CONTROL OF AUTOMATIC VOLTAGE REGULATOR SYSTEM USING PID CONTROLLER WITH OPTIMAL TRANSIENT RESPONSES

LIEVYIA AMRITHA A/P BALAKRISHNAN

BACHELOR OF ELECTRICAL ENGINEERING WITH HONOURS
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

CONTROL OF AUTOMATIC VOLTAGE REGULATOR SYSTEM USING PID CONTROLLER WITH OPTIMAL TRANSIENT RESPONSES

LIEVYIA AMRITHA A/P BALAKRISHNAN

**A report submitted
in partial fulfilment of the requirements for the degree of
Bachelor of Electrical Engineering with Honours**

Faculty of Electrical Technology and Engineering

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

DECLARATION

I declare that this thesis entitled **"CONTROL OF AUTOMATIC VOLTAGE REGULATOR SYSTEM USING PID CONTROLLER WITH OPTIMAL TRANSIENT RESPONSES"** is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in the candidature of any other degree.

Signature :

Name : LIEVYIA AMRITHA A/P BALAKRISHNAN

Date : 15/6/24

APPROVAL

I hereby declare that I have checked this report entitled **"CONTROL OF AUTOMATIC VOLTAGE REGULATOR SYSTEM USING PID CONTROLLER WITH OPTIMAL TRANSIENT RESPONSES",** and in my opinion, this thesis fulfils the partial requirement to be awarded the degree of Bachelor of Electrical Engineering with Honours

Signature	:	
Supervisor Name	:	TS. DR. HAZRIQ IZZUAN BIN JAAFAR
Date	:	25.06.2024

DEDICATIONS

To my respected parents, supervisor, and beloved friends.

ACKNOWLEDGEMENTS

First of all, I would like to convey my gratitude and special thanks to my supervisor, Ts. Dr. Hazriq Izzuan Bin Jaafar who offered me this wonderful opportunity to work on this project. Besides, I would like to thank my supervisor for his guidance and support throughout this project. Over the course of this project, I learned and gained new knowledge in control system techniques and skills that can be applied in MATLAB Simulink.

Secondly, I would like to express my thanks to our Universiti Teknikal Malaysia Melaka (UTeM) for providing the journals in library which have helped me in my research about my project .

Finally, I express my appreciation to my beloved family, friends and everyone else who have supported, helped and motivated me on this project.

ABSTRACT

This research focuses on the optimal transient response of Automatic Voltage Regulator (AVR) in electrical systems. The AVRs are vital components of present-day electrical systems, assuring consistent voltage output, protecting equipment, improving power quality, and optimizing generator performance. Transient response refers to a dynamic system's initial behaviour after a quick change or disruption in its input. The analysis is carried out using an AVR model through MATLAB/Simulink. The behaviour of the AVR system’s transient response without a controller is one of the primary objectives of this research. Then conventional tuning methods and Particle Swarm Optimization PSO tuning method are used on AVR model using MATLAB/Simulink. The conventional tuning methods which are used is in this research is Trial-and-Error method and Ziegler-Nichols. The PSO tuning procedure was done by simulating three iterations, each iteration was simulated ten times. The AVR system without PID, conventional tuning methods, and PSO tuning methods were compared in order to determine which produced the best results in terms of peak time, rising time, settling time, overshoot, and steady-state error. Overall, the study emphasises the need of improving AVR systems for better stability, power quality and generator performance in current electrical applications.

ABSTRAK

Penyelidikan ini memberi tumpuan kepada respons transisi optimum sistem Pengawal Voltan Automatik (AVR) dalam sistem elektrik. AVR adalah komponen penting dalam sistem elektrik hari ini, memastikan output tegangan yang konsisten, melindungi peralatan, meningkatkan kualiti kuasa, dan mengoptimumkan prestasi generator. Respon transisi merujuk kepada tingkah laku awal sistem dinamik selepas perubahan cepat atau gangguan dalam inputnya. Analisis dilakukan menggunakan model AVR melalui MATLAB/Simulink. Tujuan penyelidikan ini memberi tumpuan kepada tingkah laku tindak balas sementara dalam kehadiran tanpa kawalan untuk sistem AVR. Tujuan kedua ialah untuk menggunakan kaedah tuning konvensional dan kaedah Algorithm Pengoptimuman Keremunan Zarah PSO pada model AVR menggunakan MATLAB/Simulink. Kaedah tuning konvensional yang digunakan dalam kajian ini ialah kaedah Trial-and-Error (T-E) dan Ziegler-Nichols (ZN). Prosedur tuning PSO dilakukan dengan mensimulasikan tiga iterasi, setiap iteration disimulasi sepuluh kali. Sistem AVR tanpa PID, kaedah tuning konvensional, dan kaedah PSO dibandingkan untuk menentukan mana yang menghasilkan hasil terbaik dalam hal respons sementara. Penyelidikan ini memberi tumpuan kepada masa sistem yang meningkat, masa penetapan, kelebihan, amplitudo puncak, dan kesilapan keadaan stabil. Secara keseluruhan, kajian ini menekankan keperluan untuk meningkatkan sistem AVR untuk lebih baik kestabilan, kualiti kuasa dan prestasi generator dalam aplikasi elektrik semasa.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

AVR	Automatic Voltage Regulator system
PID	Proportional Integral Derivative
FLC	Fuzzy-Logic controller
TCGA	Taguchi Combined Genetic Algorithm
PSO	Particle Swarm Optimization
MOL	Many Optimizing Liaisons
ABC	Artificial Bee Colony
CAS	Chaotic Ant Swarm
GA-BF	Genetic Algorithm and Bacteria Foraging
LUSO	Local Unimodal Sampling Optimization
TLBO	Teaching Learning Based Optimization
ZN	Ziegler-Nichols
$pbest$	Personal best position
$gbest$	Global best position
$V_T\ (s)$	Terminal voltage
$V_{ref}\ (s)$	Reference voltage
$V_e\ (s)$	Error voltage
K_P	Proportional gain
K_i	Integral gain
K_d	Derivative gain
K_u	Ultimate gain
P_u	Ultimate time period
T_p	Peak time
T_r	Rise time
T_s	Settling time
e_{ss}	Steady state error
OS%	Overshoot
RH	Routh Hurwitz
T-E	Trial and Error
v	Velocity
x	Current position of particle
n	Number of particles
m	Number of members
i	Current iteration number
w	Inertia weight factor
c_1	Cognition learning
c_2	Social learning
C_{max}	Peak amplitude
C_{final}	Final response of steady output

CHAPTER 1

INTRODUCTION

1.1 Background

In electrical power distribution systems, it is essential in protecting sensitive equipment, enhancing performance, and maintaining reliability. Despite changes in the power distribution system’s load, the electrical supply that flows from the power generation system to the loads needs to be controlled. In order to maintain the power system bus voltage at nominal operating conditions despite large load variations ,the synchronous generator terminal voltage has to be regulated precisely. One of the primary control concerns of the electric power system is the stability of the nominal voltage level [1]. The durability of electrical equipment is extremely sensitive to variation in the rated supplied voltage, making voltage regulation a crucial aspect of system control. Raising the voltage or implementing series capacitors in power generation system is one of the methods for increasing stability and controls voltage level in an electric power grid. Operational frequency and operating voltage levels are two crucial characteristics that should always be controlled [2]. Ensuring stability in the power grid is essential for optimal performance, productivity and reliability of electrical equipment, while simultaneously preventing blackouts and safeguarding a consistent electricity supply.

Precise regulation and maintenance of voltage in electrical systems is effectively achieved through an Automatic Voltage Regulator (AVR) system. AVRs are often used in generators to control and regulate terminal voltage throughout an electrical system. The AVR regulates the terminal voltage by controlling the generator's exciter voltage [3]. The generator's output is regulated by monitoring the voltage at its terminals and comparing it to a predetermined reference value. Any deviation from the reference voltage triggers an adjustment in the field current, either increasing or decreasing it. This change in field current, in turn, changes the voltage produced by the main stator.

The high inductance and load fluctuations inherent in generator field windings in power systems pose challenges in achieving optimal regulator stability and transient response. Enhancing the AVR's functionality, stability and effective reaction to brief variations in terminal voltage is important. Numerous control structures have seen major growth during the past decades. However, due to its ease of design and implementation, the Proportional Integral Derivative (PID) controller is the most commonly utilized controller in industries [4].

Therefore, proportional, integral, and derivative control are the three PID controller control terms. The PID controller ensures that the generator's terminal voltage remains within specified limits even if it is under variable load conditions and disturbances by effectively integrating these three control terms together. Elmer Sperry created the first PID controller in 1911 for the US Navy [5]. However, accurately adjusting the gains of PID controllers can be challenging. In recent years, many intelligent optimization algorithms based evolutionary computation techniques have been proposed to tune the parameters of the PID controller in the AVR system. Such algorithms include Fuzzy-Logic controller (FLC), Taguchi Combined Genetic Algorithm (TCGA), Particle Swarm Optimization (PSO), Many Optimizing Liaisons (MOL), Artificial Bee Colony (ABC), Chaotic Ant Swarm (CAS), Genetic Algorithm and Bacterial Foraging (GA-BF), Local Unimodal Sampling Optimization (LUSO), and Teaching Learning Based Optimization (TLBO) [6, 7].

AVRs are vital components of contemporary electrical systems and play a crucial role in maintaining stable and consistent voltage output in various applications. Their ability to protect sensitive equipment, enhance power quality, and optimize generator performance makes them indispensable components in modern electrical systems.

1.2 Motivation

As the demand for reliable and efficient power sources grows, AVRs play a larger role in protecting sensitive equipment and ensuring the stable operation of electrical systems. AVR system can be useful for maintaining and stability of the terminal voltage of a system. However, to improve and enhance the system performance AVR needs a PID controller and optimization algorithms for optimal transient responses.

1.3 Problem Statement

The AVR system without a controller might cause equipment damage due to uncontrolled voltage fluctuations. Inserting PID controller to the system will overcome this problem by tuning the gains of PID controller. However, the three proportional, integral, and derivative interact with one another. Adjusting one parameter might alter the behavior of signal and causes a change in transient response. The AVR system would be unable to actively regulate output voltage, causing difficulties in maintaining the stable voltage level and AVR system performance. Optimal transient response needs to be obtained in AVR system to improve the system performance and reduce the risk of equipment malfunction [8].

1.4 Objectives

1. To study the dynamic behaviour of Automatic Voltage Regulator (AVR) transient responses in terms of peak time, rise time, settling time, overshoot, and steady-state error without controller implementation.

2. To implement the conventional methods (T-E), ZN and Particle Swarm Optimization (PSO) as tuning methods of PID controller for optimal transient responses.

3. To verify the effectiveness and robustness of optimal PID controller under various tracking of input voltage as a real industrial practice.

.

1.5 Scope of study

1. Apply the established modelling of AVR referring to the previous research [8].
2. Simulate the model of AVR without and with PID controller via MATLAB/Simulink environment.
3. MATLAB 2023b is used for simulation executions.
4. PID controller is chosen as a closed-loop system.
5. PSO algorithm is selected for tuning the PID controller.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter emphasis on plenty of research that has been conducted on AVR system, PID controller, Ziegler-Nichols tuning method and Particle Swarm Optimization (PSO) method that has been used to enhance performance of AVR system.

2.2 Automatic Voltage Regulator (AVR) System

The AVR continuously monitors the output voltage and automatically alters it to maintain a consistent level. It will also increase the system's performance and efficiency. Consistent voltage ensures that electrical gadgets and equipment operate at peak performance. The four primary parts of an AVR system are the generator, sensor, amplifier, and exciter. Each component is represented as a first order system with a gain and a time constant [9]. The four AVR major components transfer functions are shown in Figure 2.1.

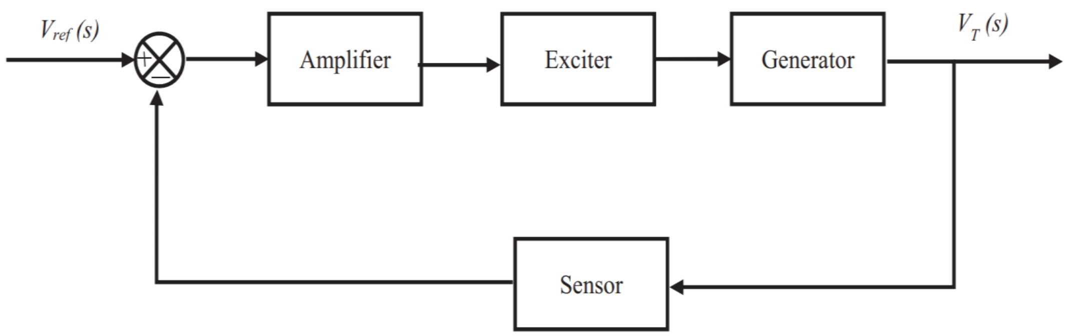

Figure 2.1 : Block diagram of AVR system without controller [10]

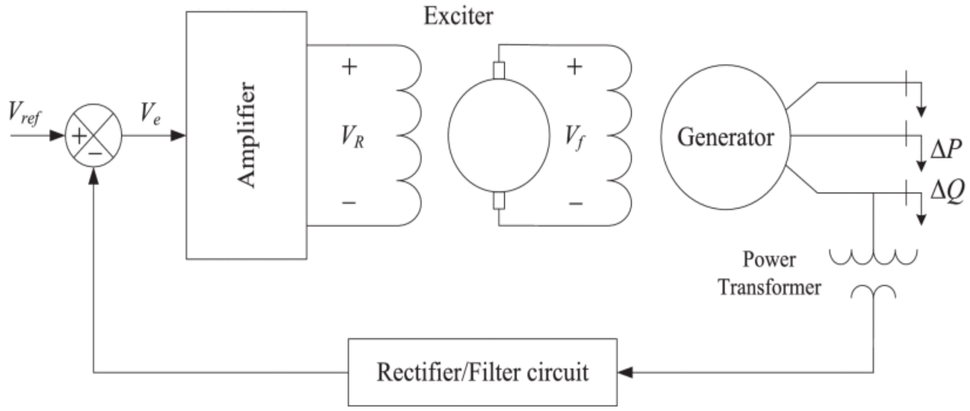

Figure 2.2 : Circuit diagram of AVR system [11]

