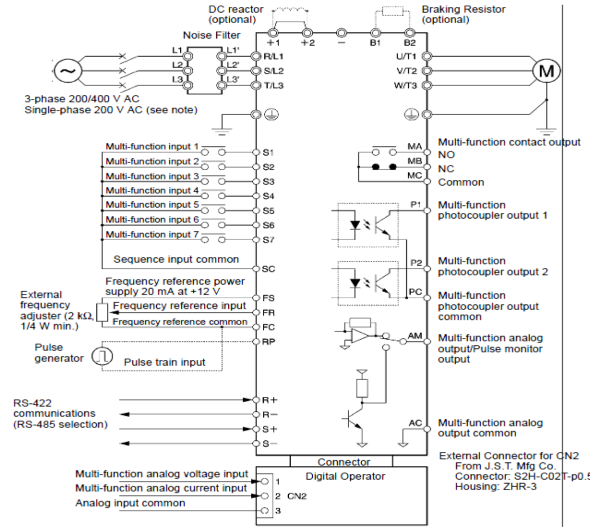



Figure 2. Omron Sysdrive Inverter System [1]

Rotor rotation speed is the function from electromagnetic torque, and load torque such as:

$$\frac{J}{p}\frac{d}{dt}\omega_r + K_g\,\omega_r = T_e - T_l \quad (2)$$

*2) Genetic Algorithms*

In general, there are several steps in the process of Genetic Algorithm. Here are the steps:

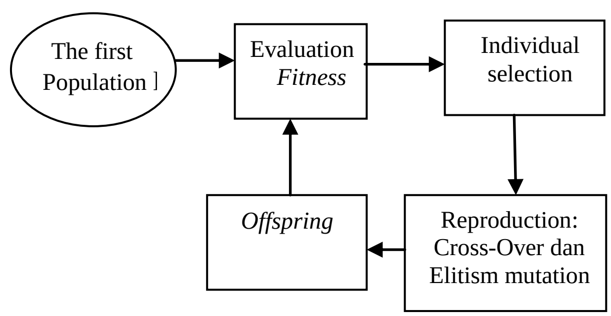



Figure 3. General genetic algorithm cycle

In general, the description of figure 3 above as follows :

a. Generating the initial population of a few individuals at random or through certain procedures. For population size depends on the problem to be solved and the type of genetic operator will be implemented. Once the population size is determined, then made the generation of the initial population. The terms of which must be carried out to demonstrate the solution should really be considered in the generation of each individual. Techniques in the generation of the initial population, there are several ways, including the following :

   i. Random Generator

      The essence of this method is to involve the generation of random numbers for each gene according to the value chromosome representation is used .

ii. Specific approaches ( insert particular value into the gene). This way by inserting a gene into a specific value of the initial population is formed.

iii. Gen permutation

One permutation of genes in the generation of the initial population is the use of Josephus permutation.

b. Evaluate the suitability of each individual state to the desired result , which is called the fitness evaluation. This process aims to obtain the highest value of fitness.

c. Selection is used to obtain individuals which will be selected for crossbreeding and mutation. The higher the fitness value of an individual, the greater the chance of being selected . There are several methods of selection, as follows :

i. Roulette Wheel selection with

The roulette wheel selection method is the simplest method and is often known as stochastic sampling with replacement . The workings of this method are as follows :

- Calculated fitness value of each individual (fn where i is the individual from - 1 to - n ).
- Calculated total fitness of all individuals.
- Calculated probability of each individual.
- From these probabilities, calculated ration respectively indvidu the numbers 1 to 100.
- Raised random numbers between 1 to 100.
- From the generated random numbers, which determined individuals chosen in the selection process.

ii. Selection with tournament

At the tournament selection method, applied a value-indivdu tour for individuals selected at random from a population. Best individuals in this group will be selected as a parent. The parameters used in this method is the size of the tour is worth between 2 to N ( the number of individuals in a population ).

d. Reproduce the next process, which entered into a cross between individuals elected interspersed mutation. Reproductive success is divided into two, namely :

i. Moving Cross (Crossover)

Crossbreeding ( crossover ) is the operator of a genetic algorithm that generates two stem to form a new chromosome. Crossing over produces new point in the search space that is ready to be tested. This operation is not always done on all existing individuals. Individuals were randomized to do with Pc crossing between 0.6 s / d 0.95. If the crossover is not done, then the value of the parent will be passed down to offspring. Principle of this crossover is doing operations (exchanges, arithmetic) in thehe corresponding genes of two parents to produce a specified probability of crossover. This crossover operator depends on the chromosome representation is done.

