AN APPLICATION OF MODIFIED ADAPTIVE BATS SONAR ALGORITHM (MABSA) ON FUZZY LOGIC CONTROLLER FOR DC MOTOR ACCURACY

NURAINAA BINTI ELAS

MASTER OF SCIENCE

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	:	NURAINAA BINTI ELIAS
Date of Birth	:	27 OCTOBER 1993
Title	:	AN APPLICATION OF MODIFIED ADAPTIVE BATS SONAR ALGORITHM (MABSA) ON FUZZY LOGIC CONTROLLER FOR DC MOTOR ACCURACY
Academic Session	:	SEMESTER 1 2020/2021

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☒	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

TS. DR. NAFRIZUAN BIN MAT YAHYA
PENSYARAH KANAN
FAKULTI TEKNOLOGI KEJURUTERAAN
PEMBUATAN & MEKATRONIK ,
UNIVERSITI MALAYSIA PAHANG
Tel: 09-424 5800 Faks: 09-424 5888

(Student's Signature)	(Supervisor's Signature)
931027-01-6148	DR. NAFRIZUAN MAT YAHYA
New IC/Passport Number	Name of Supervisor
Date:15 FEBRUARY 2021	Date: 15 FEBRUARY 2021

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree of Master of Science

(Supervisor's Signature)

Full Name : DR. NAFRIZUAN BIN MAT YAHYA

Position : SENIOR LECTURER

Date : 15 FEBRUARY 2021

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

____

(Student's Signature)

Full Name : NURAINAAA BINTI ELIAS

ID Number : MMI17001

Date : 15 FEBRUARY 2021

AN APPLICATION OF MODIFIED ADAPTIVE BATS SONAR ALGORITHM (MABSA) ON FUZZY LOGIC CONTROLLER FOR DC MOTOR ACCURACY

NURAINAA BINTI ELIAS

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Master of Science

Faculty of Manufacturing & Mechatronic Engineering Technology

UNIVERSITI MALAYSIA PAHANG

FEBRUARY 2021

ACKNOWLEDGEMENTS

Foremost, praise and thanks to the Almighty for granting me a lifetime opportunity and able to complete this research successfully. I am deeply indebted to my respected supervisor, Dr. Nafrizuan, for his motivation, encouragement, immerse knowledge and constructive criticism. His supervision and guidance led and helped me throughout the research.

I am very thankful to the staff of Faculty of Manufacturing and Mechatronic Engineering Technology of University Malaysia Pahang for providing the physical necessities, supplying information and resources during the research. It is an honor for me to appreciate the professors and educators for the knowledge, inspiration and life lessons that I have learnt.

Moreover, not to mention my family members in offering me massive supports in the aspects of moral, social, emotion and financial throughout the research years and this research certainly could not happen without entire reassurance from both of them. My committee members would be issued extra gratitude. I want to appreciate their findings and recommendations that are important for this study to be effectively completed. Last but not least, I want to acknowledge those whom direct and indirect encouragement has enabled me to complete my thesis on time. Only Allah may repay your benevolence.

ABSTRAK

Pengawal selalu digunakan untuk meningkatkan prestasi sistem kawalan. Kerja-kerja yang menggunakan pengawal menjadi perhatian para penyelidik kerana pengawal dapat diaplikasikan untuk menyelesaikan banyak masalah industri yang melibatkan kepantasan dan kedudukan. Pengawal logic kabur (FLC) terkenal kerana ia digunakan secara meluas dalam aplikasi industri. Walaubagaimanapun, struktur FLC masih kurang dari segi ketepatan dan tindak balas masa. Oleh itu, penyelidikan ini membincangkan tentang sistem FLC yang akan dioptimumkan oleh algoritma sonar kelawar adaptif (MABSA) untuk mengawal kedudukan motor servo DC. MABSA akan dioptimumkan dengan rangkaian input dalam sistem FLC yang akan direka. Tujuan kajian ini adalah untuk mencapai ketepatan sambil meminimumkan tindak balas masa motor servo DC. Ini dilakukan dengan mereka bentuk FLC menggunakan perisian Matlab. Setelah FLC direkabentuk sepenuhnya, gambarajah blok Simulink untuk motor servo DC dan FLC akan dibina untuk menganalisis prestasi pengawal. Julat fungsi untuk input dan output akan dioptimumkan oleh MABSA untuk mendapatkan nilai kedudukan terbaik. Prestasi FLC yang dibangunkan dengan MABSA yang dioptimumkan disahkan melalui ujian simulasi dan ketahanan dengan sistem yang tidak menggunakan FLC dan juga sistem tanpa MABSA. Hasil kajian mendapati bahawa FLC yang dicadangkan dengan pengoptimuman algoritma MABSA mampu menghasilkan peningkatan 3.8% sehubungan dengan waktu kenaikan dibandingkan dengan skema kawalan lain yang dinilai. Semasa membandingkan dengan algoritma PSO, FLC yang dicadangkan yang dioptimumkan oleh MABSA menunjukkan peningkatan sebanyak 12.5% pada waktu kenaikan dan 10% dalam masa penyelesaian. Kesimpulannya, hasilnya mengesahkan prestasi yang lebih baik dari segi waktu kenaikan dan waktu penyelesaian FLC yang dicadangkan yang telah dioptimumkan oleh MABSA.

ABSTRACT

Controllers are mostly used to improve the control system performance. The works related to controllers attract researchers since the controller can be applied to solve many industrial problems involving speed and position. Fuzzy logic controller (FLC) gains popularity since it is widely used in industrial application. However, the FLC structure is still lacking in terms of the accuracy and time response. Although there are optimization technique used to obtain both accuracy and time response, it is still lacking. Therefore, this research presents works on the FLC system which is the fuzzy inference system that will be optimized by the modified adaptive bats sonar algorithm (MABSA) for the DC servo motor position control. The MABSA will be optimized with the range of the membership input in the FLC. The research aims are to achieve accuracy while minimizing the time response of the DC servo motor. This is done by designing the FLC using the Matlab toolbox. After the FLC is designed completely, the Simulink block diagram for the DC servo motor and FLC are built to see the performance of the controller. The range of the membership function for inputs and outputs will be optimized by the MABSA to get the best positional values. The performance of the developed FLC with the optimized MABSA is verified through the simulation and robustness tests with the system that did not use the FLC and also the system without MABSA. It was demonstrated from the study that the proposed FLC with optimization of MABSA algorithm was able to yield an improvement of 3.8% with respect to the rise time in comparison to other control schemes evaluated. When compared with PSO algorithm, proposed FLC optimized by MABSA showed improvement by 12.5% in rise time and 10% in settling time. PSO-FLC also give 0.6% steady state error compared to the MABSA-FLC. In conclusion, the results validate the better performance in terms of rise time and settling time of the developed FLC that has been optimized by the MABSA.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

L_a	Armature Inductance
v_a	Armature Voltage
R_a	Armature Resistance
I_a	Armature Current
E_b	Back Emf
w	Angular Speed
T_m	Motor Torque
θ	Angular Position of Rotor Shaft
J_m	Rotor Inertia
B_m	Viscous Friction Coefficient
K_T	Torque Constant
K_b	Back Emf Constant
Δe	Error
ce	Change of Error
T_r	Rise Time
T_s	Settling Time

LIST OF ABBREVIATIONS

ABC	Artificial Bee Colony
ABSA	Adaptive Bats Sonar Algorithm
AC	Alternating Current
BA	Bat Algorithm
BAST	Bats Algorithm Based Scheduling Tool
BSA	Bats Sonar Algorithm
CS	Cuckoo Search
DC	Direct Current
EBA	Evolved Bat Algorithm
FIS	Fuzzy Inference System
FLC	Fuzzy Logic Controller
GA	Genetic Algorithm
HBA	Hybridized Bat Algorithm
MABSA	Modified Adaptive Bat Sonar Algorithm
MOBA	Multi Objective Bat Algorithm
NL	Negative Large
NM	Negative Medium
NS	Negative Small
PL	Positive Large
PM	Positive Medium
PID	Proportional Integral Derivative
PS	Positive Small
PSO	Particle Swarm Optimization
PWM	Pulse Width Modulation
ZE	Zero

CHAPTER 1

INTRODUCTION

1.1 Introduction

This chapter introduces an overview of the research that has been carried out. Firstly it discusses the background of research and also the problem statement. This chapter explains briefly the overall description of the research and highlight the problem faced in the industry nowadays. Then, the objectives and scope of this research are defined and finally flow of the research process is formulated.