Each of the components have their transfer function and range of parameters as shown in Table 2.1. The sensor continuously senses the voltage $V_T\ (s)$ at the generator's terminal and compares it to the required reference voltage $V_{ref}\ (s)$. The difference in voltages between the reference and sensing terminals error voltage $V_e\ (s)$ is amplified by the amplifier and utilized to activate the generator via the exciter [9, 10]. The transfer function of AVR system without a controller is expressed as below:

$$\frac{\Delta V_T\,(s)}{\Delta V_{ref}(s)} = \frac{K_A K_E K_G (1 + sT_S)}{(1 + sT_A)(1 + sT_E)(1 + sT_G)(1 + sT_S) + K_A K_E K_G K_S} \quad (2.1)$$

$$\mathrm{G_s} = \frac{0.1\mathrm{s} + 10}{0.004\mathrm{s}^4 + 0.0454s^3 + 0.555s^2 + 1.51s + 11} \quad (2.2)$$

Table 2.1 : Transfer function and range of parameters of the AVR system [10]

	Transfer Function	Range of parameters	
		Gain constant	Time constant
Amplifier	$T(s) = \frac{K_a}{1 + \tau_a s}$	$10 \le K_a \le 40$	$0.02 \le \tau_a \le 0.1$
Exciter	$T(s) = \frac{K_e}{1 + \tau_e s}$	$1 \le K_e \le 10$	$0.4 \le \tau_e \le 1.0$
Generator	$T(s) = \frac{K_g}{1 + \tau_g s}$	$0.7 \le K_g \le 1.0$	$1.0 \le \tau_s \le 2.0$
Sensor	$T(s) = \frac{K_s}{1 + \tau_s s}$	$0.9 \le K_s \le 1.1$	$0.001 \le \tau_s \le 0.06$

2.3 Proportional Integral Derivate (PID) Controller

In industrial applications, PID controllers are the most often utilized controllers. Approximately 90% of controllers in industries are PID controllers [12]. These are the most widely used controllers. This can be attributed to its high-performance behavior as well as simple structure. Proportional, integral, and derivative modes are the three primary modes of the PID controller. A proportional controller shortens the rise time but does not completely eliminate steady-state error. The integral controller can eliminate the steady-state error, but it could worsen the transient response. A derivative controller increases system stability, reduces $OS\%$, and improves transient response. If the derivative gain is high, the process may become unstable [13]. This controller has three parameters: proportional gain K_p integral gain K_i , and derivative gain K_d. The controller's gains are tuned using a trial-and-error process based. The block diagram of PID controller and PID controller with an AVR system is shown in Figure 2.3.

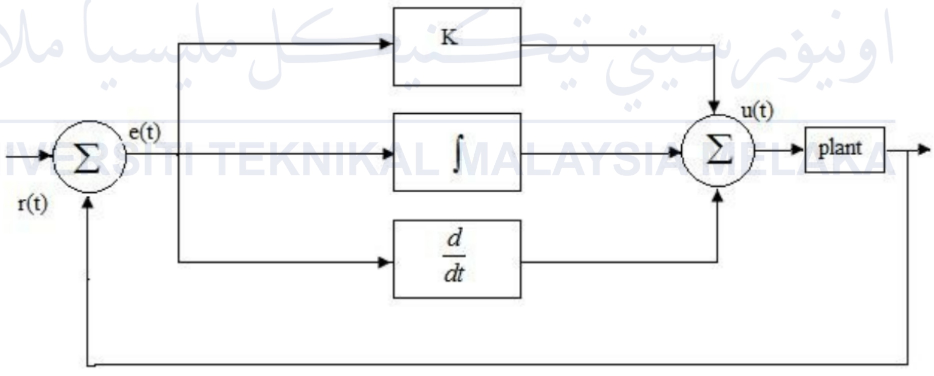

Figure 2.3 : PID controller closed loop block diagram [13]

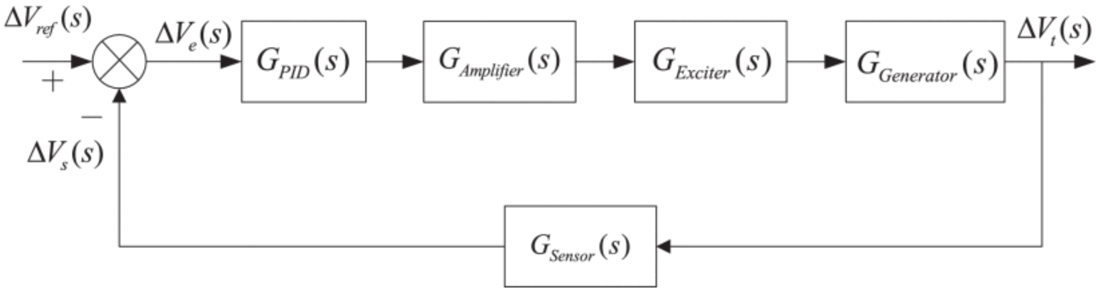

Figure 2.4 : AVR system with closed loop PID controller block diagram [11]

The transfer function of PID controller is :

$$G_{PID}(s) = \left(K_p + \frac{K_i}{s} + \frac{K_d}{s} \right) \tag{2.3}$$

where K_p, K_i and K_d are the proportion coefficient, differential coefficient, and integral coefficient, respectively.

2.4 Ziegler-Nichols Method

The Ziegler-Nichols method was introduced in the 1940s. ZN has two method step response and frequency response method which are also known as open loop and closed loop system. The method that has been used in this project is closed loop method. The closed loop ZN method is a comprehensive tuning technique that is frequently used in industry to optimize the parameters of PID controllers. Besides, the ZN method has been mostly used as a benchmark to modify PID parameters in several research. It is because the closed loop ZN rule does not require any model knowledge and is likely to achieve sufficient results just by using the given formula [14]. The ultimate gain K_u is the gain which triggers the system to oscillate undamped, and the corresponding time period is the ultimate time period T_u. The value of K_u and T_u can be determined by using the Routh Hurwitz (RH) table and the characteristic equation [15]. The obtained K_u and T_u value must be substituted into the given formula in Table 2.2 .

Table 2.2: Parameters of Ziegler Nichols using Closed-loop method [16]

Controller	K_p	T_I	T_D
P	$0.5K_u$	-	-
PI	$0.45K_u$	$0.83T_u$	-
PID	$0.6K_u$	$0.5T_u$	$0.125T_u$

$$K_i = \frac{K_p}{T_i} \tag{2.4}$$

$$K_d = K_p T_d \tag{2.5}$$

2.5 Particle Swarm Optimization (PSO)

Particle Swarm Optimization technique is an evolutionary optimization algorithm which is based on the principles of biological evolution inspired by observing and analysing the swarm patterns that occur naturally such as bird flocks and fish schools. PSO was first introduced by Kennedy and Eberhart in 1995 [16]. PSO is a metaheuristic because it provides a flexible and adaptive method of optimization. It researches a wide range of solutions without making any strong assumptions about the problem, making it useful in solving a wide range of real-world optimization issues. The parameters of PSO are the number of particles, weight of inertia, number of iterations, cognitive and social learning. The inertia weight begins with a value of 0.9 and decreases linearly to 0.4 over the duration of the simulation, it has the potential to significantly enhance PSO performance. Most author used 20 particles when simulating PSO. The authors states that PSO algorithms performs best with the population size set withing 20 to 50 [17, 18]. In PSO, particles are "flown" around a hyperdimensional search space, exploring, and navigating for identifying the optimal point to the optimization problem. Particles are classified into two categories which are known as personal best position $(pbest)$ and global best position $(gbest)$. The $(pbest)$ particle represents the position that the particle has discovered to have the optimal fitness function value of all the positions it has explored so far. The global best position $(gbest)$ represents the optimal position determined by the entire group or swarm of particles [19]. Then, each particle analyses its performance and interacts with others, enabling them to make better decisions.

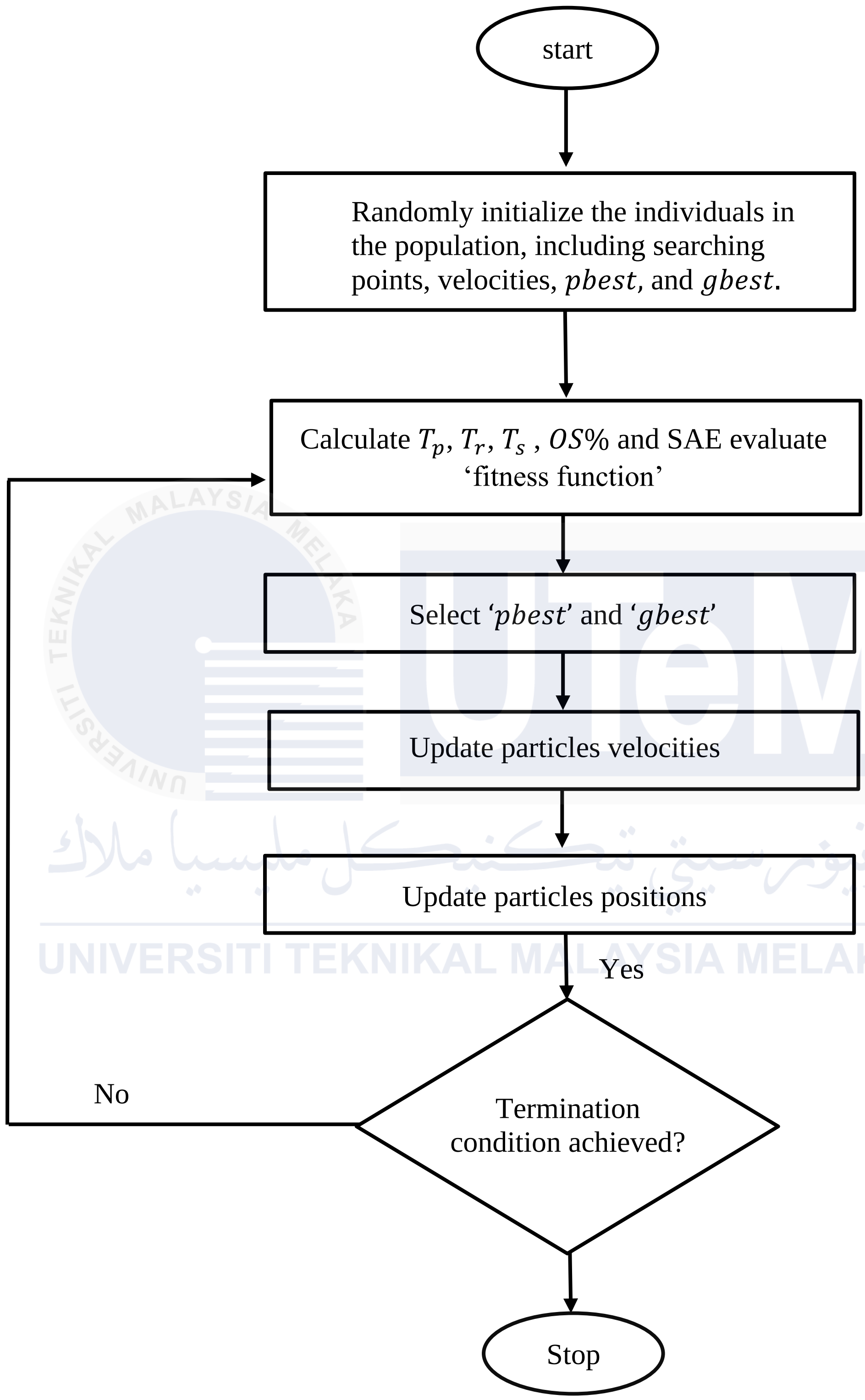

Figure 2.5 : PSO-PID flowchart [20]

2.6 K-Chart

Figure 2.5 shows the completed K-Chart of AVR system. This is useful for identifying the process flow and summarizes for the literature reviews.

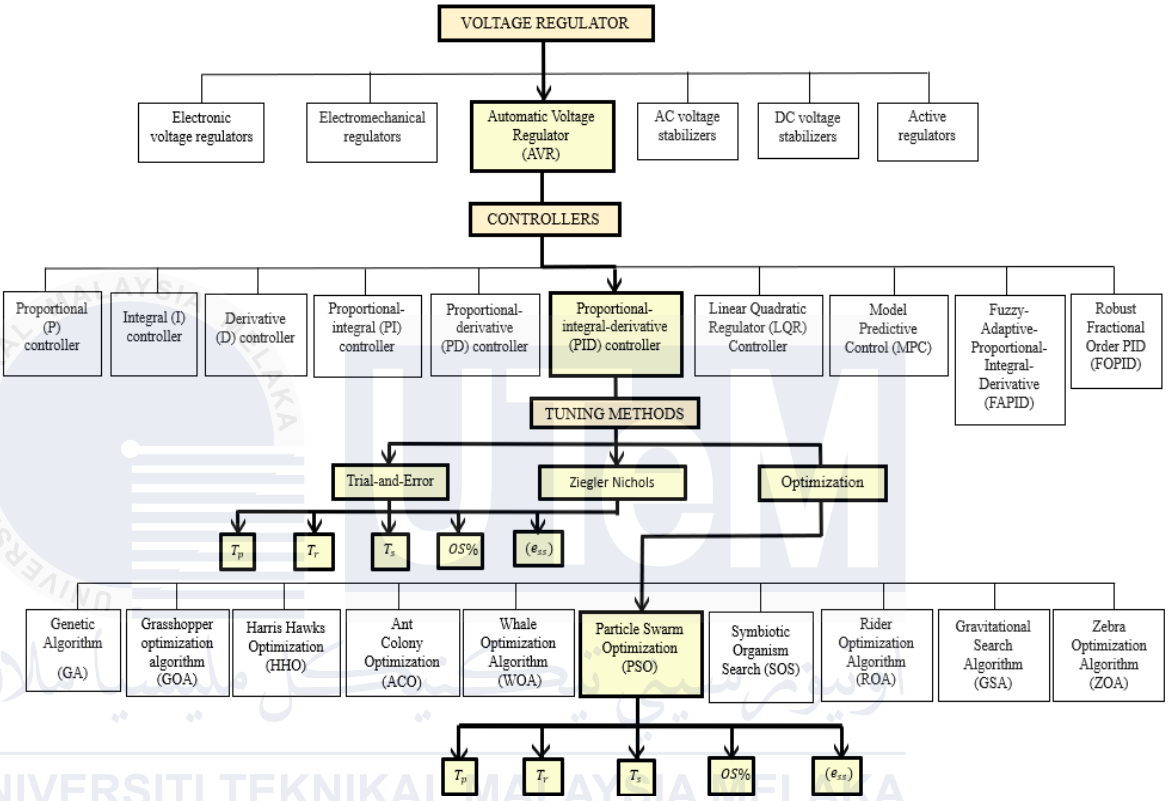

Figure 2.6 : K-Chart of AVR System

CHAPTER 3

METHODLOGY

3.1 Introduction

This chapter emphasis the methodology and flow of the project. This process is done by referring to various journals as stated in Chapter 2. This chapter involves the flowchart, procedure, parameters, formula, calculation and software implemented in this project. This project is based on simulation which has been fully conducted on MATLAB Simulink software.