Flowchart in Figure 4 above where the crossover operator is the operator of a genetic algorithm that involves two parents in the population to form a new individual. Crossover is not always done on all individuals. So individuals chosen at random for each crossed in accordance with the probability of crossover (Pc), as shown in figure 5.

ii. Mutation

The next operator in the genetic algorithm is the gene mutation. This operator serves to replace the missing genes from the population due to the selection process that allows the re-emergence of genes that do not appear in the initial population. Chance of mutation (Pm) is defined as the percentage of the total number of genes there are populations that have mutations. Illustrated in Figure 5 on the work of the mutation operator in genetic algorithms. Starting from the selected individual then the individual mutated by adding a very small random values (mutation step size), the probability of mutation.

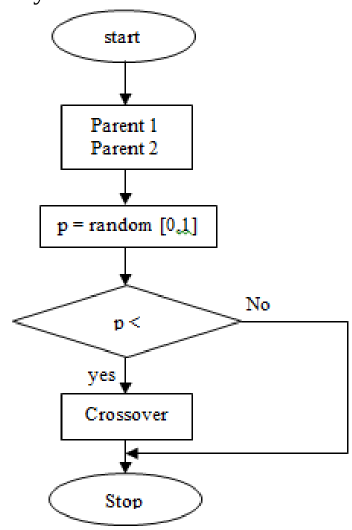



Figure 4. The Flowchart of Crossover [6]

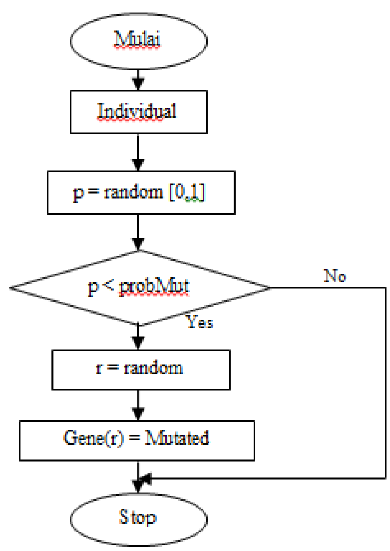



Figure 5. Flowchart of Mutation

- Repeat steps 2-4 until you find the desired individual (such as the crossover process).
- Elitism process

When a new population is created through crossover and mutation, there is a chance that the best chromosome of a generation will be lost in the next generation. For that we need a method to preserve the best chromosome. This method is called elitism. Elitism a copy (or several) best chromosomes to the new population. The rest is done in the old way. Elitism can improve the performance of GA quickly, because it can prevent the loss of the best solution discovery.

Block diagram in figure 6 is a process where PID control parameters are searched by GA method, where the target speed input a given reference speed. This reference input as target data where the output of the plant is the action performed by the PID control parameters where the value obtained from the GA later in feedback it back to back comparison with a reference input. Results of the comparison will generate a control signal which is processed by the system error control back. Results of processing the error values will produce the output value of the PID controller where the value of the parameters sought by the GA method.

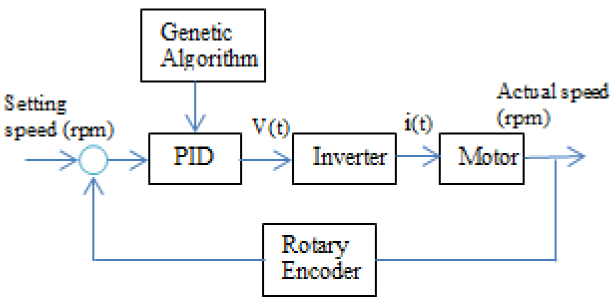



Figure 6. Diagram block of overal control system

The existence of genetic algorithms in process control system here is to improve the search process of tuning the parameters of PID. Genetic Algorithms ability to provide a model which, although not the best solution but is more likely to be accepted .