1.2 Research Background

In recent years, Industry Revolution 4.0 (IR 4.0) which is also known as the Fourth Industrial Revolution is introduced to a new stage of technological change where a highly automated, interconnected and smart production processes is developed (Peters and Yang, 2007). With the introduction of machine, driver, actuator, and sensor especially in factory, a new level of optimization and productivity is applied (Murray, 2017). IR 4.0 is also an initiative with technology revolution like big data, automation, cloud computing, connection and electric vehicles (Xu et al., 2018). In the context of IR 4.0, there are also advanced technologies in the smaller manufacturing industry but successfully improving product quality, maximizing production, increasing preventative maintenance, and reducing downtime (Li et al., 2017).

One of the categories of manufacturing and electrical industry that is relevance to the IR 4.0 is the use of electric vehicles (Mosterman and Zander, 2016; Chan and Wong, 2004). Electric vehicles are not limited to electric cars but also include electric aircraft, road and rail vehicles, manufacturing forklift trucks and industry crane (Rad et al., 2011). Since the term IR 4.0 represents the advanced connectivity for machines in manufacturing with the aid of electric vehicles, the electric motors also acts as a key technology of that current trends (Hidrue et al., 2011).

For the electric vehicles, it is control by an electric motor that is running by utilizing power put away in the batteries (Bhatt et al., 2019). Relating with the developing of revolution in electric vehicle, it has come out to be essential to obtain a far reaching comprehension of the criteria linked in regulation of electric motors (Mayr et al., 2018). It is discovered that the implementation of electric motor has been varied from an industry to another. As a result, various types of electric motors are presently used based on the power needed (Yin et al., 2018). There are many types of electric motors such as Alternating Current (AC) motor, Direct Current (DC) motor, linear motors, servo motors, and steppers motors, however DC motors are often used in the manufacturing industry (Tomar et al., 2014). Many applications such as robot manipulator, electric trains and electric vehicles use DC motors especially when speed and position control is required (George, 2008).

DC motors become elements of movement which obtain electrical power in terms of direct current and transform the current into the mechanical rotation (Meshram and Kanojiya, 2012). Nearly all forms of DC motors have some inner function, namely electromechanical or electrical. The reason is that the DC motor system is essential especially when exactness positioning as well as speed control is required (Huh and Lee, 2018). DC motors utilize feedback controller for speed or position control, or both. There were also several types of DC motors including brushed, brushless, and servo motors. However, the DC motor that is widely used in a variety of application such as industrial electronics and robotics is DC servo motor (Meike and Ribickis, 2011).

More technologies starting to use DC servo motor because of the suitability and efficiency that can be used for varies purposes (Shanmugasundram, et al., 2012). DC servo motor can be equipped with controller to regulate speed, controlling position and also protecting the motor from overloads and faults (Dimeas et al., 2017). With the introduction of a controller to the motor, faster response can be generated (Szabat and Orlowska-Kowalska, 2007).

There are various types of controller use in DC servo motor for example Proportional–Integral-Derivative (PID) controller, lead controller, lag controller, fuzzy logic controller (FLC) and neural network controller. However, PID controller is by far the most prominent and commonly utilized type of controller across manufacturing sector. PID is often used since it has accurate set point and fast reaction to disturbances (Verna et al., 2013). Even so, PID controllers do not give adequate outcomes whenever it needs adaptive algorithms but FLC offers some solutions (Al-Odienat and Al- Lawama, 2008). Basic advantage of FLC is it would not involve fully knowledge and understanding of an entire mathematical structure model (Nanda and Mangla, 2004). FLC's success is demonstrated by the fact that the algorithm manages the direct and easy execution of human reasoning.

However, the current situation is that the performance of FLC cannot be more precise since there are problems faced like the interruption of fuzzy inference system when controlling the input and output of the desired point/position (Chang and Chang, 2006). Fuzzy inference system problem that cannot achieve the motion and stabilization control of desired point/ position becomes an interesting issue to investigate. Fuzzy inference system which are fuzzification, membership function, rule-based and deffuzification cannot function properly when there have to optimize 2 parameters (Subramanian et al., 2013). Since FLC is designed automatically by Matlab toolbox, the optimization on input or output parameters will be difficult (Rong et al., 2011).

1.3 Problem Statement

The issue with most of the fuzzy inference system is that the performance is still lacking since it cannot achieve best accuracy while minimizing time response (Prabu et al., 2016). Besides, excellent balance between these two characteristics in the FLC can be beneficial for certain process (Kusagur et al., 2010). Therefore, this problem must be tackled so that the FLC can be used effectively for controlling any type of DC servo motor position system.

(Rahmani et al., 2012) designed fuzzy logic controller optimized by particle swarm optimization for DC motor speed control. The result is that the designed FLC-PSO speed controller obtains much better dynamic behaviour compared to PID. However, the designed FLC-PSO faced problem with the rise time. Another approaches to solve the problem with the optimization is by designing fuzzy logic controller by particle swarm optimization for wind turbine (Bachache and Wen, 2013). The simulation results demonstrate that the Optimized Fuzzy Logic Control (OFLC) gets a better parameters of fuzzy sets using PSO, and realizes a good dynamic behaviour compared with conventional FLC. But the PSO algorithm still give steady state error in the rotation of the wind turbine.

(Manikandan and Arulmozhiyal, 2014) developed the fuzzy logic controller for controlling the position of DC servo motor drive. The position of DC motor can be controlled and return back to desired value easily but the settling time is very still high. Besides that, (Yadaz, 2015) suggested to control the position of DC Motor by using Fuzzy Logic Controller (FLC) with MATLAB application and comparing with conventional PID control. In spite of the easy implementation of traditional control "PID", its response is not so good for non-linear systems. The improvement is remarkable when controls with fuzzy logic are used, obtaining a better dynamic response from the system.

Researcher Premkumar and Manikandan, 2018 tried approaches that optimization of fuzzy controller is carried out using nature inspired optimization algorithms such as particle swarm, cuckoo search, and bat algorithms. The bat optimized fuzzy proportional derivative controller has superior performance than the other optimization considered. However, optimized fuzzy still has noise at rise time.

(Subramanian et al., 2014) reported that several methods have been suggested in recent years to address the issue with the fuzzy inference system efficiently. However, the technique still cannot solve the main problem with the fuzzy inference system (Koçak et al., 2018) (Uraon and Kumar, 2016) (Wieczorek, 2018) (Vyas et al., 2015). Subramanian et al. (2014) stressed that many algorithm to optimize the fuzzy inference system was suggested too for tackling the problems facing by most FLC. But, all proposed methods cannot optimize the accurate position while minimizing time response simultaneously (Premkumar and Manikandan, 2015). Since new optimization algorithms are introduced at almost yearly, many optimization problems have been solved (Mahdavi et al., 2007). Thus, the fuzzy inference system shall take the opportunity to utilize this kind of technique for producing a better trade-offs between accuracy and time response characteristics of FLC.