3.2 Project Flowchart

The project flowchart has been divided into three parts, each of it represents all three objectives separately. The first objective is presented on how the AVR system with and without controller are be designed. The AVR system without controller is stimulated with a step input of 1. Besides for AVR system with PID (T-E) is stimulated with three step input which are 0.5, 1.0, 1.5. Then, the results are compared and analyzed to know the difference between the transient response when using difference value of step inputs. The obtained results are taken forward to Objective 2. Objective 2 is focused on the ZN tuning methods of PID controllers. Before stimulating PID controller using ZN method, the parameters of PID controller are calculated using the provided formula. In Objective 2 all the stimulation is carried out using step input 1. These results are compared and analyzed to identify the optimal method so it can be compared with PSO algorithm which is Objective 3. In Objective 3 research on PSO is conducted and AVR system stimulated using PSO algorithm. Then the obtained transient response is compared and analyzed with AVR system without controller, with PID (T-E) method and ZN method to find out the best optimization for AVR system.

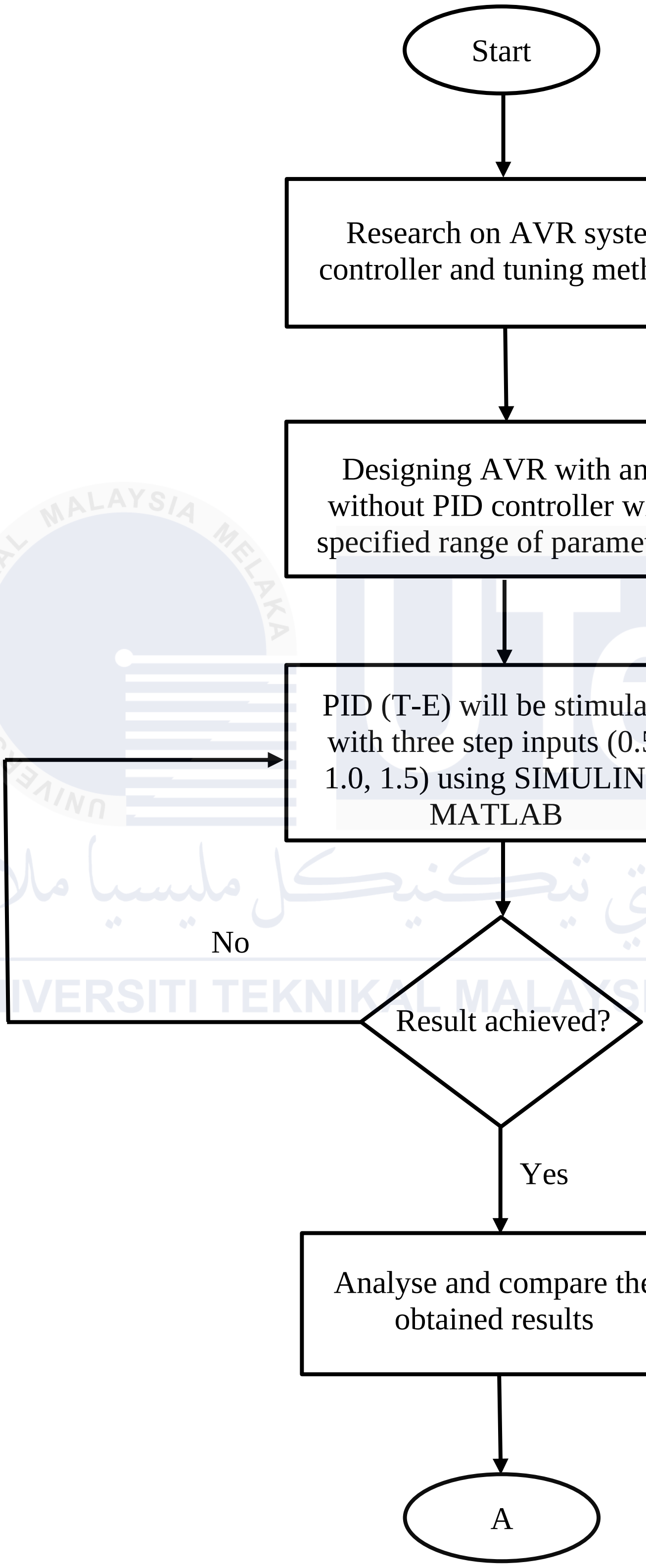

Figure 3.1 : Flowchart of Objective 1

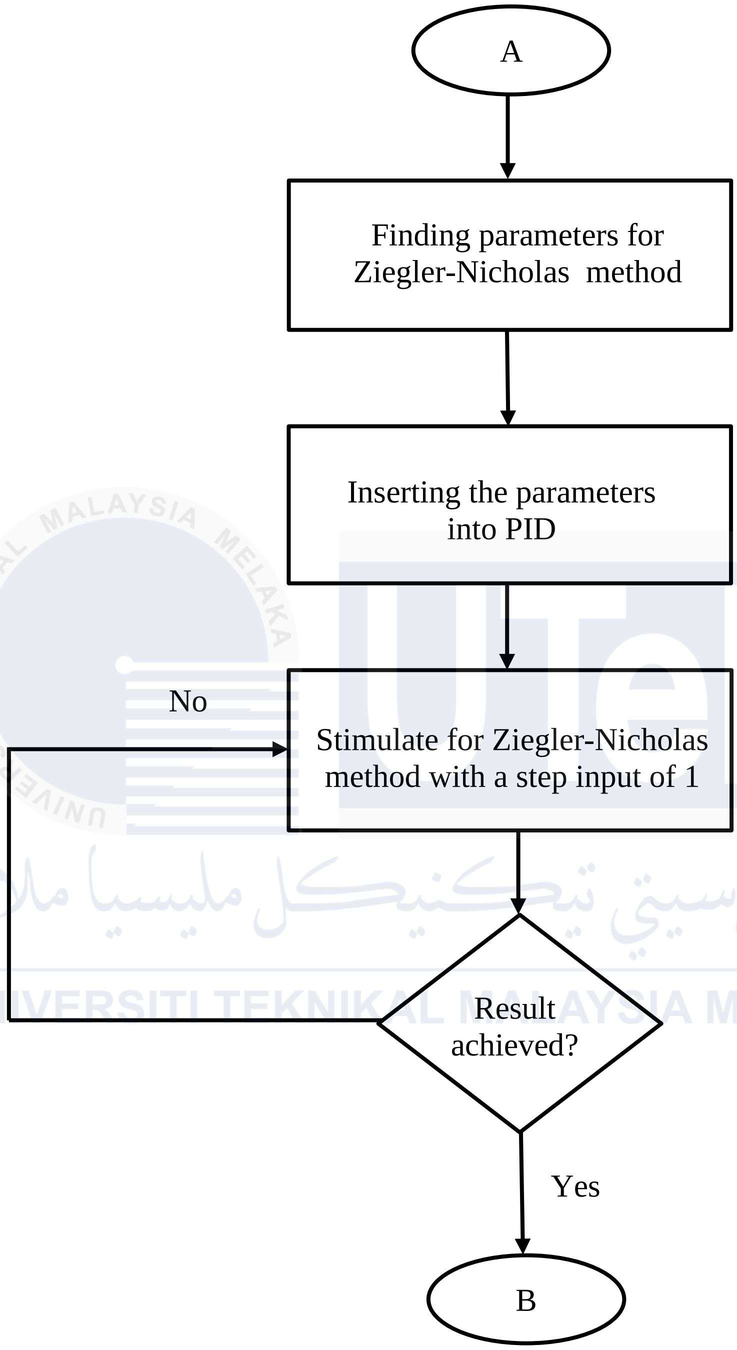

Figure 3.2 : Flowchart of Objective 2

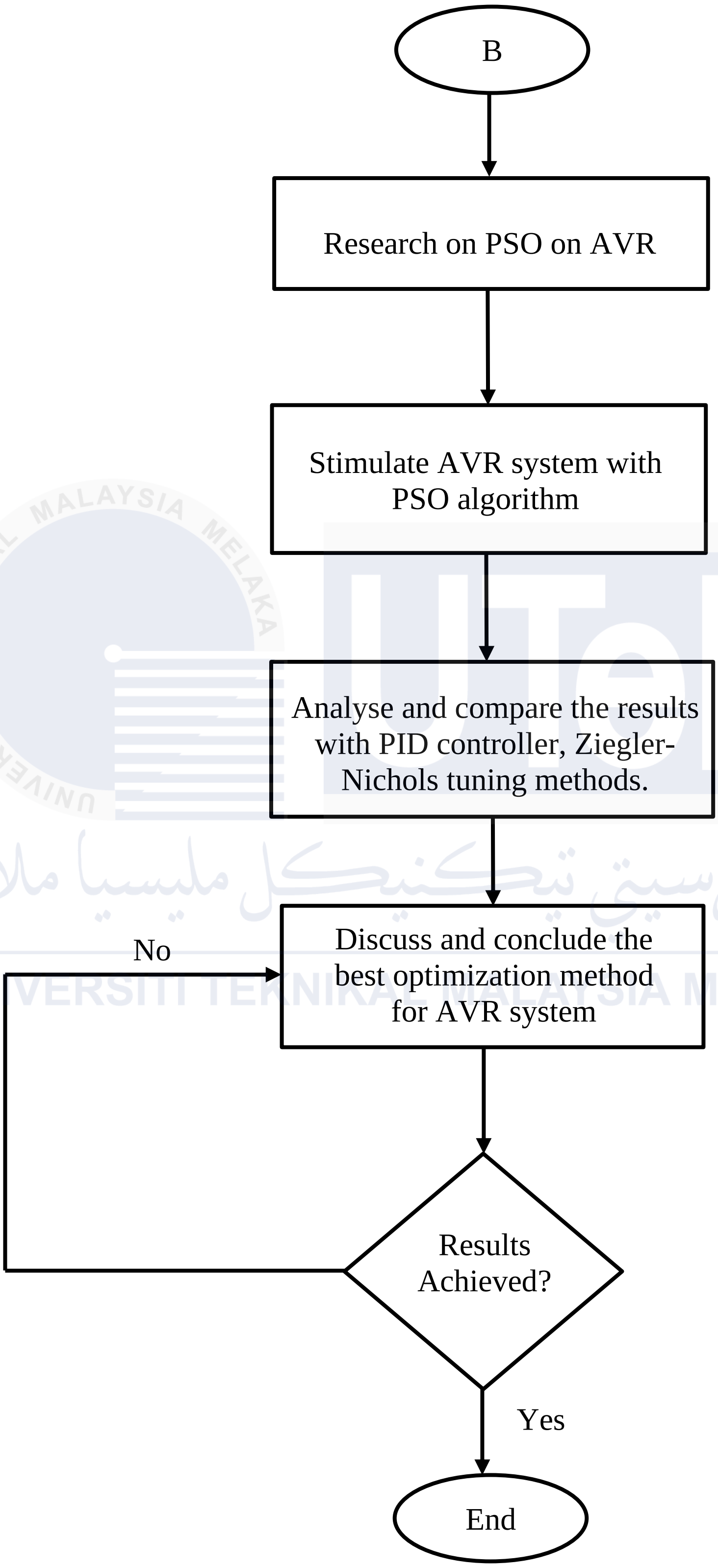

Figure 3.3 : Flowchart of Objective 3

3.3 Automatic Voltage Regulator (AVR) system without controller

The AVR system without controller is designed in Simulink model and the parameters were selected within the given range as shown in Table 3.1. The first model is step input resembles as V_f. The step time, initial value, final value and sample time in step input were set to 0, 0, 1 and 0.1, respectively. Next followed by the amplifier model, exciter, generator, sensor and scope (output) of the system are as shown in Figure 3.4.