Process that will be conducted in the end of the project of which this is generating the initial population, evaluate the value of fitness, individual selection, reproduction, and produce a new population. Here are the steps in the process of tuning the parameters of PID constants by the method of off-line genetic algorithms, which include :

i. Chromosome Initialization

At the beginning of the initialization process of the genetic algorithm chromosome , where chromosomes at the end of this project is the solution to the formation of the PID value is the value of Kp , Ki , Kd. For the first chromosome initialization is done to determine the length of the chromosome. As at the end of the project in user using genetic algorithm specification as follows :

TABLE 1.
PARAMETER OF GENETIC ALGORITHM SPESIFICATION

| Parameter | Low level | High level |
|---|---|---|
| $K_p$ | 0 | 5 |
| $K_i$ | 0 | 0.1 |
| $K_d$ | 0 | 0.1 |

Generating random initial population, where the value of the genes according to the defined limits .

Specifies the number of population is 6, then :
Chromosome [1] = [Kp;Ki;Kd] = [2.5;0.025;0.075]
Chromosome [2] = [Kp;Ki;Kd] = [3;0.012;0.035]
Chromosome [3] = [Kp;Ki;Kd] = [4.2;0.014;0.045]
Chromosome [4] = [Kp;Ki;Kd] = [1;0.05;0.05]
Chromosome [5] = [Kp;Ki;Kd] = [2;0.032;0.015]
Chromosome [6] = [Kp;Ki;Kd] = [1.2;0.01;0.01]

ii. Evaluation of Chromosome and fitness function.

The fitness function is used to see where the most fit chromosome. Fitter chromosomes will have a chance to survive the next genarasi process. Fitness value sought by the characteristics of the plant system. The model will be a street or rules for the genetic algorithm to search the optimization parameter values Kp, Ki, and Kd. From the evaluation of chromosomes, the fitness function at the end of this project are as follows :

$$m(t) = Kp\ e(t) + Ki \int e(t)dt + Kd \frac{de(t)}{dt}. \quad (3)$$

In the process of generating the population, obtained as follows :
Chromosome[1] = [Kp;Ki;Kd] = [2.5;0.025;0.075]
Chromosome[2] = [Kp;Ki;Kd] = [3;0.012;0.035]
Chromosome[3] = [Kp;Ki;Kd] = [4.2;0.014;0.045]
Chromosome[4] = [Kp;Ki;Kd] = [1;0.05;0.05]
Chromosome[5] = [Kp;Ki;Kd] = [2;0.032;0.015]
Chromosome[6] = [Kp;Ki;Kd] = [1.2;0.01;0.01]

Of the value of the chromosome is inserted in the formula (3) to determine the fitness value of each chromosome. Fitness value of each chromosome is above is as follows :
Fitness[1] = 1/bobot = 0.000475
Fitness [2] = 1/bobot = 0.000302
Fitness [3] = 1/bobot = 0.000197
Fitness [4] = 1/bobot = 0.00085
Fitness [5] = 1/bobot = 0.000953
Fitness [6] = 1/bobot = 0.008025
Total of fitness = 0.000475 + 0.000302 + 0.000197 + 0.00085 + 0.000953 + 0.008025
= 0.010802

Formula to find the probability P[i] = fitness [1]/total fitness
P[1] = 0.000475/0.010802 = 0.043988
P[2] = 0.000302/0.010802 = 0.027932
P[3] = 0.000197/0.010802 = 0.018274
P[4] = 0.00085/0.010802 = 0.07866
P[5] = 0.000953/0.010802 = 0.088242
P[6] = 0.008025/0.010802 = 0.742904

From the calculation of the probability of fitness above where the largest fitness value on chromosome 6. This enables the next generation of chromosome 6 likely to be selected again.

iii. The Roulette Wheel Selection

Selection process using roulete wheel. In this process relies on random values generated. This selection provides an opportunity for the parent who has the greatest fitness to continue in the next process.