For the optimization of the DC servo motor position control using FLC, the MABSA is chosen to be used. The main reason is because there is no method that used MABSA for tackling the problem with the fuzzy inference system problem. MABSA also performs better in terms of accuracy and convergence speed for optimization problem compared to other several existing algorithms. Hence, this research investigates whether the DC servo motor can achieve accurate positioning while minimizing time response by adopting FLC optimized by MABSA on its control system.

1.4 Research Objectives

The three objectives of this research are:

1. To design and develop an optimization technique by applying modified adaptive bats sonar algorithm on fuzzy logic controller for DC motor accuracy.
2. To compare the performance of the proposed optimization technique by applying modified adaptive bats sonar algorithm on fuzzy logic controller for DC motor accuracy with PSO algorithm.
3. To analyse the performance of the proposed optimization technique fuzzy logic controller optimized by the modified adaptive bats sonar algorithm for DC servo motor position control when compared with PSO algorithm.

1.5 Research Scope

In order to achieve the research objectives, research scopes have been identified:

1. The designing and validation performance of fuzzy logic controller (FLC) by MABSA for DC motor control using MATLAB/Simulink software.
2. The FLC is designed using the Matlab toolbox.
3. The type of DC motor used is DC servo motor.
4. The transfer function use for DC servo motor modelling is based on mathematic calculation and design by Simulink transfer function block diagram.

5. The comparison of response system only using standard MABSA and PSO.

6. The optimization value using MABSA and PSO are in offline method and iteration value only run for 30 times.

7. This research only focuses on the DC servo motor position control using fuzzy controller. Input signal use is step, sine-wave and pulse generator and output will be the step response.

8. This performance of proposed FLC is comparing by using optimization with PSO algorithm only. Another type of controller and algorithm is not part of this research.

1.6 Thesis Organisation

The thesis is structured as follows:

Chapter 1: Introduces the background and the problem statement of the research, research objectives, research scope, research methodology, research contribution and publications and lastly the organisation of the thesis.

Chapter 2: Discusses the varies types and methods of modelling DC servo motor, structures of fuzzy logic controller (FLC) and the modified adaptive bats sonar algorithm (MABSA) in the perspective of the research literature and background knowledge.

Chapter 3: Elaborates in detail the modelling of DC servo motor transfer function, the designing of FLC and the optimization of design FLC with MABSA.

Chapter 4: Deliberates the simulation result of proposed FLC optimized by MABSA and compared with the experimental result.

Chapter 5: Presents the conclusions of the research as well as the recommendation for the future work of the research.

1.7 Summary

This chapter has discussed the background of the research, highlighted the problem statement and identified the objectives and scopes of the research. Besides that, research methodology has been planned so that the flow of work can be done smoothly.

For the next chapter, more explanation on the DC servo motor, fuzzy logic controller and the modified adaptive bats sonar algorithm will be discussed.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter is divided into several sections. For the first section, it covers about the optimization technique by using the modified adaptive bats sonar algorithm (MABSA). Second section discusses the fuzzy logic controller (FLC). This section reviews the application of FLC and how FLC is able to improve the accuracy of position. The third section is all DC motor including the types and transfer function of DC servo motor. The application of MABSA with FLC is explained in details in this section.

2.2 Optimization

Optimization is the proses to maximize or minimise some function to some set, often representing a range of choices available in a certain situation. The function allows comparison of the different choices for determining which might be "best." (Cheng et al., 2015). Optimization commonly be used in applications such as for the minimal cost, maximal profit, minimal error, optimal design, optimal management, variation principles. In optimization of a design, the design objective could be simply to minimize the cost of production or to maximize the efficiency of production (Cheng et al., 2015). An optimization algorithm is a procedure which is executed iteratively by comparing various solutions till an optimum or a satisfactory solution is found. With the advent of computers, optimization has become a part of computer-aided design activities (Rini and Yuhaniz, 2011).

Optimization by algorithm is widely be used since there are many advantages of the system or process in terms of faster response or slower response depends on the users requirement (Mirjalili, 2016). There are two distinct types of optimization algorithms widely used today. The first one is the Deterministic Algorithms. They use specific rules for moving one solution to other. These algorithms are in use to suite some times and have been successfully applied for many engineering design problems (Kelner, 2008). The second is the Stochastic Algorithms. The stochastic algorithms are in nature with probabilistic translation rules. These are gaining popularity due to certain properties which deterministic algorithms do not have (Kelner, 2008).

2.2.1 Optimal Problem Formulation

A naive optimal design is achieved by comparing a few (limited up to ten or so) alternative solutions created by using a priori problem knowledge. In this method feasibility of each design solution is first investigated (Coello, 2000). Thereafter an estimate of underlying objective for example cost and profit of each solution is compared and best solution is adopted. Figure 2.1 shows an outline of the steps usually involved in an optimal design formulation.

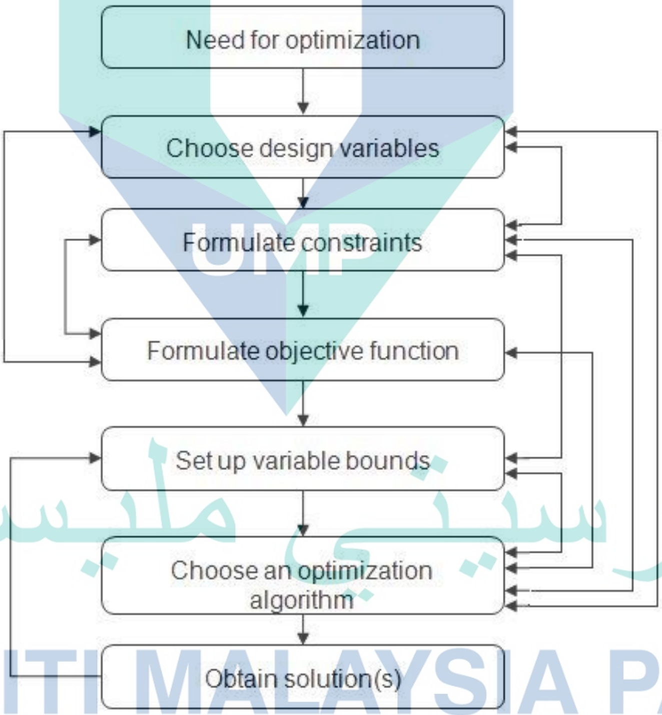

Figure 2.1 A flowchart of the optimal design procedure
Source: Mirjalili (2016).

It is impossible to apply single formulation procedure for all engineering design problems, since the objective in a design problem and associated therefore, design parameters vary product to product different techniques are used in different problems

(Rini and Yuhaniz, 2011). Purpose of formulation is to create a mathematical model of the optimal design problem, which then can be solved using an optimization algorithm.

The formulation of an optimization problem begins with identifying the underlying design variables, which are primarily varied during the optimization process (Kelner, 2008). A design problem usually involves many design parameters, of which some are highly sensitive to the proper working of the design. These parameters are called design variables in the parlance of optimization procedures. Other (not so important) design parameters usually remain fixed or vary in relation to the design variables. The first thumb rule of the formulation of an optimization problem is to choose as few design variables as possible (Kelner, 2008). The outcome of that optimization procedure may indicate whether to include more design variables in a revised formulation or to replace some previously considered design variables with new design variables (Mirjalili, 2016).

The constraints represent some functional relationships among the design variables and other design parameters satisfying certain physical phenomenon and certain resource limitations (Coello, 2000). The nature and number of constraints to be included in the formulation depend on the user. Constraints may have exact mathematical expressions or not. For example, maximum stress is a constraint of a structure (Coello, 2000). If a structure has regular shape they have an exact mathematical relation of maximum stress with dimensions. But in case irregular shape, finite element simulation software may be necessary to compute the maximum stress.