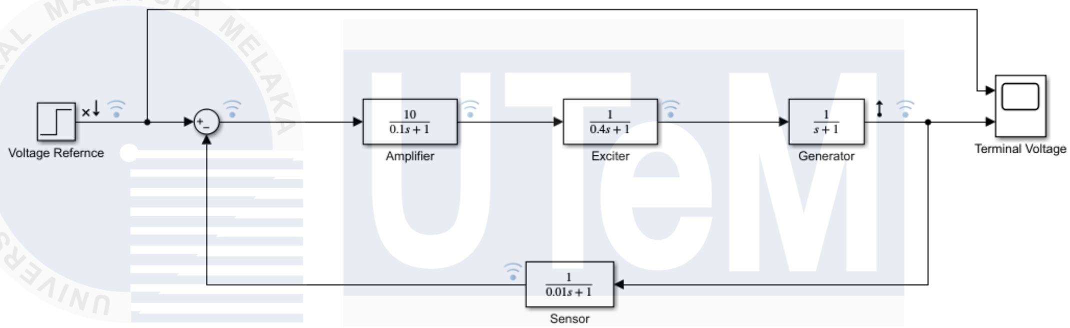

Figure 3.4 : Block Diagram of AVR without controller in Simulink

Table 3.1 : The parameters for AVR system

	Chosen range	Range of parameters	
		Gain constant	Time constant
Amplifier	$K_a = 10$ $\tau_a = 0.1$	$10 \leq K_a \leq 40$	$0.02 \leq \tau_a \leq 0.1$
Exciter	$K_e = 1$ $\tau_e = 0.4$	$1 \leq K_e \leq 10$	$0.4 \leq \tau_e \leq 1.0$
Generator	$K_g = 1$ $\tau_s = 1$	$0.7 \leq K_g \leq 1.0$	$1.0 \leq \tau_s \leq 2.0$
Sensor	$K_s = 1$ $\tau_s = 0.01$	$0.9 \leq K_s \leq 1.1$	$0.001 \leq \tau_s \leq 0.06$

3.4 Automatic Voltage Regulator (AVR) system with Manual (Trial and Error) PID Controller

This system has a similar block diagram as the AVR system, but with an additional model of PID controller. As shown in Table 3.2, the values of proportional gain K_p, integral gain K_i, and derivative gain K_d are manually set using a trial-and-error method. A clock and workspace are added to the diagram to obtain the waveform of the tuned PID controller. In this part, the difference between the peak time (T_p), rise time (T_r), overshoot, settling time (T_s) and steady state error (e_{ss}) be compared using three step inputs (0.5, 1.0, 1.5).

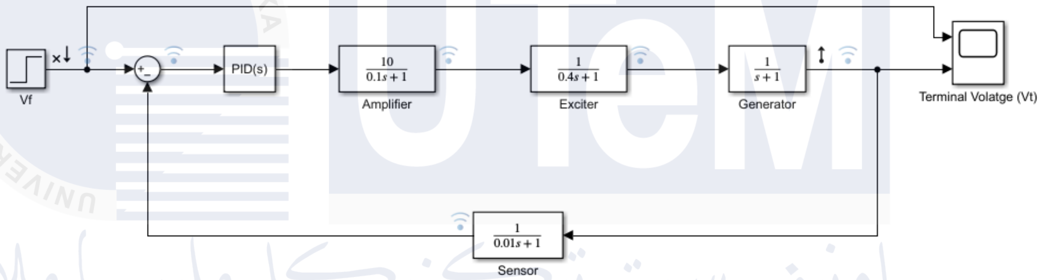

Figure 3.5 : Block diagram of AVR with PID controller in Simulink

Table 3.2 : Parameters of PID controller of AVR system

<table>
<tr><th></th><th>PID range</th><th>Manual tuning range</th></tr>
<tr><td>P</td><td rowspan="3">$0.2 \leq K_p, K_i, K_d \leq 2.0$</td><td>$K_p = 0.8$</td></tr>
<tr><td>I</td><td>$K_i = 1$</td></tr>
<tr><td>D</td><td>$K_d = 1.2$</td></tr>
</table>

3.5 Ziegler-Nicholas Method

After obtaining the value Pu and Ku using Routh Hurwitz (RH) table and characteristic equation , the values K_p, K_i and K_d were computed using the formula in Table 2.2 and Table 3.3 displays the estimated values of, K_p, K_i and K_d

$$G_s = \frac{0.1s + 10}{0.004s^4 + 0.0454s^3 + 0.555s^2 + 1.51s + 11} \quad (3.1)$$

$$G_s = \frac{K\,(0.1s + 10)}{0.004s^4 + 0.0454s^3 + 0.555s^2 + 1.51s + 11} \quad (3.2)$$

$$G_s = 0.004s^4 + 0.0454s^3 + 0.555s^2 + 1.51s + 0.1Ks + 10K + 11 \quad (3.3)$$

$$G_s = 0.004s^4 + 0.0454s^3 + 0.555s^2 + (1.51 + 0.1K)s + 10K + 11 \quad (3.4)$$

s^4	0.0004	0.555	10K +11
s^3	0.0454	2.52 + 0.1K	0
s^2	$\frac{0.024593 - 0.00004K}{0.0454}$	10K + 11	0
s^1	2.0094 – 0.345K	-	-
s^0	10K + 11	-	-

$$10K + 1 > 0$$

$$K > \frac{-11}{10}$$

$$K > -1.1$$

(3.5)

$$2.00949 - 0.354K > 0$$

$$K < \frac{2.0094}{0.354}$$

$$K < 5.68 \quad (3.6)$$

$$-1.1 < K_u < 5.68 \quad (3.7)$$

$$\frac{\omega n^2}{s^2 + 2\zeta\omega n + \omega n^2} = \frac{0.1s + 10}{0.004s^4 + 0.0454s^3 + 0.555s^2 + 1.51s + 11} \quad (3.8)$$

$$\omega n^2 = 11$$

$$\omega n = \sqrt{11}$$

$$\omega n = 3.3166\, rad/sec \quad (3.9)$$

$$\omega n = 2\pi f$$

$$f = \frac{3.3166}{2\pi}$$

$$= 0.5279 \tag{3.10}$$

$$T = \frac{1}{f}$$

$$= \frac{1}{0.5279} \tag{3.11}$$

$$= 1.8943s$$

Table 3.3 : Parameters of PID controller for ZN

	$\mathbf{K_P}$	$\mathbf{T_I}$	$\mathbf{T_D}$
$\mathbf{K_u} = \mathbf{1}$	$0.6K_u = 0.6(1) = 0.6$	$\mathbf{K_i} = \frac{\mathbf{K_p}}{\mathbf{T_I}} = \frac{0.6}{0.94715} = 0.6335$	$\mathbf{K_d} = \mathbf{K_p T_d} = 0.6x0.23678 = 0.1421$

3.6 Particle Swarm Optimization (PSO)

AVR system with PSO has similar block diagram as in Figure 3.5 but additionally added with to file. In this part, the values of K_p, K_i and K_d are stimulated using three iterations 30, 50 and 100. Each set of iterations is executed ten times in order to get the optimal result for all of the simulations performed.

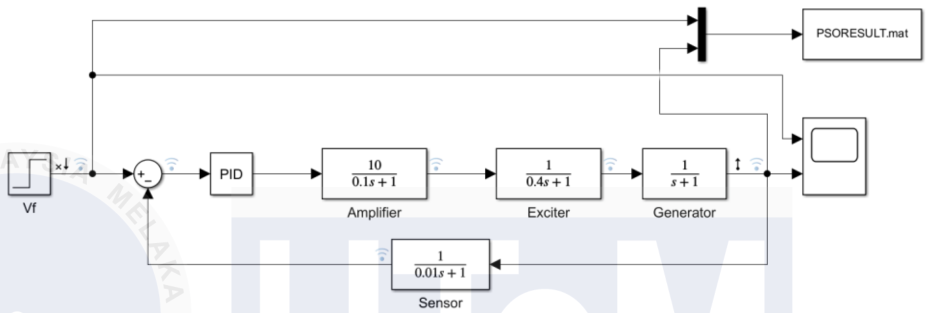

Figure 3.6 : Block Diagram of AVR with PSO-PID controller in Simulink

The following equations are used to update each particle's position and velocity in a multidimensional [18] :

$$v_{j,g}(t+1) = w.v_{i,g}(t) + c_1 r_1 [pbest_{j,g}(t) - x_{j,g}(t)] + c_2 r_2 [gbest_g(t) - x_{j,g}(t)] \quad (3.12)$$

$$x_{j,g}(t+1) = x_{j,g}(t) + v_{j,g}(t+1) \quad (3.13)$$

where :

v velocity of particle

x current position of particle j at iteration t

w inertia weight factor

c_1 cognition learning

c_2 social learning

$pbest$ the individual best position of particle j until iteration t

$gbest$ the best particle in the swarm at iteration t

The values of r_1 and r_2 are random number within a range between 0 and 1.

Table 3.4 Parameters of PSO

Parameters	Values
c_1	2
c_2	2
n	20
Wmax	0.9
Wmin	0.4
i,maxiter	30, 50 and 100
Fitness function	SAE

Table 3.5: Parameters of PID controller for PSO (i =30)

	STEP INPUT = 1		
Test	K_p	K_i	K_d
1	1.8121	0.3896	0.3245
2	1.1461	0.7465	0.8389
3	1.0506	0.7479	0.2900
4	1.4750	1.0899	0.2651
5	1.1028	0.6935	0.5681
6	1.2851	1.1244	0.7549
7	1.3550	0.9862	0.2562
8	1.3847	0.8612	0.8603
9	1.6949	1.0189	0.2677
10	1.9458	0.4741	0.3564

Table 3.6 : Parameters of PID controller for PSO (i =50)

	STEP INPUT = 1		
Test	K_p	K_i	K_d
1	1.5242	1.1992	0.9171
2	1.9409	0.9535	1.0360
3	1.6967	0.2430	0.2386
4	1.9994	0.9731	0.2714
5	1.8179	0.3704	0.3355
6	1.9880	0.3150	0.2714
7	1.8287	1.0951	0.3816
8	1.8772	0.9666	0.2480
9	1.2523	0.9325	0.2294
10	1.8864	0.3791	0.3426

Table 3.7 : Parameters of PID controller for PSO (i =100)

	STEP INPUT = 1		
Test	K_p	K_i	K_d
1	1.3646	0.8242	0.3704
2	1.2589	0.9537	0.7551
3	1.9753	0.3471	0.3789
4	1.9547	0.3785	0.3849
5	1.9546	1.0161	0.3956
6	1.1768	1.0453	0.8424
7	1.9534	0.6841	0.3814
8	1.8716	1.0176	1.0596
9	1.8968	0.3492	0.3249
10	1.8793	0.8404	0.3327

3.7 Robustness Analysis of the PID Tuning Techniques

This system will determine the robustness of the conventional and optimization tuning techniques. The block diagram of AVR system without controller, PID (T-E), ZN and PSO is modified by replacing the step input to the signal builder block as shown in Figure 3.8 shows four signals 1, 2, 3 and 4 in the signal builder which is connected using the sum block to produce the various input. The role of the signal builder is to produce and construct groups of interchangeable signals with piecewise linear waveforms. Then the system runs to analyses the transient response based on the various input that has been implemented.

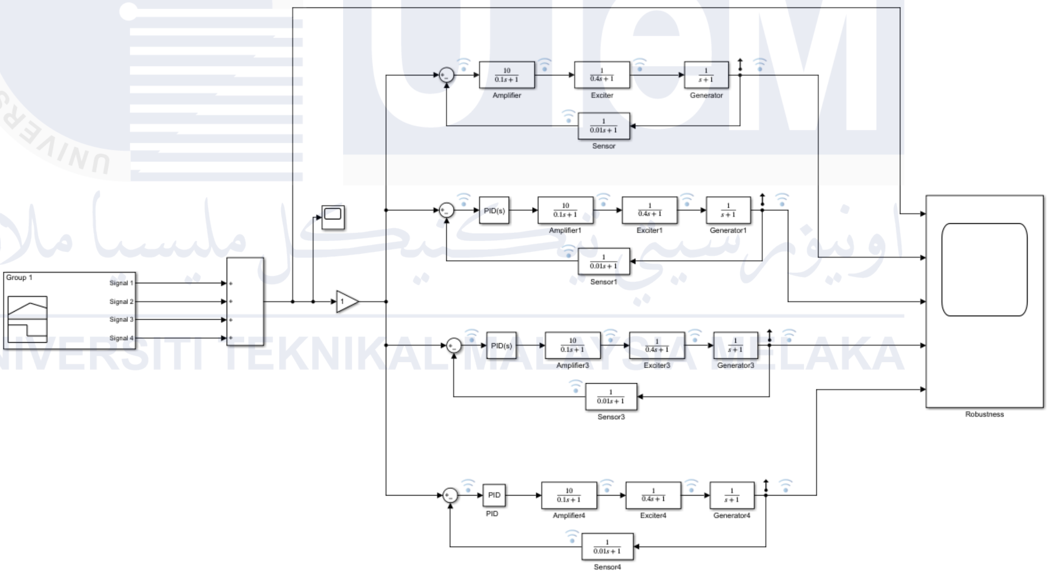

Figure 3.7 : Block diagram for determining the robustness

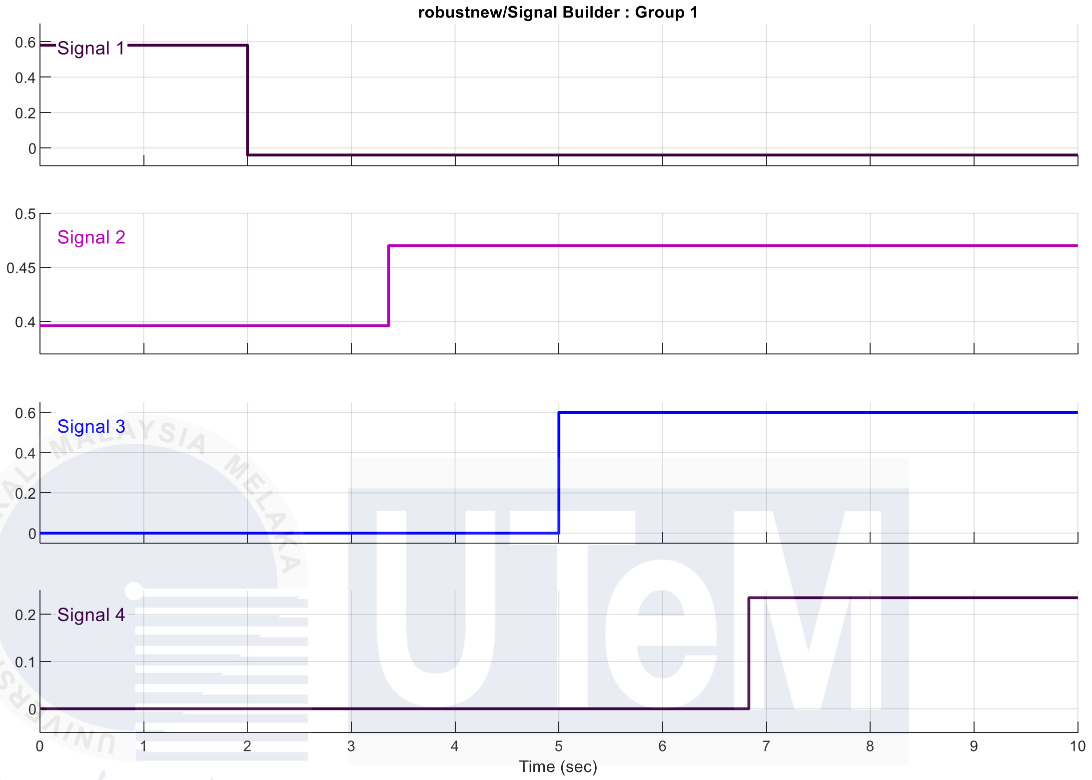

Figure 3.8 : Various input of signal using signal builder

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter outlines the stimulation of an AVR system both with and without a controller using the MATLAB/Simulink software. Furthermore, a comparison of ZN tuning method and PSO algorithm with a specific range of parameters has been performed, and the results are shown in this chapter.