For the first find the value of the cumulative probability of fitness :
Cumulative[1]=0.043988
Cumulative[2]=0.043988+0.027932= 0.07192
Cumulative[3]=0.043988+0.027932+0.018274
= 0.090194
Cumulative[4]=0.043988+0.027932+0.018274
+0.07866 = 0.168854
Cumulative[5]=0.043988+0.027932+0.018274
+0.07866 + 0.088242 = 0.257096
Cumulative[6]=0.043988+0.027932 +0.018274
+ 0.07866 + 0.088242 + 0.742904 = 1

Having calculated the value of the cumulative probability of selection process can be done using the roulette wheel. The process is to generate random number R in the range 0-1. In this process if R[k] < cumulative [1] then select chromosome 1 as a parent, besides selecting chromosome k as a parent with cumulative provisions [k - 1] < R < cumulative [k]. then rotate the roulette wheel as much as 6 times the total population (generating a random number R) and every round selected one chromosome for a new population. Generating random numbers :
R [1] = 0.0332
R [2] = 0.0675
R [3] = 0.0065
R [4] = 0398
R [5] = 0.501
R [6] = 0822

Of random numbers that have been raised, then the new chromosome results of the selection process :
Chromosome [1] = chromosome [1]
Chromosome [2] = chromosome [2]
Chromosome [3] = chromosome [1]
Chromosome [4] = chromosome [3]
Chromosome [5] = chromosome [4]
Chromosome [6] = chromosome [6]
New chromosome selection process results :
chromosome[1] = [Kp;Ki;Kd] = [2.5;0.025;0.075]
chromosome[2] = [Kp;Ki;Kd] = [3;0.012;0.035]
chromosome[3] = [Kp;Ki;Kd] = [2.5;0.025;0.075]
chromosome[4] = [Kp;Ki;Kd] = [4.2;0.014;0.045]
chromosome[5] = [Kp;Ki;Kd] = [1;0.05;0.05]
chromosome[6] = [Kp;Ki;Kd] = [1.2;0.01;0.01]

iv. Crossing over (Crossover)

Crossing over (crossover) performed on chromosome escapes roulette wheel selection process. In this process two chromosomes are taken sequentially to do the process of interbreeding. The number of chromosomes that are produced is even, for crossbreeding process. In this study used a crossover probability value was 75%. This means that the value of random chromosome less than the value of Pc (crossover probability) will undergo crossover process. Beginning the process of crossover that generate random numbers R 0-1 as follows :
R[1] = 0.191
R[2] = 0.859
R[3] = 0.760
R[4] = 0.862
R[5] = 0.458
R[6] = 0.006

Thus, chromosome k to be selected as a parent if R [k] <Pc, random number R from above, which is used as the parent chromosome[1], chromosomes[5], chromosomes[6].

The following process crossovers :
Offspring[1] chromosomes[1] X chromosomes[6]
= [2.5;0.025;0.075] X [1.2;0.01;0.01]
= [2.5;0.025;0.01]
Offspring[5] chromosomes[5] X chromosomes[6]
= [1;0.05;0.05] X [1.2;0.01;0.01]
= [1;0.05;0.01]
Offspring[6]=chromosomes[6] X chromosomes[1]
= [1.2;0.01;0.01] X [2.5;0.025;0.075]
= [1;0.025;0.075]