The next task in the formulation procedure is to find the objective function in terms of the design variables and other problem parameters (Cheng *et al.*, 2015). The common engineering objectives involve minimization of overall cost of manufacturing or minimization of overall weight of a component or maximization of total life of a product or others. Although most of the objectives can be quantified (expressed in mathematical form), there are some objectives (such as aesthetic aspect of a design, ride characteristics of a car suspension design and reliability of a design) that may not be possible to formulate mathematically (Coello, 2000). In such a case an approximating mathematical expression is used.

The final task of the formulation procedure is to set the minimum and the maximum bounds on each design variable (Coello, 2000). Certain optimization algorithms do not require this information. In these problems, the constraints completely surround the feasible region. Other problems require the search algorithm with in these bounds (Kelner, 2008).

2.2.2 Swarm Intelligence Algorithm

Swarm intelligence algorithms are getting popular in the optimization fields since engineering design problem can be solved through it (Janga Reddy and Nagesh Kumar, 2007). Swarm intelligence algorithms for example particle swarm optimization (PSO), fish swarm optimization, artificial bee colony (ABC), bats algorithm (BA) and whale optimization algorithm (WOA) have been proven as a better methods for handling difficult optimization problems (Mavrovouniotis. et al., 2017). Nonetheless, the smart behaviours of bats echolocation have motivated many researchers to create new algorithms, particularly over the last decade (Tsai et al., 2012).

Within the last two decades, swarm intelligence algorithms have been recognised in the field of optimisation (Karaboga and Akay, 2009). Swarm intelligence algorithms are influenced by the behaviours of numerous animal and insect swarms, like birds, rats, bees, fish and bats (Lope and Coelho, 2005). Researchers also studied animal, plant, and human behaviours, examined the underlying force behind the phenomenon, and motivated researchers towards create different forms of algorithms. It began with the particle swarm optimization (PSO) suggested by Kennedy and Eberhart (1995) which represented fish or bird swarms behaviour. In 1999, Dorigo and Di Caro pioneered ant colony algorithm (ACO) and then artificial bee colony (ABC) were suggested by (Karaboga and Basturk, 2007) which observe the forage of ants and bees respectively.

Then, bat sonar algorithm (BSA) was proposed by inspiration from the intelligence features in behaviours of bats (Tawfeeq, 2012). From BSA, adaptive bats sonar algorithm (ABSA) was introduced for solving unconstrained single objective optimisation problems (Yahya et al., 2016). After that, a new improved version that can solve constrained optimization problem called modified adaptive bats sonar algorithm (MABSA) was proposed (Yahya and Tokhi, 2017).

Bat algorithm, proposed by Yang (2010) is inspired by the echolocation behaviour of bats when finding their prey, hunting, and avoiding obstacles. The advantages of BA are to provide a better optimum solution for a specific nonlinear problems, excellent global optimization ability, and less control parameters compared to other existing swarm intelligence algorithms (Gandomiet al., 2013). As the BA has a clear and specific definition, ease of implementation and rapid convergence, it has currently received significant interest and wide-ranging implementations in multiple areas (Dashtiet al., 2010).

The application of BA has been utilised in solving various engineering problem especially in ergonomic research for example by Raghav et al. (2011). BA with fuzzy modification is used to quickly screen the workplaces of the company that have high ergonomic risk. Besides that, BA also indicates a possible solution for full human body pose estimation in video sequences (Akhtar et al., 2012). Other applications that embedded BA in manufacturing is the development of the bat algorithm based scheduling tool (BAST) that is used to solve multi-stage multi-machine multi-product scheduling problems (Malakooti et al., 2012). The algorithm is aimed to minimise the combination of earliness and tardiness penalty costs. BA has also been used in electrical and electronic areas such as using BA for controlling speed of brushless DC motor (Premkumar and Manikandan, 2015).

Apart from bat algorithm (BA), another significance bats-based algorithm was bats sonar algorithm (BSA) by Tawfeeq (2012). BSA is inspired from echolocation process of a colony of the bats. A bat's sonar is an efficient echolocation device. The bat sonar evokes knowledge regarding the target's relative velocity, the scale of the target's various characteristics and the target's height for learning the range of a target (Denny, 2004). However, some drawbacks have been detected in BSA such as not efficient in making a searching process due to small number of bats and exists the possibility of redundancy location (Denny, 2004). Therefore, in order to modify the shortcomings of the BSA based on the nature of echolocation of bats, an adaptive bats sonar algorithm (ABSA) by Yahya et al.(2016) is presented. The purpose of ABSA is to improve precision, accuracy and convergence rate of BSA for single objective optimisation problem.

The application of ABSA can be seen by (Yahya and Osman Zahid, 2016) where a practical business optimization problem for a single objective needs to be solved. The outcomes is that ABSA can obtain the optimal parameters which are the cost optimization of shipping refined oil and profit optimization of selling television sets (Yahya and Osman Zahid, 2016). However there is a limitation in ABSA especially when dealing with a constrained single objective optimization problems (Yahya et al., 2016). Therefore, a new algorithm called a modified adaptive bats sonar algorithm (MABSA) is proposed.

Through redefining some elements in ABSA and even reformulating the key feature of BSA (Yahya, 2016), MABSA will be able to address constrained optimisation problems. MABSA has been used to overcome vehicle side effect design weight optimisation. Besides, MABSA is also efficient to optimize brushless wheel DC motor performance (Yahya and Tokhi, 2017). A study by (Yahya and Tokhi, 2017) proposed MABSA for solving the constrained optimisation problems coupled with penalty function method as constraint handling technique.

The performance of the algorithm is verified through rigorous tests with four constrained optimisation benchmark test functions. The acquired results show that the proposed algorithm performs better to find optimum solution in terms of accuracy and convergence speed. The statistical results of MABSA to solve all the test functions also has been compared with the results from several existing algorithms taken from literature on similar test functions. The outcomes from the findings displayed that MABSA outperforms other establish algorithms, and thus, it can be an efficient alternative method in the solving constrained optimisation problems.

2.2.3 Modified Adaptive Bats Sonar Algorithm (MABSA)

The MABSA is developed after the initial ABSA has updated three search techniques and introduced a new element to it. The three procedures are on how to set the beam length (L), decide the starting angle (θ_m) and angle between beams (θ_i) and also measure the end point position (pos_i). The bounce back method, on the other hand, is a new feature which may have been implemented in the MABSA and was not previously included in ABSA. The other MABSA components will not be addressed further though, as they are close to the ABSA described in Yahya et al. (2016). The latest L is setup in MABSA as:

$$L = Randx\ (\frac{SS_{size}}{10\%\ xBats})$$

Where the solution range (SS_{size}) is the value between the upper search space (SS_{max}) limit and the lower search space (SS_{min}) limit. Every dimension (Dim) has its specific or known as Dim constraints. The selection of solutions is categorized into micron size, such as 10% of the overall search space bats population. The percentage of each bat's search space size is labelled as practicable for transmitting sound without collision with each other. The random value of L is given to allow actual difference in the beam lengths of increasing the number of beams (NBeam) at each Dim (but staying within the Dim constraints) at each iteration. This emphasis forces each bat to look for a greater perimeter each time with the potential to diversify the search strategies through iterations and therefore discover the best global approach which can be near to them.

Each NBeam with L is emitted from specific angle location. In the ABSA, the θ_m and θ_i are determined randomly in every iteration. Thus all bats would transmit the NBeam from a collection of equivalent angle position at each iteration. For incorporating another randomisation character within MABSA, θm and θi will be calculated arbitrarily and independently for each bat at each iteration. Thus at each iteration, each bat would transmit the NBeam from a specific set of angle position. Consequently, this randomization would also lead to the diversification of the search phase at MABSA.

2.3 Fuzzy Logic Controller

As a distinction between PID and FLC for servomotor control, PID parameters have to be adjusted again under differences of plant variables or noise whereas FLC parameters need not to be defined and adjusted (Maraiya and Ray, 2018). Fuzzy logic is utilized in a variety of controls, since it does not allow a specific device model to be managed (Monmasson and Cirstea, 2007). Fuzzy logic operates by implementing rules which equate controller inputs with the desired output. The subsequent topic is regarding the Fuzzy logic's main features.