4.2 Automatic Voltage Regulator (AVR) system without controller

The block diagram of this system is shown in the previous Figure 3.4. By using the step input and scope as input and output respectively, the system has been stimulated. Furthermore, the chosen parameters are within the given range as shown in the previous Table 3.1. The system is being tested to identify the peak time (T_p), rise time (T_r), settling time (T_s), overshoot ($OS\%$) and steady state error (e_{ss}) without the presence of controller in a AVR system. T_p is determined by observing the first peak (highest peak) that occurs in the system. Next the T_r can be observed by determining the difference between the early rising stage and before the step response reaches the peak value. Furthermore, T_s can be determined by forming two lines in between the desired voltage with a difference of +/- 2%. The point at which the step response stays within this range between the two lines is known as the system's settling time. The $OS\%$ is obtained by subtracting C_{max} with C_{final} , divide it with C_{final} and finally times by 100%. Steady state error can be obtained by calculating the difference between the $V_T\ (s)$ and $V_{ref}(s)$. As shown in Figure 4.1, the acquired value from the stimulation has been recorded in the table below alongside with the results of [8].

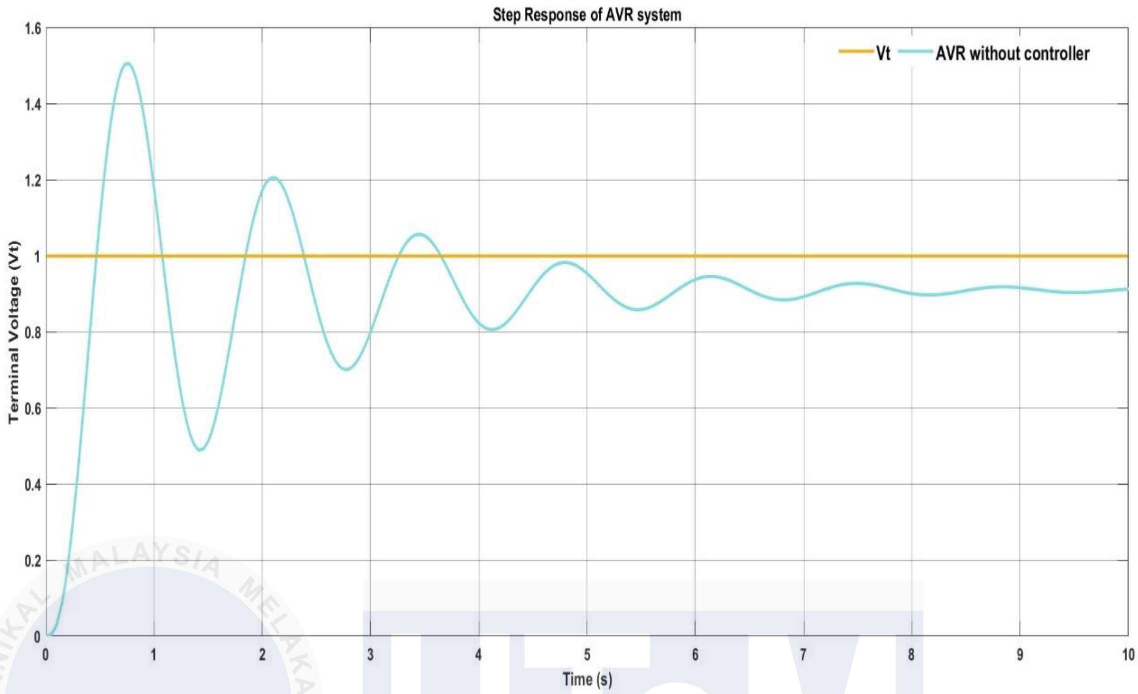

Figure 4.1 : Step Response of AVR without controller

Table 4.1 : Validation results of AVR system without controller

	STEP INPUT =1	
Transient response	**Simulation results**	**Journal result [8]**
T_p (s)	0.752	0.7547
T_r (s)	0.261	0.2607
T_s (s)	6.99	6.9711 (+- 2%)
OS (%)	65.7	65.4272
(e_{ss})	0.1	0.0907

4.3 Automatic Voltage Regulator (AVR) with PID (T-E) controller

The AVR system is stimulated using PID controller with three different step inputs as shown in Figure 4.2, 4.3 and 4.4. This experiment is conducted in order to investigate the significant differences between the T_p, T_r, T_s, $OS\%$ and (e_{ss}).

4.3.1 Step input 0.5

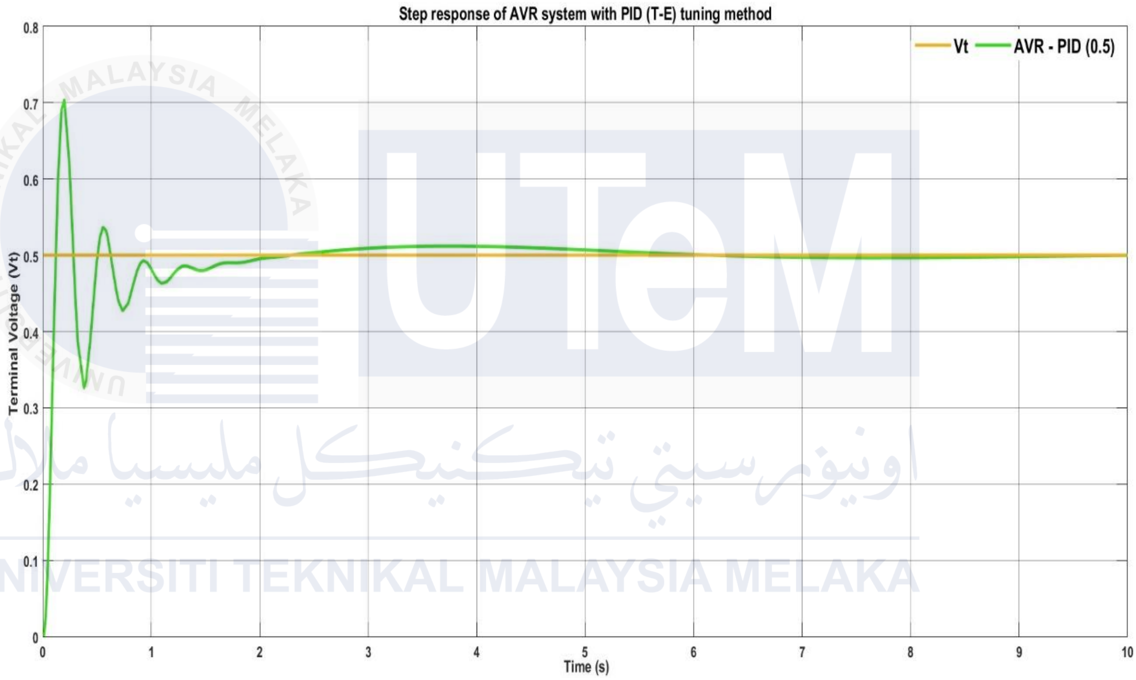

Figure 4.2 : Step response of AVR with PID (T-E) controller (step input 0.5)

4.3.2 Step input 1.0

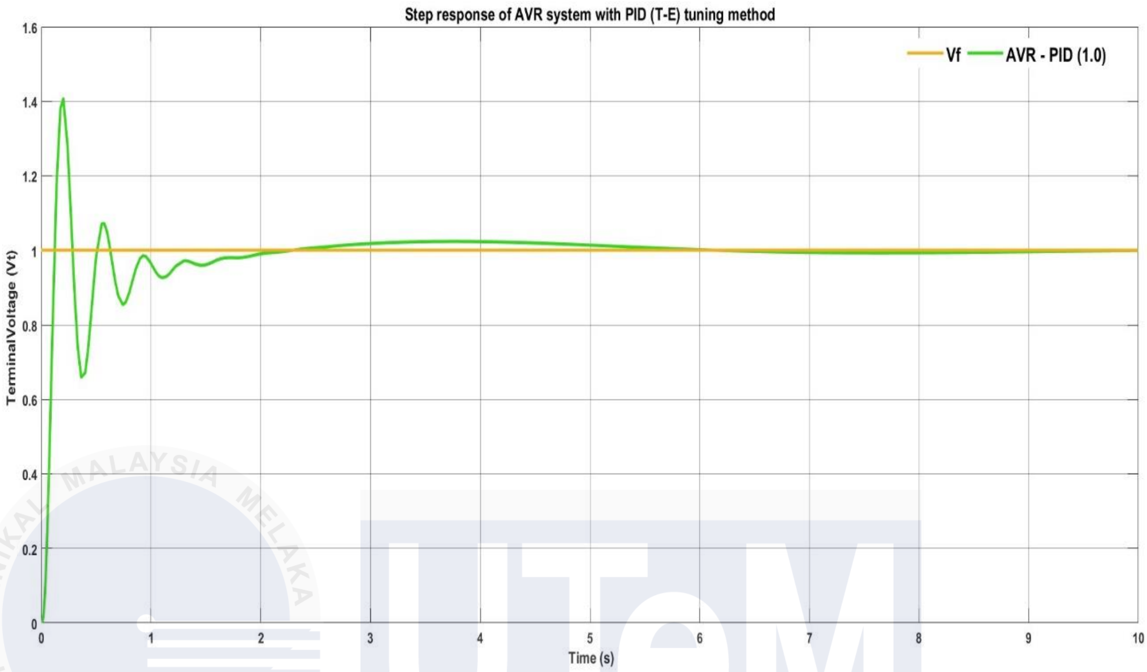

Figure 4.3 : Step response of AVR with PID (T-E) controller (step input 1.0)

4.3.3 Step Input 1.5

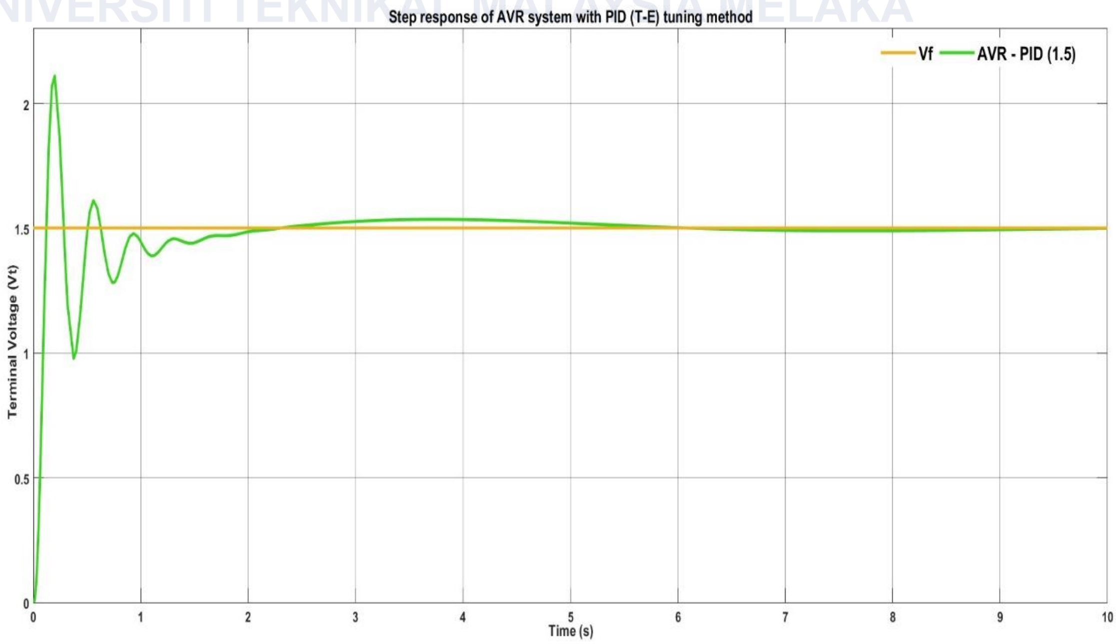

Figure 4.4 : Step response of AVR with PID (T-E) controller (step input 1.5)

Table 4.2 : Results of AVR system with PID controller

Transient response	**STEP INPUTS**		
	0.5	**1.0**	**1.5**
$T_p\ (s)$	0.200	0.200	0.200
$T_r\ (s)$	0.075	0.076	0.076
$T_s\ (s)$	4.41	4.48	4.74
OS (%)	42.143	42.143	42.143
(e_{ss})	0	0	0
K_p	0.8		
K_i	1		
K_d	1.2		

Table 4.2 shows the result of AVR system with PID (T-E) controller (three step input). The results show minimal differences in the value of the transient response when using different step input. The T_s and T_r changes as the step input changes. The T_p and $OS\%$ values remain constants for all three step inputs. As the step input value increases the settling time increases too. All step inputs have the same T_p and $OS\%$ value. The T_r increases by 0.001s when the input is changed from 0.5 to 1.0 and it remains 0.076 s when step input is 1.5.