A new population after experiencing a crossover process, as bellow :
chromosomes[1] = [Kp;Ki;Kd] = [2.5;0.025;0.01]
chromosomes[2] = [Kp;Ki;Kd] = [3;0.012;0.035]
chromosomes[3]= [Kp;Ki;Kd] = [2.5;0.025;0.075]
chromosomes[4]= [Kp;Ki;Kd] = [4.2;0.014;0.045]
chromosomes[5] = [Kp;Ki;Kd] = [1;0.05;0.01]
chromosomes[6] = [Kp;Ki;Kd] = [1;0.025;0.075]

v. Mutation

Mutation process is done by replacing one gene selected at random by a new value obtained at random . The probability of mutation ( Pm ) at the end of the project is 5 % . Mutation process is as follows :
Total gene = (gene number in chromosom)*Number of population
=3*6
Total gene =18
Number of Mutation = 0.05*18 = 0.9 =1

Number of genes that are mutated there is 1 gene, wherein the position of randomly selected genes 5 selected genes gene, then the gene will be replaced :
chromosome[1] = [Kp;Ki;Kd] = [2.5;0.025;0.01]
chromosome[2] = [Kp;Ki;Kd] = [3;0.01;0.035]
chromosome[3] = [Kp;Ki;Kd] = [2.5;0.025;0.075]
chromosome[4] = [Kp;Ki;Kd] = [4.2;0.014;0.045]
chromosome[5] = [Kp;Ki;Kd] = [1;0.05;0.01]
chromosome[6] = [Kp;Ki;Kd] = [1;0.025;0.075]

Flow chart in Figure 7 shows the flow of the process in which the Genetic Algorithm starts from defining the problem to be resolved is where the PID constants offline search. Begin initialize chromosomes of the problems will be solved. Then evaluate the chromosomes that have been raised to seek fitness function. Having known her fitness function, the next process is the roulette wheel selection process. From the results of the roulette wheel selection process will be obtained chromosome is selected to proceed to the next process. In the crossover process predefined crossover probability (Pc) is 75 %, if the random number is less than Pc was raised then individuals selected for crossover. The process of mutation is to change the value of a gene on a chromosome. Beginning the process of mutation which determines probability of mutation (Pm) is 5 %, so for the sum of all gene mutations multiplied by Pm the result is the number of genes to be mutated. After all the process is exceeded it can be seen that the best chromosome will be the best solution to this quandary. If not found the best solution, then the process will be repeated again from chromosomal evaluation. Once we define the genetic representation and the ability to function, genetic algorithms will process the random initialization of population settlement, and improve application through repetition with applications mutation operators, crossover and selection. The more sampling is taken it will be more rapid genetic algorithms to find the best solution chromosome.