FLC has several benefits compared to other traditional controls, such as access flexibility, low cost and the potential to build without understanding the specific mathematical model of the mechanism (Nanda and Mangla, 2004). Fuzzy logic provides an innovative form of thought that facilitates complex system simulation utilizing greater degrees of abstraction from information and practice. Fuzzy logic may be defined literally as 'word computation instead of counting' and 'sentence control instead of equation control' (Monmasson and Cirstea, 2007).

Fuzzy Logic Controller (FLC) is focused on rational thinking and is a way to turn the linguistic regulation technique into an automated one by creating a rule based and regulates the system's behaviour (Nanda and Mangla, 2004). Fuzzy offers a surprisingly clear means of drawing definitive conclusions from unclear, uncertain or imprecise details. It is ideal for applications such as DC motor speed regulation with non-linearity (Yen and Pfluger, 1995). Fuzzy logic method was developed in 1965, by Zadeh, and is a mathematical tool to deal with uncertainty (Passino *et al.*, 1998). It offers a method for coping with the complexity of inaccuracies and details. Fuzzy logic offers a structure of inference which allows for reasonable human thinking capability.

2.3.1 Method of Fuzzy Inference System

There are two different methods of fuzzy inference system (FIS) which are Mamdani FIS and Takagi-Sugeno Fuzzy model (TS Method). The different method will provide different consequent of fuzzy rules.

2.3.1.1 Mamdani Fuzzy Inference System

Mamdani Fuzzy Inference system is developed for synthesizing a set of linguistic control rules collected from expert human operators in fuzzy control system (Mamdani and Assilian, 1999). From the findings on the fuzzy algorithms for complex systems and decision making, Mamdani method is widely used for understanding the fuzzy logic works. Since the output from Mamdani FIS is easier for understanding the rule bases, researchers recommended using this method for decision support application (Hamam and Georganas, 2008). Mamdani is more suited to human input because it entails a substantial computational burden (Kaur and Kaur, 2012).

The detailed process of the Mamdani FIS is shown in Figure 2.2. The two inputs x and y is applied to the two rule Mamdani FIS for deriving the overall output, z.

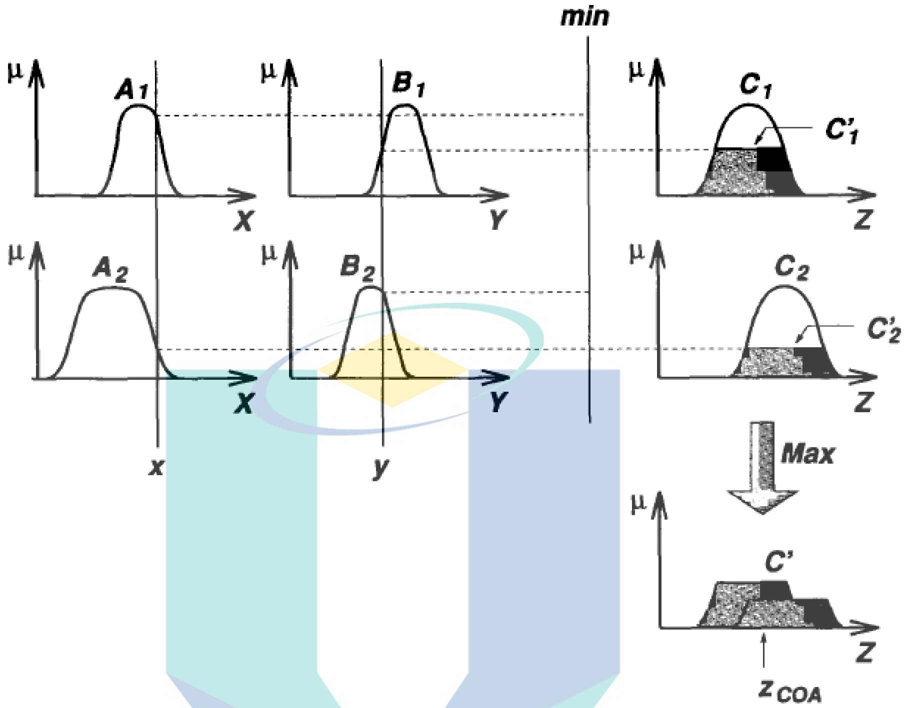

Figure 2.2 Mamdani fuzzy inference system model
Source: Kothamasu and Huang (2007).

The Mamdani FIS uses the original min and max composition for producing the output result (Kothamasu and Huang, 2007). The fuzzy reasoning behind the output is because of the scaled down of each rule in the fuzzy set via algebraic reduction.

The advantages of using Mamdani FIS are easy to interpreted and also easy to be formalized since the output can transform to a linguistic structure before defuzzification process (Blej and Azizi, 2016). Besides that, the output obtained is reasonable based on the relatively simple design. Mamdani FIS can also be used for multiple input single output (MISO) system and multiple input multiple output (MIMO) system (Zaher*et al.*, 2014).

2.3.1.2 Takagi-Sugeno Fuzzy Model (TS Method)

Takagi-Sugeno fuzzy model (TS method) is proposed in 1985 by Takagi, Sugeno and Kang. TS method is produced only one output membership functions that are either a linear or constant based on the input value (Wang and Chen, 2014). A common fuzzy rule in TS method is in the form:

$$if\ x\ is\ A\ and\ y\ is\ B\ then\ z = f\ (x, y)$$

Where A and B are the input variables and z is the output variable that indicates the appropriate function suitable for the fuzzy region. The output variable f(x,y) can be either zero-order Sugeno fuzzy model or a first-order polynomial depending on the rule based that have been proposed (Cao and Frank, 2001). Since the zero-order Sugeno model mostly will not happen and can be viewed as special case of the Mamdani FIS or Tsukamoto fuzzy model (Abonyi*et al.*, 2002), this model is neglected. Therefore only the first-order polynomial is discussed.

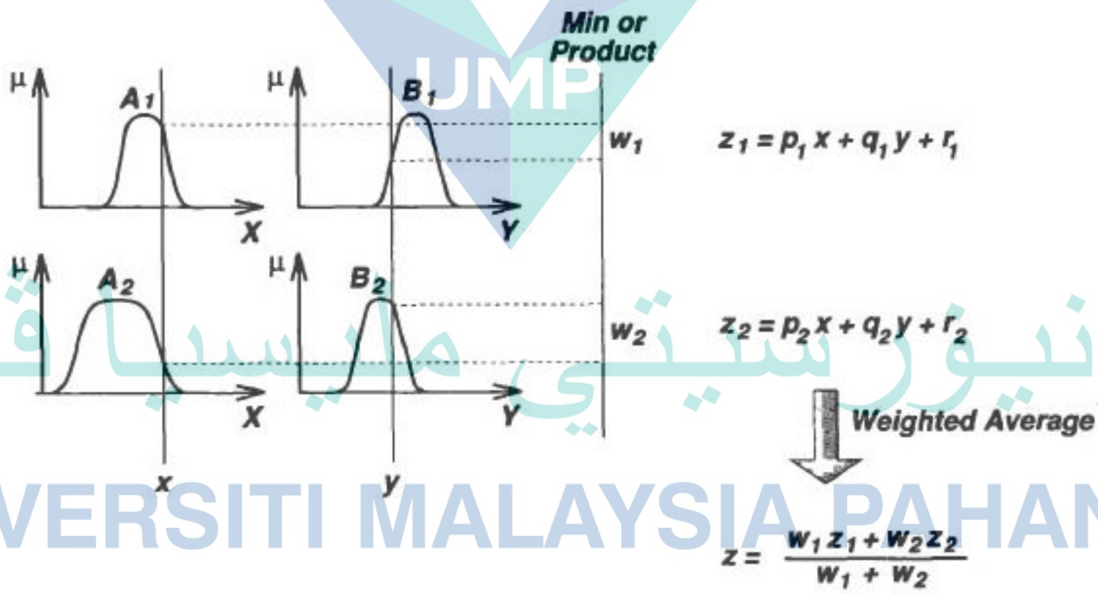

Figure 2.3 First order of Takagi-Sugeno method model
Source: Hamam and Georganas (2008).