4.4 Comparison between AVR System without controller and with PID (T-E) Tuning Methods

Figure 4.5 shows the step response between AVR system without controller and with PID (T-E) tuning method. From the figure we can obtain that AVR system PID (T-E) tuning method (light green) is more likely to achieve the desired $V_T\ (s)$ compared to AVR system without controller (purple).

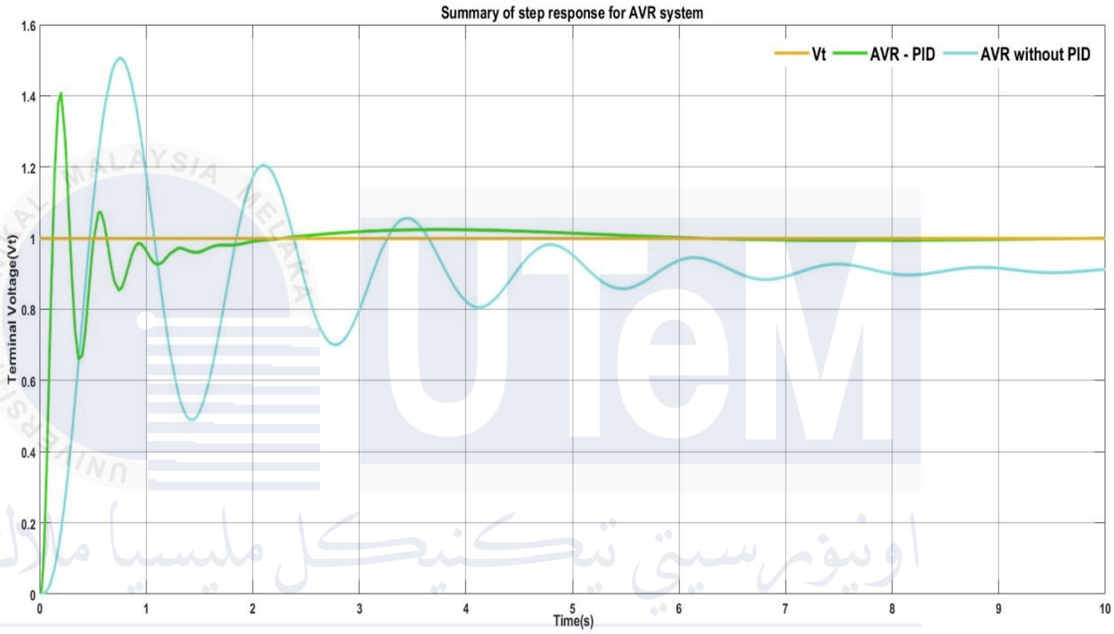

Figure 4.5 : Step response of AVR without Controller and PID (T-E) tuning

4.5 Ziegler-Nicholas Method

4.5.1 AVR-ZN Controller with K_U (1.0)

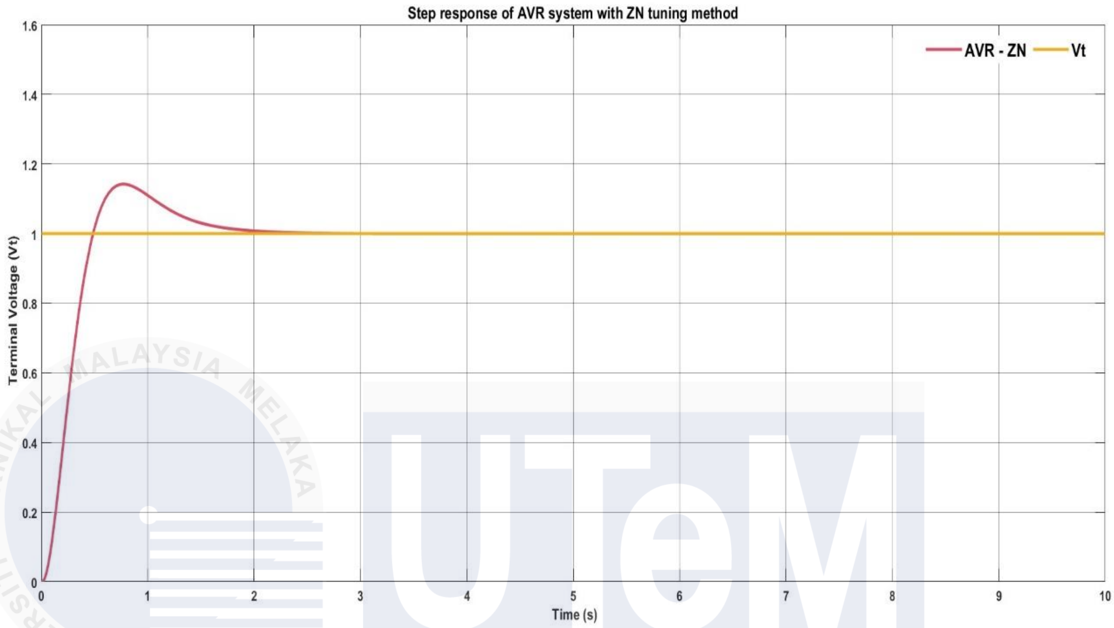

Figure 4.6 : Step response AVR-ZN with Ku (1.0)

Table 4.3 : Result of AVR system with ZN

Transient response	STEP INPUT = 1
T_p (s)	0.766
T_r (s)	0.326
T_s (s)	1.65
OS (%)	14.2
(e_{ss})	0
K_p	0.6
K_i	0.6335
K_d	0.1421

Table 4.3 shows the results for ZN method. The PID parameters determined using the equations in Table 3.3. After finding the values for K_p, K_i and K_d it was inserted into the AVR system's PID controller, then simulation was executed. The results showed a 27.94 % and 2.83 s reduction in $OS\%$ and T_s respectively compared to the manual tuning method. However, the peak time and rise time have increased by 0.566 s and 0.25 s respectively.

4.6 Comparison between AVR System without controller, with PID (T-E) Tuning Method and Ziegler-Nicholas method

Figure 4.5 shows the step response between AVR system without controller (purple), AVR system PID (T-E) (light green) and AVR system with ZN (maroon) tuning methods. Compared to AVR without controller and AVR PID (T-E) , the ZN is more likely to achieve the desired $V_T\ (s)$. Besides, the OS% and settling is lesser than AVR without controller and AVR PID (T-E) tuning method.

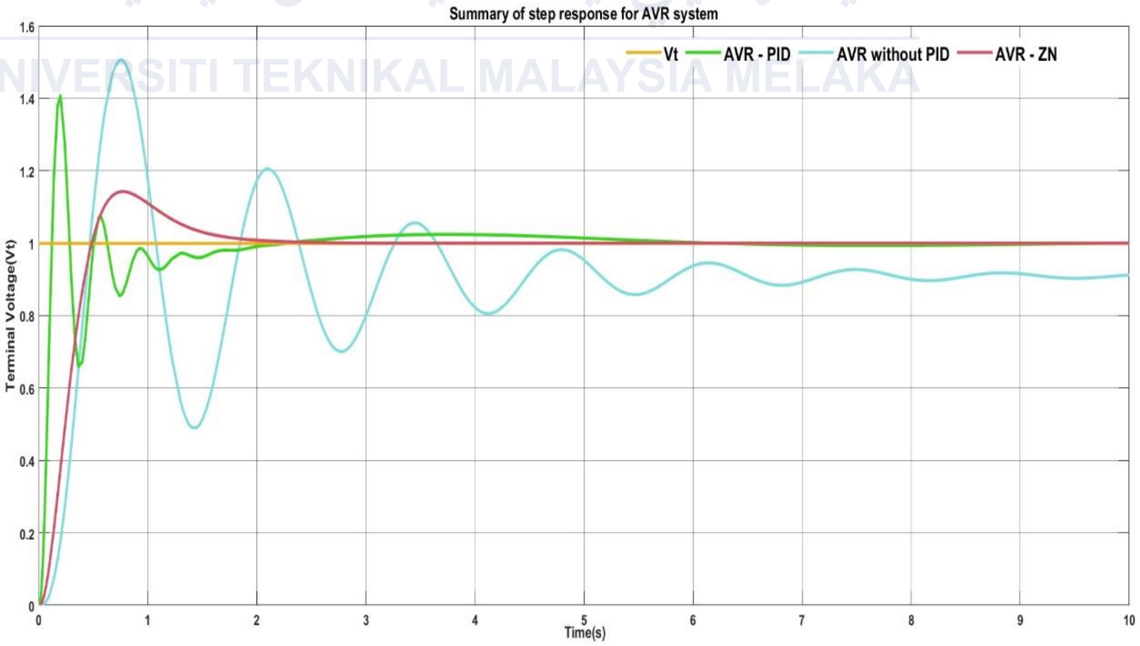

Figure 4.7 : Step response of AVR without Controller, PID (T-E) and ZN tuning methods

4.7 Particle Swarm Optimization (PSO)

PSO is stimulated using the model shown in Figure 3.6 in MATLAB. There were three number of iterations that are stimulated under PSO which are 30, 50 and 100. The number of iterations (N) is adjusted in the editor section after 10 tests for each iteration have been stimulated as shown in Figure 4.8, 4.9 and 4.10. For each iteration the test is conducted 10 times to find the optimal values for K_p, K_i and K_d. Then, the stimulated K_p, K_i and K_P value is be inserted into AVR with PID controller model as shown in Figure 3.5. Tables 4.4, 4.5, and 4.6 illustrate the transient response data obtained during AVR-PID model stimulation.

4.7.1 PSO (30 iterations)

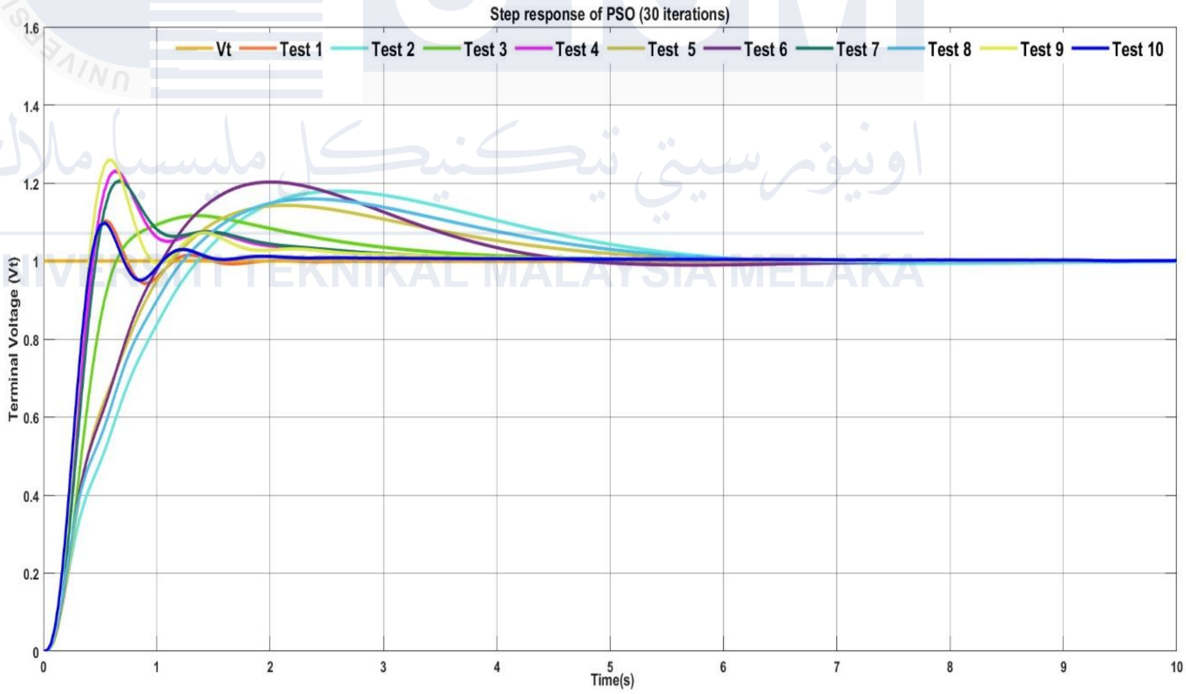

Figure 4.8 : Step response of AVR – PSO for 30 iterations

Table 4.4 : Result of AVR system with PSO (30 iteration)

	STEP INPUT = 1				
Test	$T_p\ (s)$	$T_r\ (s)$	$T_s\ (s)$	OS (%)	(e_{SS})
1	0.544	0.255	1.025	10.556	0
2	2.600	0.954	4.999	18.452	0
3	1.378	0.381	2.546	11.798	0
4	0.632	0.261	1.904	22.840	0
5	2.127	0.746	4.257	14.368	0
6	2.000	0.702	4.244	21.341	0
7	0.674	0.279	1.899	19.880	0
8	2.368	0.862	4.641	15.698	0
9	0.600	0.238	1.655	25.949	0
10	0.546	0.250	0.954	9.341	0

According to the Table 4.4, the lowest peak time is seen during Test 1 at 0.544 s, and the slowest rise time occurs during Test 9 at 0.238 s. Test 10 has the lowest $OS\%$ at 9.341% and the shortest T_s at 0.954 s. Before selecting on the optimal value, all transient response values were analysed and compared to the ZN result to identify the best PSO outcome (i=30). Test 10 has less $OS\%$ and lower values of T_P, T_r and T_s than ZN, making it the best choice for PSO (i = 30).