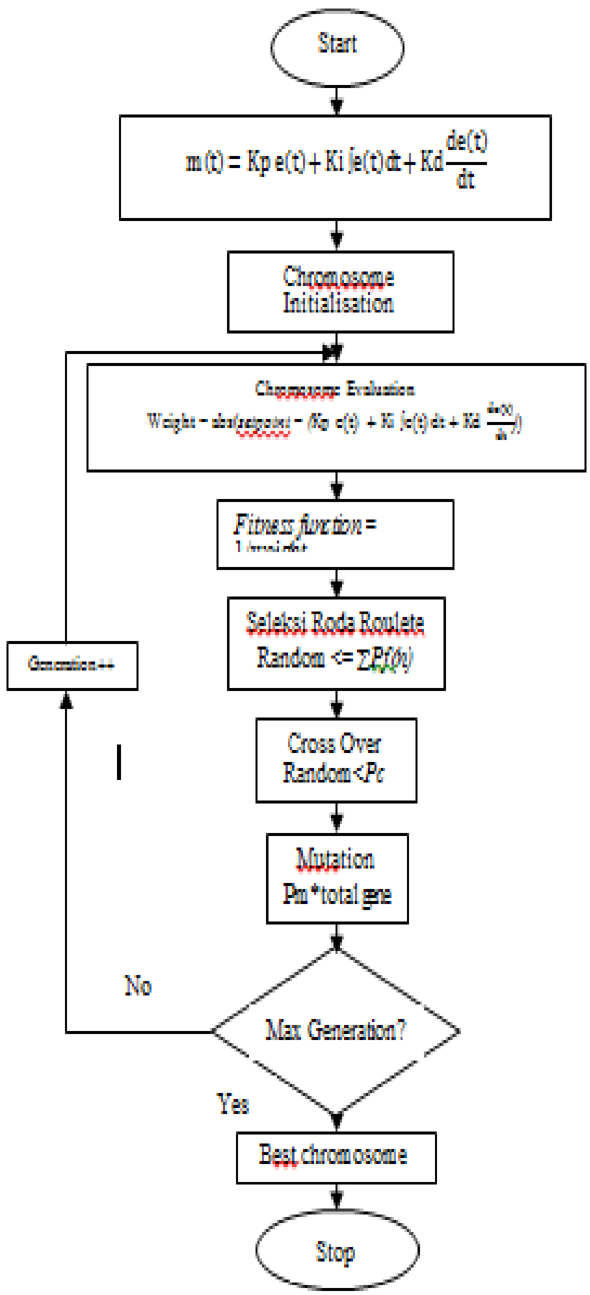



Figure 7. Flowchart of Genetic Algorithm

The testing process is done by giving a frequency of 60 Hz and then seen the maximum motor speed response. Then look at the maximum voltage of the inverter with a voltmeter. At this stage, making test program to try hardware induction motor in which the program will be included in the PIN FR and FC in the inverter. After that is done by replacing the value of the incoming voltage into the inverter which uses a scale of 0.1 volts.

## III. Results and Discussion

The following is a response to the test results of three-phase induction motor is performed by applying a voltage to the inverter 5 volt Omron 3G3MV SYSDRIVE derived from the PID output voltage is controlled by a microcontroller. OCR1A voltage magnitude using 10-bit PWM has a maximum value of 1023. OMRON SYSDRIVE 3G3MV, the inverter can be controlled locally or remotely, depending on the settings. If the inverter to be controlled from the outside, can be done by setting its mode. Can also use the remote control voltage between 0V-5V. For more convenience, the remote setting is done with the input voltage, this is because the maximum output of the microcontroller is 5V. If the output is modulated by PWM, the resulting output will vary from 0V to 5V.

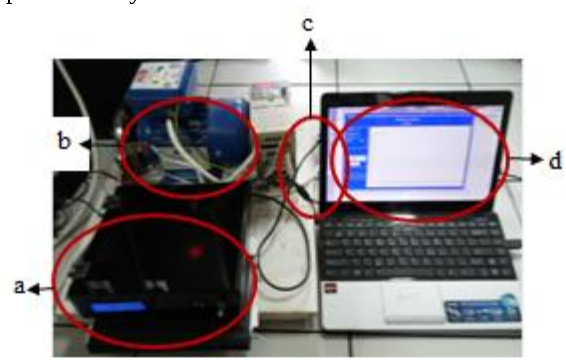



Figure 8. Testing Module of induction motors

Note :
a → Controller
b → 3-phases induction motors
c → 3-phases Inverter
d → GUI for motor speed monitoring

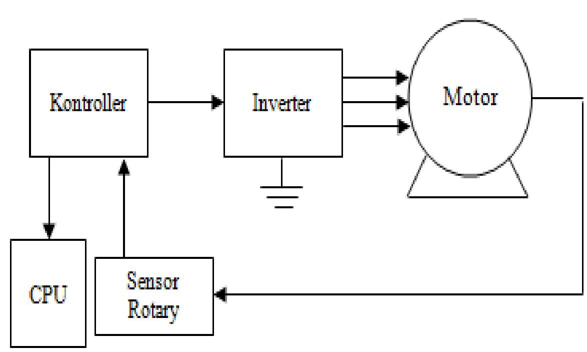



Figure 9. Three-phases induction motors testing diagram

Figure 10. Respon Graph induction motor at 1792 PRM

At the time of maximum speed, show the frequency of 60 Hz inverter, input voltage is measured at the inverter 5Volt . The resulting motor speed is 1792 Rpm as shown in Figure 10. From the graph it can be seen that the response rise time is 10 seconds, then 10 seconds settling time. Ess % (Error steady state) to about 0.44%.