Figure 2.3 illustrates the fuzzy reasoning steps in producing the first order of TS method. The overall output is gained by weighted average because each rule has a crisp output (Du and Zhang, 2008). Therefore, there is no defuzzification process in TS method. The weighted average can be replaced with the weighted sum depending on the user demands (Chen *et al*., 2007). However, the weighted sum could lead to the loss of membership function parameters.

The advantage of using TS method is more flexibility in the system design resulting in getting more parameters in the output (Blej and Azizi, 2016). Moreover, the processing time is better compared to Mamdani FIS because the weighted average deducts the defuzzification time. However, TS method can only be used for multiple input single output (MISO) system (Zaher *et al*., 2014).

Therefore for this project, Mamdani type is chosen because Mamdani type inference system gives the output based on the fuzzy set. While for Sugeno type, it is more suitable to be used if mathematical calculation is involved.

2.3.2 Functional Block of Fuzzy Inference System

Generally fuzzy logic inference system comprises of fuzzy sets, membership functions, linguistic variables, fuzzy rules and fuzzy reasoning (Nanda and Mangla, 2004). The important specifications of FLC are the fuzzy conceptual skill in perceiving the output of system dynamics and implementing these consistency concepts to control device simultaneously.

A basic block diagram of a fuzzy logic system that contains the position of input that goes through the fuzzification, rule base, decision and defuzzification for getting an output as shown in Figure 2.4.

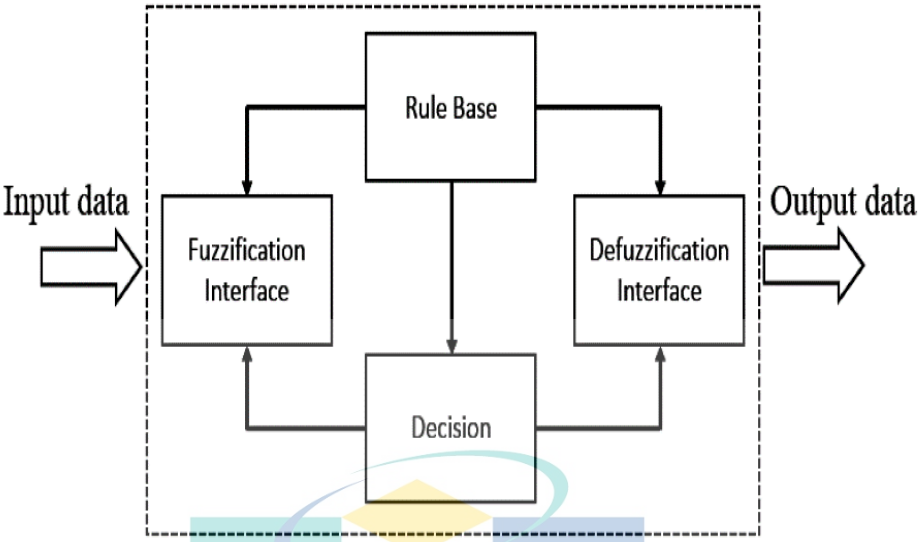

Figure 2.4 Structure of FLC
Source: Williams(2009).

2.3.2.1 Fuzzification

With each input fuzzy collection with the fuzzification method, data inputs or crisp measurements from certain measuring equipment are transformed into fuzzy values (Wang and Feng, 2013). The fuzzification system carries out the following functions (Khan and Engelbrecht, 2007).

i. Calculate the value of input variables.
ii. Carries out a scale mapping that shifts the range of values of input variables into the corresponding universes of discourse.
iii. Conducts the function of fuzzification that transforms input data to suitable linguistic values which can be interpreted as labels of fuzzy sets.

2.3.2.2 Membership Function

The fuzzy engine is the kernel of a fuzzy logic controller capable of modeling human decision-making based on fuzzy principles and inferring fuzzy control behavior utilizing fuzzy implication (fuzzy relation) and inference rules in fuzzy logic (Espin-Andrade *et al.*, 2016).This ensures further that fuzzy inference engine performs rule inferences as human knowledge being conveniently inserted by linguistic rules.

2.3.2.3 Rule Based (Knowledge Base)

The collection of rules is called a rule base. The rules are in the form of "If Then," and technically the If side is called the terms, and the conclusion is called the Then side (Angelov and Buswell 2002). Based on the calculated input error (e) and the change of error (ce) the machine will implement the rules and determine a control signal. The control technique is processed in a more or less natural language, in a rule-based system. For a non-specialist end-user, a rule base controller is simple to grasp and simple to manage and an analogous system may be introduced using traditional methods.

2.3.2.4 Defuzzification

Defuzzification happens as all triggered activities are merged and transformed into a continuous non-fuzzy output signal since that is the system's control signal. The performance values rely upon its rules and positions of the mechanisms, and on the non-linearity of the structures. To accomplish the goal, establish the control curve of the system reflecting the system's I/O relationship and depending from the details, determine the degree of output of the membership function in order to reduce the impact of the non-linearity (Khan and Engelbrecht, 2007).The output is output gain which can be tuned as well as becoming an integrator. The crisp output value may be determined by centre of gravity or by weighted average (Teimouri and Baseri, 2015).

The structure of fuzzy logic controller can be optimized for better performance in terms of faster response (Din *et al*., 2016). Since optimization method especially by swarm intelligence algorithm can improved the performance of a fuzzy logic controller by tuning the parameterized structure, a new optimization method for fuzzy logic controller design is proposed.

2.4 DC Motor

The electrical motors can be divided into two main types which are the direct current (DC) and also the alternating current (AC) motors. The electrical current that flowed will be based on the reference of either DC or AC (Krishnan, 2001). Both AC and DC motors serve the same functions but should be used depending on the user demands since the control will be totally different (Polat *et al.*, 2011). DC motor can be divided into two common types which are brushes or brushless (synchronous motor). AC motor also has two different types which are single phase and three phases (Lee and Kim, 2007). For this research, only the design of DC motor transfer function is covered and the next subtopic will discuss the details on types of DC motor and the method for producing transfer function of DC motor.

2.4.1 Type of DC Motor

DC motor such as brushed motor, brushless motor and servo motor are generally used in the industry for various purposes.

2.4.1.1 Brushed Motor

Basically in a wound rotor (the component which spins), a brushed motor generates a magnetic field by transmitting an electrical current through a commutator and a carbon brush assembly (Yedamale, 2003). The magnetic field of the stators (the stationary part) is generated either by winding the wound stator field, or by permanent magnets. Commonly brushed DC motors are inexpensive, small and simple to control.

A typical brushed DC motor generally consists of two parts, the motor's stationary body named the Stator and the internal component that rotates to generate the movement called the Rotor or "Armature" for DC machines (Afjei *et al.*, 2007). The motor wound stator is an electromagnetic circuit consisting of electrical coils linked in a circular structure for generate the appropriate North Pole then a South Pole and a North Pole style magnetic field network for rotation, unlike AC machines whose stator field rotates continuously with the frequency applied (Vazquez et al., 2011).

The current flowing inside such field coils is regarded as the current of the motor sector. These electromagnetic coils that generate the stator field are being linked electrically with the motor armature in series, parallel or both combined (compound) (Hernández-Guzmán *et al.*, 2015). A wound DC motor of the series does have stator field windings attached to the armature in series. Similarly, as seen in Figure 2.5, a shunt wound DC motor has its stator field windings attached in parallel to the armature.