4.7.2 PSO (50 iterations)

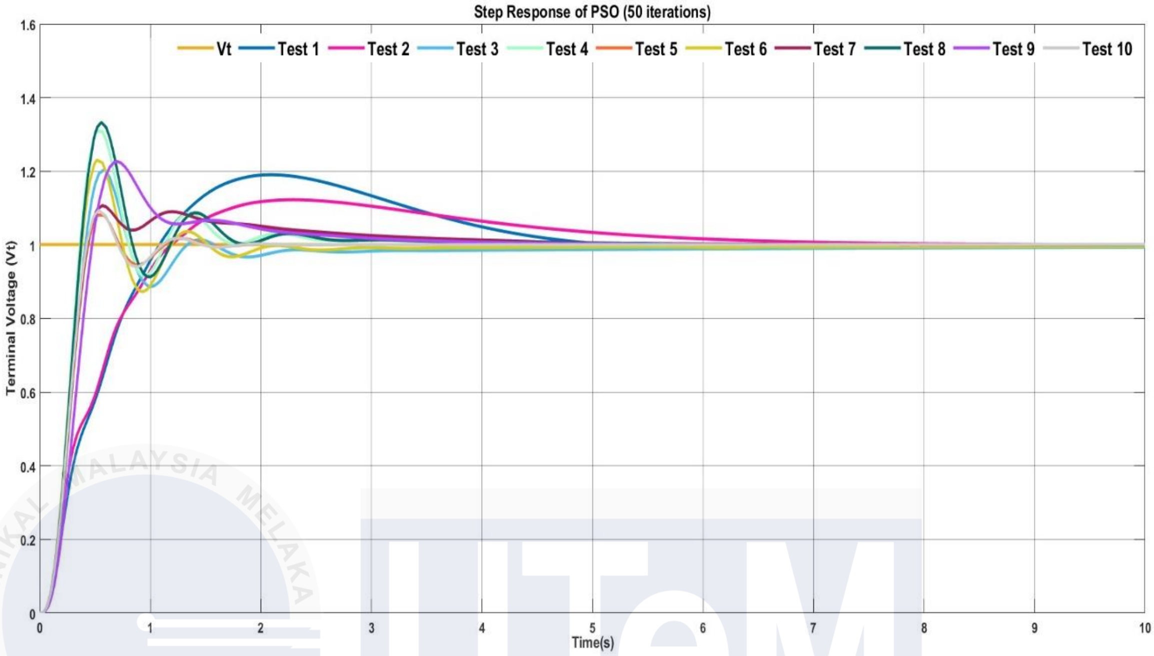

Figure 4.9 : Step response of AVR – PSO for 50 iterations

Table 4.5 : Result of AVR system with PSO (50 iteration)

	STEP INPUT = 1				
Test	T_p (s)	T_r (s)	T_s (s)	OS (%)	(e_{SS})
1	2.100	0.747	4.532	19.880	0
2	2.300	0.820	4.926	11.798	0
3	0.580	0.243	1.206	21.341	0
4	0.516	0.215	2.389	30.921	0
5	0.523	0.259	0.999	8.152	0
6	0.526	0.221	1.899	24.375	0
7	0.564	0.257	2.217	10.556	0
8	0.557	0.220	1.622	32.667	0
9	0.700	0.284	2.063	22.840	0
10	0.535	0.252	0.976	9.341	0

Based on the Table 4.6 , the lowest peak time and the slowest rise time occurs during Test 4 at 0.516 s and 0.215 s respectively. However, Test 4 can't be chosen as best optimal results because it has higher $OS\%$ and T_s compared to ZN results. The slowest settling time for 50 iterations occurs during Test 10 at 0.976 s, however the overshoot amount remains similar as for 30 iteration overshoot value. Test 5 was chosen as the most ideal value since it has the smallest overshoot of the ten tests, and its T_P, T_r and T_s are all lower than the ZN values.

4.7.3 PSO (100 iterations)

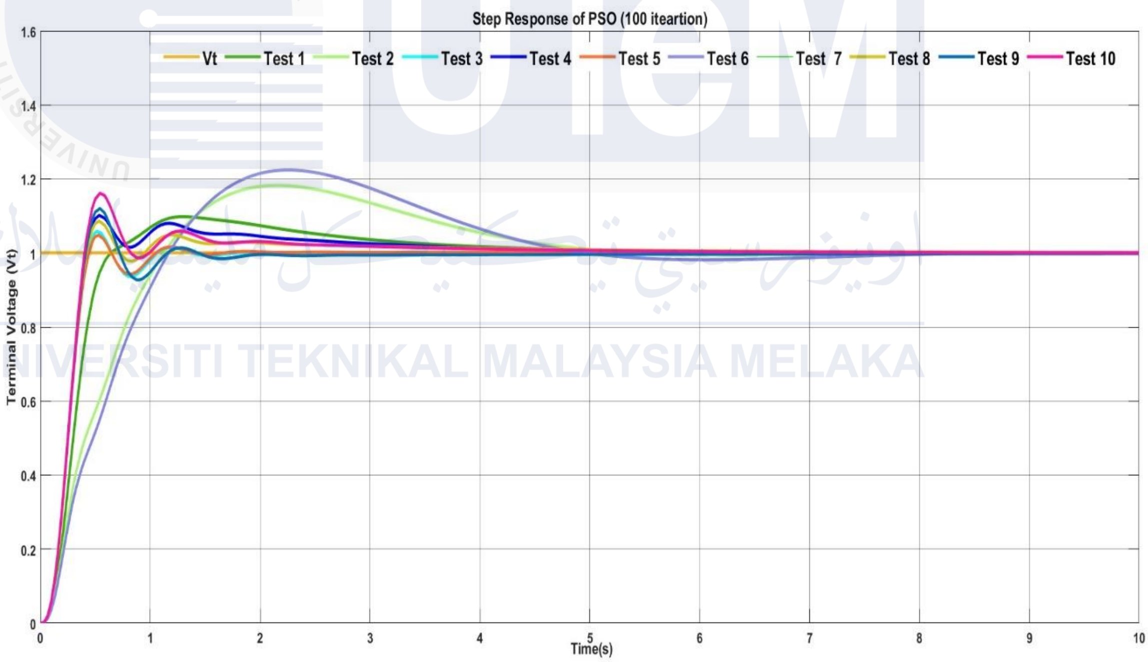

Figure 4.10 : Step response of AVR – PSO for 100 iterations

Table 4.6 : Result of AVR system with PSO (100 iteration)

	STEP INPUT = 1				
Test	T_p (s)	T_r (s)	T_s (s)	OS (%)	(e_{SS})
1	1.300	0.353	3.794	9.341	0
2	2.134	0.772	4.679	18.452	0
3	0.518	0.254	0.999	5.861	0
4	0.532	0.259	0.889	4.737	0
5	0.537	0.247	3.409	10.556	0
6	2.275	0.830	4.573	22.840	0
7	0.536	0.252	2.769	8.152	0
8	2.279	0.816	5.517	14.368	0
9	0.543	0.244	1.089	13.068	0
10	0.543	0.242	2.790	15.698	0

The above table displays the transient response values for PSO (i=100). The slowest peak time occurs during test 3 at 0.518 s, while the slowest T_r happens during Test 10 at 0.242 s. Despite having the slowest T_r, Test 10 cannot be picked as the optimal value since it has a higher $OS\%$ compared to the ZN results. Although, Test 3 shows lower transient response values than the ZN results, Test 4 has the lowest $OS\%$ at 4.737 %. Test 4 proven to be the best result for 100 iterations when compared and analysed with ZN results.

4.7.4 Comparison between AVR System with PSO for (30, 50 and 100 iterations)

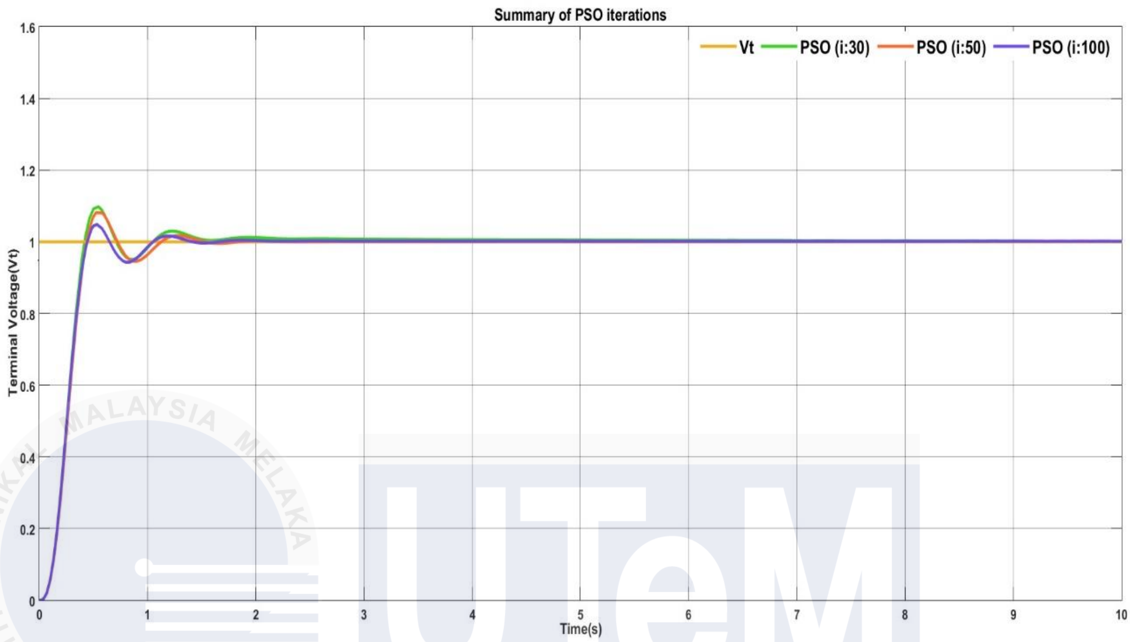

Figure 4.11 : Comparison between 30, 50 and 100 iterations

Table 4.7 : Summary of best transient response for iteration 30, 50 and 100

Iterations	T_p (s)	T_r (s)	T_s (s)	OS (%)	(e_{SS})	K_p	K_i	K_d
30 (10th)	0.546	0.250	0.954	9.341	0	1.9458	0.4741	0.3564
50 (5th)	0.523	0.260	0.999	8.152	0	1.8179	0.3704	0.3355
100 (4th)	0.532	0.259	0.889	4.737	0	1.9547	0.3785	0.3849

The Table 4.7 illustrates that only one test result was chosen from each of the 30, 50, and 100 iterations of PSO algorithm simulation. The selected results from each iteration were compared to determine the best optimal results for AVR system. The 30 iterations had the fastest rise time 0.250 s compared to the 50 and 100 iterations, but the T_s, T_p and $OS\%$ were all higher. Test 4 of the 100 iterations was selected as the best result. Although the T_P increased slightly during the 100 iterations compared to 50 iterations. The difference between the two iterations is only 0.009 s, therefore it has minimal effect. In comparison to the findings of the other two iterations, the 100 iterations showed faster T_s, shorter T_r and less $OS\%$.

4.8 Convergence curve for 100 iterations

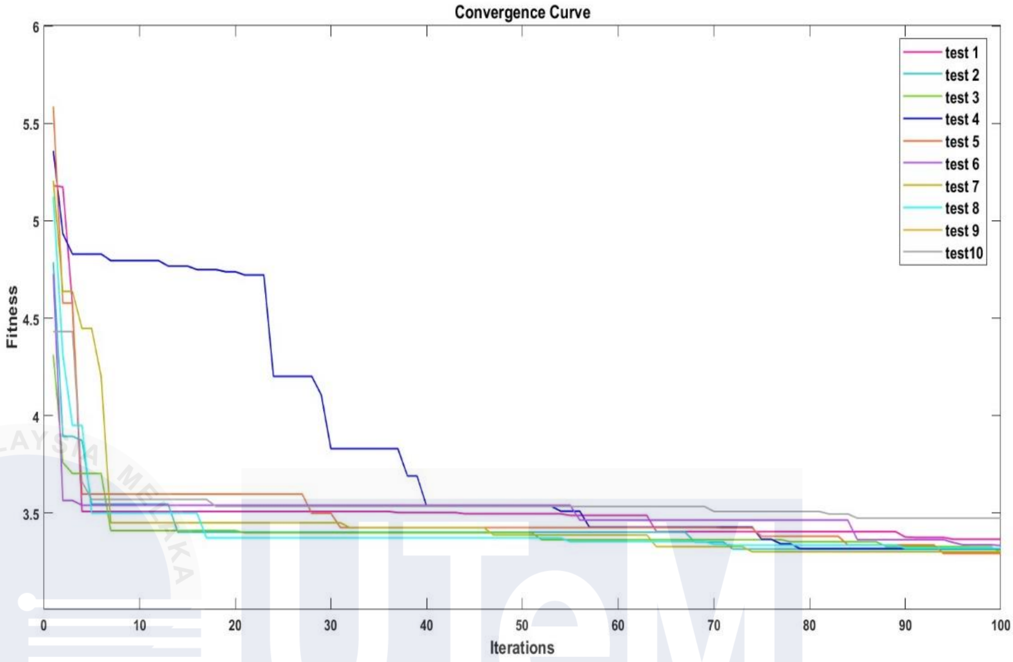

Figure 4.12 : Convergence Curve for 100 iterations

Figure 4.12 displays the optimization process taken to obtain the minimum objective function for all the particle of 100 iterations. The y-axis and x-axis represent the PSO's fitness function and the number of iterations respectively. The curve begins with a higher fitness function value since particles are randomly initiated in the search space. Test 4 was chosen as the best optimal result. Majority of the test indicates a rapid initial fall in the fitness function meanwhile Test 4 sees a significant decrease at first but then gradually declines, keeping a practically steady fitness function for more than 20 iterations.