To further test the motor output is done to give the incoming voltage to the inverter and find out how much motor speed changes that occur for different voltage values where the scale used for each change of the voltage is 0.1 volts.

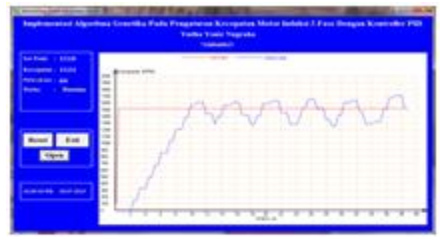
TABLE 2.
TESTING RESULT INDUCTION MOTOR 1
RELATIONSHIP BETWEEN THE INVERTER INPUT VOLTAGE, FREQUENCY , RPM

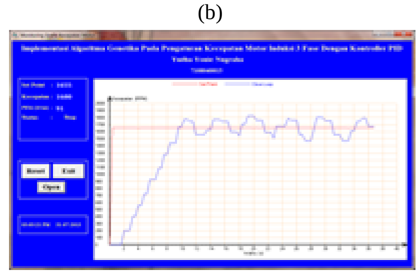
| V in | Frek. | RPM | V in | Frek. | RPM |
|---|---|---|---|---|---|
| 0.0 | 0.00 | 0 | 2.0 | 25.12 | 832 |
| 0.1 | 0.00 | 0 | 2.1 | 27.32 | 882 |
| 0.2 | 1.98 | 97 | 2.2 | 29.47 | 930 |
| 0.3 | 3.33 | 142 | 2.3 | 31.57 | 1027 |
| 0.4 | 4.47 | 202 | 2.4 | 32.77 | 1080 |
| 0.5 | 5.98 | 247 | 2.5 | 33.84 | 1110 |
| 0.6 | 7.72 | 292 | 2.6 | 35.15 | 1185 |
| 0.7 | 8.12 | 315 | 2.7 | 37.85 | 1230 |
| 0.8 | 9.24 | 382 | 2.8 | 39.25 | 1290 |
| 0.9 | 11.92 | 412 | 2.9 | 41.32 | 1312 |
| 1.0 | 12.25 | 427 | 3.0 | 42.01 | 1357 |
| 1.1 | 13.71 | 442 | 3.1 | 42.97 | 1387 |
| 1.2 | 14.47 | 502 | 3.2 | 43.80 | 1425 |
| 1.3 | 15.42 | 547 | 3.3 | 44.67 | 1467 |
| 1.4 | 16.30 | 585 | 3.4 | 46.34 | 1497 |
| 1.5 | 18.24 | 622 | 3.5 | 47.38 | 1507 |
| 1.6 | 19.74 | 667 | 3.6 | 48.05 | 1522 |
| 1.7 | 20.03 | 697 | 3.7 | 49.01 | 1560 |
| 1.8 | 22.25 | 707 | 3.8 | 50.23 | 1582 |
| 1.9 | 23.17 | 780 | 3.9 | 51.33 | 1602 |

From the above data, there is a difference between the motor rotation speed of each input frequency, this can lead to difficulty eliminating the steady state error.

In this test done with two speed control system which uses PID with Genetic Algorithm Method with 20 generations. The following is the result of the test.


(a)


(b)


(c)

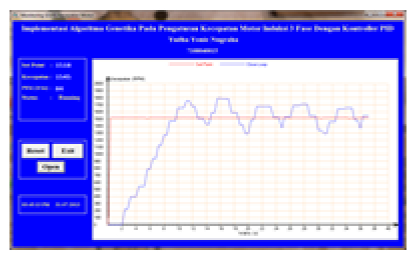

(d)

Figure 11. Graphs the response speed
( a) 1800 Rpm, ( b ) 1700 Rpm, ( c ) 1600 Rpm, ( d ) 1500 Rpm

In Figure 11(a) where the setpoint 1800 RPM rise time reaches 10 seconds with about 5 % Ess . In Figure 11(b) where the setpoint 1700 Rpm rise time reaches 10 seconds with about 5 % Ess. In Figure 11(c) where the setpoint of 1600 Rpm rise time reaches 10 seconds by about 6 % ess . In Figure 11(d) where the setpoint 1500 Rpm rise time reaches 10 seconds with ess about 7 % of the testing that has been done, the system shows unstable at 1800 Rpm setpoint taken. The smaller setpoint are given, the more unstable the system to achieve the target performance parameters has been done.