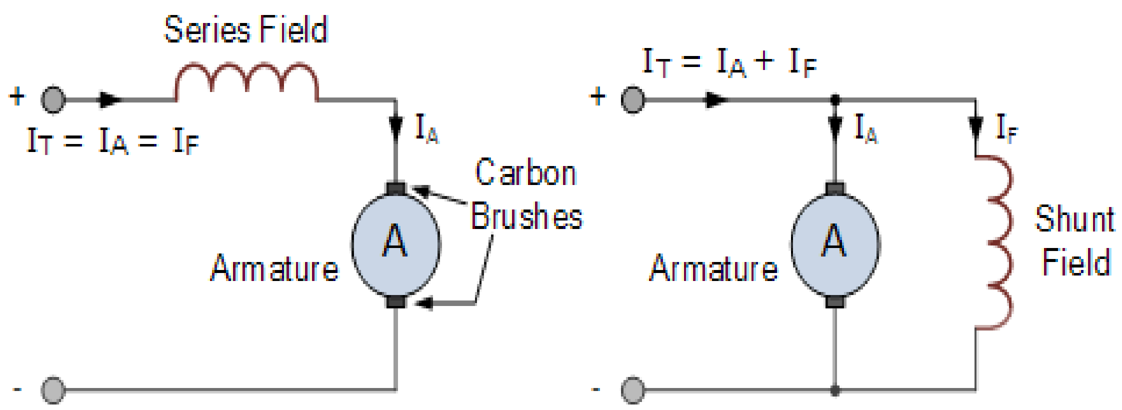

Figure 2.5 Series and shunt connected DC motor
Source: Williams(2009).

A DC machine's rotor or armature comprises of current bearing conductors attached to electrically separated copper parts, named the commutator, with one point (Xue*et al*., 2008). The connector makes an electrical link to an external power supply through carbon brushes (herewith the term "Brushed" motor) as the armature rotates (Scott*et al*., 2008). The rotor's magnetic field configuration attempts for coordinate well with stationary stator field that causes the rotor to rotate on its axis, yet might not coordinate itself attributable of delays in switching (Li*et al*., 2013). The motor's rotational speed depends on the power of the magnetic field of the rotors, and the more voltage added to the motor the quicker the rotor can spin (Xue *et al*., 2008). The rotational speed of the motor will also be adjusted by modifying the DC voltage.

2.4.1.2 Brushless Motor

This type of motor generates a magnetic field in the rotor through the use of permanent magnets connected with each one as well as electronic commutation is obtained (Krishnan, 2001). They are usually smaller but more powerful than traditional brushed style DC motors as they utilize "Hall effect" switches in the stator to generate the necessary stator field rotational sequence but have improved torque / speed characteristics, are far more effective and have a longer operational life than comparable brushed types (Song and Choy, 2004). A revolving permanent magnet in the rotor and stationary magnetic magnets in the motor housing are utilized for brushless DC motors (Nam et al, 2006). Brushless motor is a system required for transforming DC to AC

(Ozturk and Toliyat, 2010). This concept is easier from brushed drive, as it avoids the problem of power transfer from elsewhere in the drive to the rotating rotor (Yedamale, 2003). Compared to the brushed engine mentioned, this sort of motor requires less maintenance and more performance.

2.4.1.3 Servo Motor

This sort of motor is essentially a brushed DC motor with some kind of rotor shaft-connected positional feedback control (Bindu and Namboothiripad, 2012). They are attached and operated by a PWM form controller and are primarily required for control systems and devices operated by radio (Liu *et al*., 2009).

DC Servo motors have been utilized in closed loop based systems where the output motor shaft direction is passed directly to the motor control circuit. Standard "feedback" positional instruments include solvers, encoders and potentiometers required for remote control models including aircraft and boats (Akbar, 2014).

A servo motor typically has a built-in gearbox to minimize speed and seems to be able to specifically producing high torques (Shanmugasundram *et al*., 2012). Owing to the gearbox and feedback devices connected the output shaft of a servo motor will not spin loosely as do the shafts of DC motors.

A servo motor mainly comprises of a dc motor, a reduction gearbox, a positional feedback unit and some type of error correction (Bhushan and Singh, 2011). The speed or position is regulated with respect to a positional feedback or reference signal added to the system as seen in Figure 2.6.

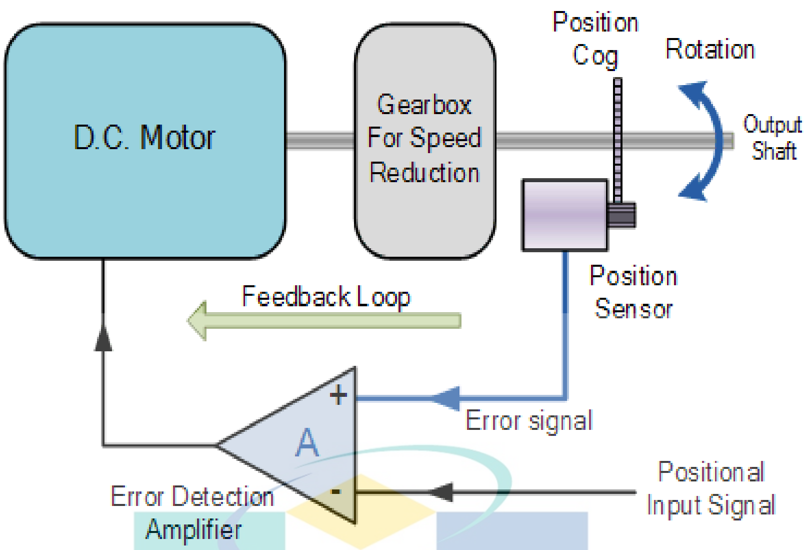

Figure 2.6 DC servo motor block diagram
Source: Brushan and Singh (2011).

The error detecting amplifier focuses on that input signal and relates it via the feedback signal from the output shaft of the motors and decides if the output shaft of the motor is in an error state and, if so, the controller then speeds up or slows down the motor (Maraiya and Ray, 2018). This response to the positional feedback device implies the servo motor is working inside a "Closed Loop Network" In addition to wide engineering applications, servo motors are often found in small remote control models and robots, with most servo motors are capable of moving in both directions up to around 180 degrees, allowing it to ideally suited for precise angular positioning (Chen and Sheu, 2002).

Nevertheless, when specifically adjusted, these RC model servos are reluctant to spin continuously at high speeds as typical DC motors do (Park et al., 2003). A servo motor comprises of many tools for regulating position, orientation or speed in one kit, including the engine, gearbox, and feedback system and error correction (Park et al., 2003). They are commonly utilized in numerous robotics and small prototypes, because they are conveniently operated by three wires, Power, Ground and Signal Control (Qureshi, *et al.*, 2016). In order to control the motor, controller is used for changing the motion direction by the Signal control. A controller is also known in making sure that any type of DC motors have precise angular position and have a quick response. Based on the characteristic of controllers, fuzzy logic controller is chosen to be used for monitoring the angular position. The reason for the selection of the controller is briefly explained in the fuzzy logic controller section.

2.4.2 Method of Modelling of DC Servo Motor

There are several methods that can be used to design the transfer function of DC motor.

2.4.2.1 Torque

The torque produced by a DC motor is usually proportional to the current of the armature and the power of the magnetic field (Ozturk and Toliyat, 2010). For this case the magnetic field is presumed to be constant and thus that the torque of the motor is proportional only to the armature current i by a constant factor K_T as shown in the equation below. This is called a motor operated by the armature.

$$T = K_T i \tag{2.1}$$

The back emf, e, is proportional to the angular velocity of the shaft by a constant factor, K_e.

$$e = K_e \dot{\theta} \tag{2.2}$$

In SI units, the motor torque and back emf constants are equal, that is $K_T = K_e$; therefore, K is used to represent both the motor torque constant and the back emf constant.