4.9 Comparison between AVR System with PID (T-E) Tuning Methods, Ziegler-Nicholas method and PSO

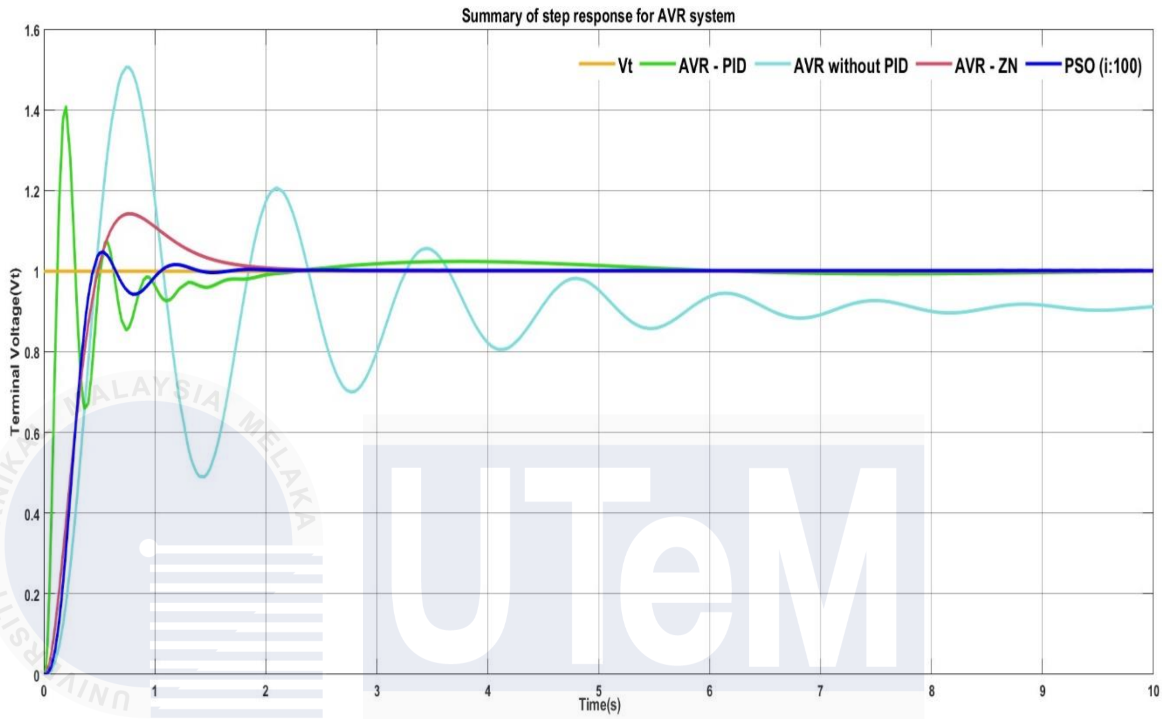

Figure 4.13 : Step response AVR system with PID tuning methods

Table 4.8 : Result of AVR system with PID tuning methods

	Tuning Methods			
Transient Responses	**Without PID**	**PID-Manual (T-E)**	**ZN (Ku=1.0)**	**PSO (i=100)**
T_p (s)	0.752	0.200	0.766s	0.532
T_r (s)	0.261	0.076	0.326	0.259
T_s (s)	6.99	4.48	1.65	0.889
OS (%)	65.7	42.143	14.2	4.737
(e_{ss})	0.1	0	0	0
K_p	-	0.8	0.6	1.9547
K_i	-	1	0.6335	0.3785
K_d	-	1.2	0.1421	0.3849

Table 4.8 shows all the results of AVR system with PID tuning methods obtained from the Figure 4.13. In FYP 1 ZN chosen as benchmark compared to PID (T-E) tuning method, despite having a higher T_p and T_r are than PID (T-E) tuning methods. This is because ZN produces 51.5% lesser overshoot than AVR system without a controller. Although having a shorter T_p and T_r, PID (T-E) tuning method can't be considered as the best optimal result because it doesn't have global optimum finding but rather lead to local optimum findings because human can't explore the entire parameter. Besides, it requires plenty of time as one has to manually adjust the parameters and its inefficient. ZN tuning method has provided more better and stable results compared to PID (T-E) tuning method. However, to obtain the best optimal result, a meta-heuristic method, known as the PSO algorithm method has been chosen in FYP 2. PSO method has a good capability for global searching in the solution space. Several numbers of iteration have been stimulated and the result were shown in Table 4.7. PSO (i =100) chosen as the best optimal results for AVR system as it produced more stable and robust results compared to AVR system without controller, AVR with PID (T-E) controller tuning method and AVR with ZN tuning method.

4.10 Robustness Analysis of the PID Tuning Techniques

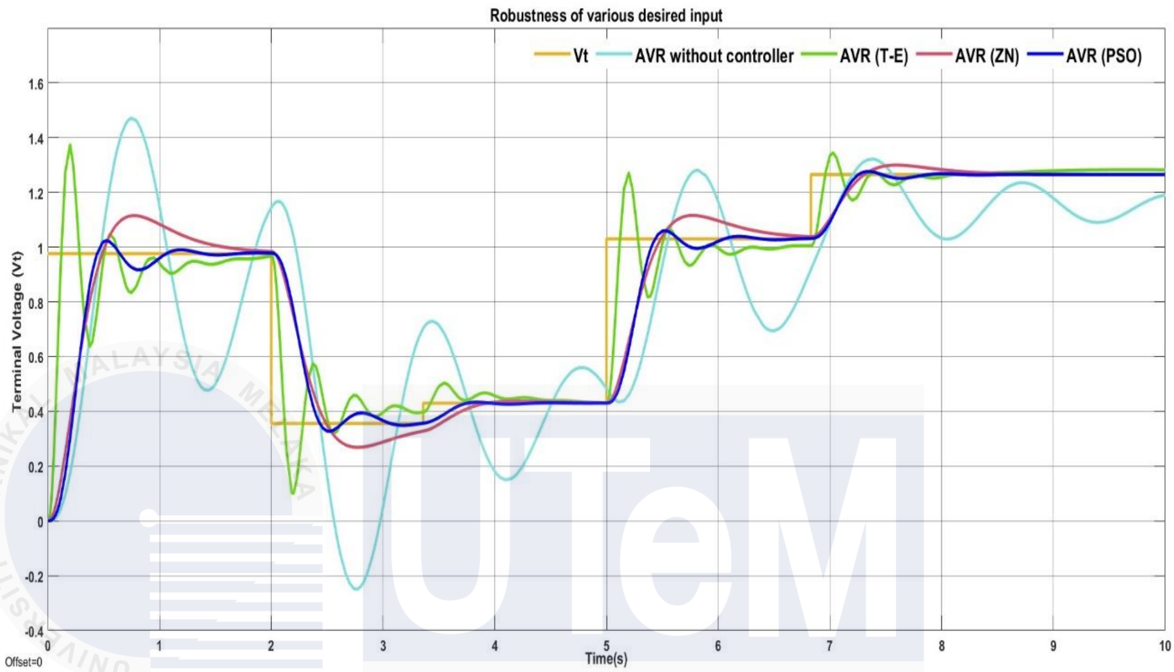

Figure 4.14 : Transient response of AVR system with various input

In this section, the robustness of PID tuning techniques has been simulated as shown in Figure 4.14. As observed the PSO (light blue) and ZN method (maroon) produce quite stable output compared with AVR without PID controller (purple) and PID (T-E) method (green) since it is able to achieve the T_s in each phase of input. In terms of $OS\%$, PSO (light blue) produces lower $OS\%$, compared to other tuning methods while AVR without controller and T-E method produce quite high of $OS\%$,. The PID (T-E) method is also not robust enough since it fails to achieve the desired level of terminal voltage for each value of input. The ZN method is considered succeeded because it was able to achieve the desired output. For PSO (light blue) produces quite stable desired output but still produces $OS\%$, however it is lesser than ZN. In conclusion, the PSO (light blue) is considered more robust than the conventional method since it can adapt to the input variation in a short period of time, as shown in Figure 4.14.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In conclusion, all the objectives are successfully executed as planned. This is accomplished by stimulating the AVR system model in MATLAB/Simulink without a controller and the stimulated results are compared to the journal results. Transient response of AVR without controller provides higher value in terms of T_p, T_r, T_s, $OS\%$ and e_{ss}. Furthermore, the AVR system with PID controller is stimulated using conventional methods such as PID (T-E) method and ZN tuning method. The PID (T-E) method involves manually modifying the PID parameters K_p, K_i and K_d to obtain an ideal system response. Besides, the Ziegler-Nichols method is implemented by finding the K_u and P_u value using RH criteria and characteristic equations and implementing the values in the given formula. The second goal was attained by employing the traditional procedure described above. According to the stimulation results, ZN outperforms PID (T-E) in terms of fast rising time and settling time. The ZN tuning method was used as a benchmark. The PSO algorithm was stimulated with three iterations. The best optimal transient response is achieved at 100 iterations compared to 30 and 50 iterations. As shown by the conducted simulation for all three iterations. The number of iterations increases, correspondingly increases the stability of the transient response. The chosen 100 iterations of PSO then compared with ZN result to determine optimal transient response result for AVR system. PSO (i:100) has been chosen as the ideal tuning approach in the presence of varying input voltages. By implementing the PSO, it provides optimal results compared to PID (T-E) and ZN tuning methods. The industries that use AVR system in their operations may benefit particularly due to the stability of the PSO technique. Lastly, it can be observed that the PSO tuning method is effective and robust under various tracking of input voltage.

5.2 FUTERE WORKS

It is necessary to improve the performance of conventional tuning techniques for further study and research. In this research, the methods that have been discussed have their advantages and disadvantages. The PID (T-E) methods should be investigated further to achieve the desired outcomes in a shorter period of time. Besides, for ZN method should be improved to produce a more suitable value of parameters. PSO algorithm can be improved through adjustments to the number of iterations and numbers of particles or by implementing different strategies such as hybridization to achieve more optimal transient responses. Other metaheuristic algorithms are also can be considered in the future projects.

REFERENCE

[1] L.Y. Ghamri, H. Awadh, N.A. Shamsi, S. AlKhateri, A. Khurram, and Rehman. *Robust AVR design for the synchronous generator*. The Journal of Engineering, 17 ,4111–4115, 2019.

[2] M. Furat and G. G. CuCu. *Design, implementation, and optimization of sliding mode controller for automatic voltage regulator system*. IEEE Access, 10, 55650-55674, 2022.

[3] M.J. Blondin, P. Sicard, and P.M. Pardalos. *Controller tuning approach with robustness, stability and dynamic criteria for the original AVR system*. Mathematicsand Computers in Simulation, 163, 168–182, 2019.

[4] Z. Bingul, and O. Karahan. *A novel performance criterion approach to optimum design of PID controller using cuckoo search algorithm for AVR system.* Journal ofthe Franklin Institute, 355, 5534-5559, 2018.

[5] Rakesh P. Borase, D. K. Maghade, S. Y. Sondkar, & S.N Pawar. *A review of PID control, tuning methods and applications.* International Journal of Dynamics and control, SpringerLink, 2020.

[6] C.T. Chao, N. Sutarna, J.S. Chiou, and C.J. Wang. *An optimal fuzzy PID controller design based on conventional PID control and nonlinear factors*. Applied Science, 9(6), 1224, 2019.

[7] M.G.H. Omran, and S. Al-Sharhan. *Improved continuous Ant Colony Optimization algorithms for real-world engineering optimization problems.* Engineering Applications of Artificial Intelligence, 85, 818-829, 2019.

[8] E. Köse, *Optimal control of AVR system with tree seed algorithm-based PID controller*. IEEE Access, 8, 89457-89467, 2020.

[9] A.M. Mosaad, M.A. Attia, and A.Y. Abdelaziz. *Whale optimization algorithm to tune PID and PIDA controllers on AVR system.* Ain Shams Engineering Journal, 10(4), 755-767, 2019.

[10] N. Pachauri. *Water cycle algorithm-based PID controller for AVR*. International Journal for Computation and Mathematics in Electrical and Electronic Engineering, 39(3), 551-567, 2020.

[11] Md.M. Gani, Md.S. Islam, and M.A. Ullah. *Optimal PID tuning for controlling the temperature of electric furnace by genetic algorithm*. SN Applied Sciences, 1(8), 2019.

[12] Kim Seng Chia. *Ziegler-Nichols based proportional-integral- derivative controller for a line tracking robot*. Indonesian Journal of Electrical Engineering and Computer Science, Institute of Advanced Engineering and Science, 2018.

[13] S. Bharat, A. Ganguly, R. Chatterjee, B. Basak, D. Sheet, and A. Ganguly. *A review on tuning methods for pid controller, Semantic Scholar*, 2019.

[14] Y.T. Sung, Chung, K.E. Chang and T.C. Liu. *Parameter optimization of nanosilver colloid prepared by electrical spark discharge method using ziegler-nichols method*. Journal of Physics and Chemistry of Solids, 148, 109650, 2021.

[15] A. Ghosh, A.K. Ray, Md. Nurujjaman, and Mo. Jamshidi. *Voltage and frequency control in conventional and pv integrated power systems by a particle swarm optimized ziegler–nichols based pid controller*. SN Applied Sciences, 3(3), 2021.

[16] F. Ahcene and H. Bentarzi, *Automatic voltage regulator design using particle swarm optimization technique,* 2020 International Conference on Electrical Engineering (ICEE), 1-6, 2020.

[17] A. P. Piotrowski, J. J. Napiorkowski, and A. E. Piotrowska. *Population size in Particle Swarm Optimization*. Swarm and Evolutionary Computation, 58, 100718, 2020.

[18] A. Lin, W. Sun, H. Yu, G. Wu, and H. Tang. *Global genetic learning particle swarm optimization with diversity enhancement by ring topology*. Swarm and Evolutionary Computation, 44, 571–583, 2019.

[19] J. Aberbour, M. Graba and A. Kheldoun, *Effect of cost function and pso topology selection on the optimum design of pid parameters for the avr system*. 4th International Conference on Electrical Engineering (ICEE), 1-5, 2015.

[20] M. Rahimian and K. Raahemifar, *Optimal pid controller design for avr system using particle swarm optimization algorithm*, 2011 24th Canadian Conference on Electrical and Computer Engineering(CCECE), 000337-000340, 2011.