Generational change in the operation of Genetic Algorithm influential enough to fix the system is not yet stable. However, the weakness in the system that have been made at the end of this project is to improve the weaknesses of the system in the event of an error due to the system that has made the study created offline. This makes the system unstable. Then sampling to run the control systems too little learning system can not be made for a generation that carried on the Genetic Algorithm is still lacking, this result allows chromosomes derived chromosome is still not good enough or have not been the best fit of the Three-phase induction motor plant.

## IV. CONCLUSION

This research has been done on the control of Three-phase induction motor using PID controller with a genetic algorithm method for offline to get the three control parameters Kp, Ki, and Kd. From the results of the testing that has been done can be concluded as follows :

In the PID control testing Ess average around 22%, in testing the PID control system in which the value of the parameter search using Genetic Algorithms Ess average of about 11%.

In testing the maximum speed is 1792 Rpm, system shows good response, but it will be brought to the speed of system instability.

Characteristics of induction motors in industrial machinery shows to reach a maximum speed (1792 Rpm) it takes about 10 seconds.

If the set point is smaller than 1800 rpm occurs less stable because it is affected by the impedance is the resistance (ohm) of an induction motor and the motor frequency (Hz) given on the motor.

## REFERENCES

[1]. Ali Emadi, Young Joo Lee, and Kaushik Rajashekara, Power Electronics and Motor Drives in Electric, Hybrid Electric, and Plug-In Hybrid Electric Vehicles, IEEE Transactions On Industrial Electronics, Vol. 55, no. 6, June 2008.

[2]. Ali Mohamad Bazzi, Designing Better Induction Motor Drive Systems, From Efficiency, Reliability, And Power Electronics Perspectives, University of Illinois at Urbana-Champaign, 2010, pp. 66.

[3]. Dedid Cahya H, Agus Indra G, Rusminto Tjatur W, Tianur, Kontrol Kecepatan Motor Induksi Menggunakan PID-Fuzzy, Jurusan Teknik Elektronika, Politeknik Elektronika Negeri Surabaya, 2010

[4]. M.E.H. Benbouzid, D. Diallo, and M. Zeraoulia, “Advanced fault-tolerant control of induction motor drives for EV/HEV traction applications: From conventional to modern and intelligent control techniques,” IEEE Trans. Vehicular Technology, vol. 56, pp. 519-528, March 2007.

[5]. Mariam Khan and Narayan C. Kar, Performance Analysis of Fuzzy Based Indirect Field Oriented Control of Induction Motor Drives for Hybrid Electric Vehicles, PHEV2007 Conference, Canada, Novemver 1&2, 2007

[6]. Griffin, Ian. On-line PID Controller Tuning using Genetic Algorithms;2003

[7]. Neenu Thomas, Dr. P. Poongodi, Position Control of DC Motor Using Genetic Algorithm Based PID Controller, Proceedings of the World Congress on Engineering Vol II, London U.K;2009

[8]. Benoit Robyns, F. Berthereau, Jean-Paul Hautier dan Herve Buyse , Multimodel field orientation in an indirect field oriented control (IFOC). IEEE Transactions on Industrial Electronics, Vol. 47, No. 2, April 2000.

[9]. Demba Diallo, Member, IEEE, Mohamed El Hachemi Benbouzid, Fault tolerant control system high performance induction motor drive for the electric vehicle, IEEE Transaction On Vehicular Technology, Vol.53,No.6 November 2004

[10]. Zulkarnain Lubis, A.N. Abdalla, Mortaza, Mathematical Modeling of the Three Phase Induction Motor Couple to DC, American J. of Engineering and Applied Sciences, 2009, pp. 715-719.