From the Figure 2.7, the following equations can be derived.

$$J\ddot{\theta} + b\dot{\theta} = Ki \tag{2.3}$$

$$L\frac{di}{dt} + Ri = V - \mathrm{K}\dot{\theta} \tag{2.4}$$

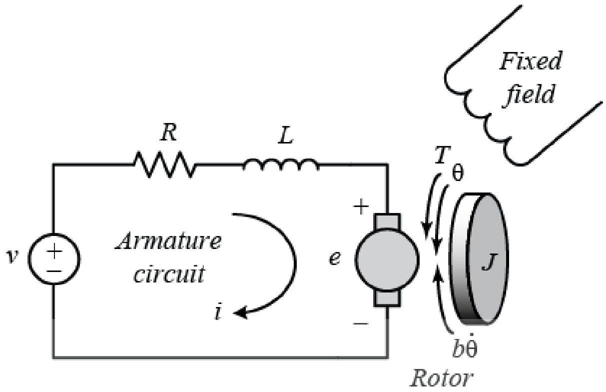

Figure 2.7 DC motor torque

Source: Ozturk and Toliyat (2010).

Where v = armature voltage (V)

R = armature resistance (Ω)

L = armature inductance (H)

i= armature current (A)

e = back emf (V)

T = motor torque (Nm)

θ = angular position of rotor shaft (rad)

J = rotor inertia (kgm^2)

b = viscous friction coefficient (Nms/rad)

Applying the Laplace transform, the above modeling equations can be expressed in terms of the Laplace variable *s*.

$$s(Js + b)\theta(s) \ = \ Ki(s) \qquad 2.5$$

$$(Ls + R)i(s) = \ V(s) - Ks\,\theta(s) \qquad 2.6$$

i(s) can be eliminated from the two above equations, where the rotational speed is considered the output and the armature voltage is considered the input.

$$\frac{\dot{\theta}(s)}{V(s)} = \frac{K}{(Js + b)(Ls + R) + \ K^2} \qquad 2.7$$

2.4.2.2 Lagrange function

The DC motor’s dynamic function is calculated using second-type equation from Lagrange. Initially, the device's kinetic energy T and potential energy, U, are measured and the dissipative function of the Rayleigh is incorporated into the equation of Lagrange that compensate for the damping and resistive forces inside the electromechanical framework (Song and Choy, 2004). For explaining the physical activity of the electromechanical DC motor, pick charging charge,q_1 and the angle of the rotor, q_2 as the generalized coordinates, such that q_1=q_1 ,q_2=θ and the first derivative as generalized current and speed as $q_1 = i_a$(t),q_2= ω(t). Thus, the Lagrange function, L is defined as the difference of the system’s kinetic and potential energy.

$$L = T - U \tag{2.8}$$

By denoting T_0 , U_0 and, T_1, U_1, the kinetic and potential energy in the electrical part and mechanical part respectively, the equation below can be obtained

$$L = (T_0 + T_1) - (U_0 + U_1) \tag{2.9}$$

According to conservation of energy, the kinetic and potential energy for the electromechanical DC motor are as follows:

$$T = \left(\frac{1}{2} L_a q_1^2\right) + (\frac{1}{2} J\, q_2^2) \tag{2.10}$$

$$U = -(V_s - V_b)\, q_1 \tag{2.11}$$

Rayleigh dissipative function in the electromechanical DC motor is:

$$R = \left(\frac{1}{2} R_a q_1^2\right) + (\frac{1}{2} B_m q_2^2) \tag{2.12}$$

Combining above equation into Lagrangian gives:

$$L = \frac{1}{2} L_a q_1^2 + \frac{1}{2} J_m q_2^2 + (V_s - V_b)\, q_1 \tag{2.13}$$

2.5 Examples of Controller Optimized by Algorithm on DC Motor

There are several examples of controller such as fuzzy logic controller (FLC) and proportional-integral-derivative (PID) that has been optimized by the swarm intelligence algorithm on DC motor as shown in Table 2.1.

Table 2.1 The example of controller optimized by algorithm for varies applications

Researchers	Approaches	Outcome
Nasri *et al.*, (2007)	A particle swarm optimization (PSO) method for determining the optimal proportional-integral derivative (PID) controller parameters, for speed control of a linear brushless DC motor.	The maximum overshoot can be minimized but the PSO designed PID is still unsatisfactory in terms of the rise time and the settling time.
Thomas and Poongodi (2009)	Design a position controller of a DC motor by selection of a PID parameters using genetic algorithm.	The designed PID with GA has much faster response than response of the classical method but still needs starting point of the PID values from other classical method.
Ibrahim *et al.*, (2011)	Developing fuzzy modelling of knee joint with genetic optimization	The new approach of modelling using fuzzy can eliminate the complicated mathematical modelling process. Simulation result shows some benefits when using fuzzy logic and genetic algorithm. But basically the performance is only proven in simulation only and not in experiment.
Rahmani *et al.*, (2012)	Designing fuzzy logic controller optimized by particle swarm optimization for DC motor speed control.	The designed FLC-PSO speed controller manages considerably improved the dynamic behavior as opposed to PID.

Table 2.1 Continued

Researchers	Approaches	Outcome
Bachache and Wen (2013)	Design fuzzy logic controller by particle swarm optimization for wind turbine.	The findings of the simulation demonstrate that the Optimized Fuzzy Logic Control (OFLC) uses PSO to achieve improved variables of fuzzy sets and perform better dynamic performance relative to traditional FLC.
Manikandan and Arulmozhiyal (2014)	The fuzzy logic controller is developed for controlling the position of DC servo motor drive	DC motor location had been easily controlled and come back to desired value however the settling time is still very high.
Yadav (2015)	Control the position of DC Motor by using Fuzzy Logic Controller (FLC) with MATLAB application and comparing with conventional PID control.	While conventional control "PID" is simple to implement, its solution is not so good for nonlinear systems. When using controls with fuzzy logic, the modification is impressive, achieving good dynamic response from the system.

2.6 Summary

This chapter has elaborated concisely on the types of DC motor. The DC motor is divided into three different sections that are: brushed motor, brushless motor and servo motor. In the same times, the method to obtain the transfer function was discussed. The different method such Kirchhoff's law, torque and Lagrange function were explained in details. Besides that, the main structure of fuzzy logic controller which is fuzzy inference system was explored from the methods and functional blocks accordingly. Then, it is followed by a brief introduction from a swarm intelligence algorithm to a modified adaptive bats sonar algorithm. Finally, the literature review on several applications that used controller optimized by existing algorithm was also deliberated.

For the next chapter, the designing process of the fuzzy logic controller, the optimization process, and simulation specifications will be presented.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The chapter starts with presenting the transfer function of the DC servo motor, the designing of fuzzy logic controller (FLC), the optimization of FLC using the modified adaptive bats sonar algorithm (MABSA), the simulation for proposed design FLC using MABSA and the comparison between PSO-FLC and MABSA-FLC. Lastly, is the specification's detail for giving more information about the method use for the simulation process. Details about the method can be useful for referring back when error occurred during running the simulation or implementing the hardware.

3.2 Research Methodology

In order to achieve the research objectives, the flow of work is planned. The research methodology can be divided into 4 phases.

Phase 1: Design fuzzy logic controller (FLC). It involves the design of fuzzy logic controller (FLC). Fuzzy logic toolbox will be used in order to build up the FLC in MATLAB software. The input and output variables will be determined and inserted into the fuzzification part. In the toolbox, the type of fuzzy inference system can be easily chosen either Mamdani or Takagi-Sugeno type. Once the type of fuzzy inference has been chosen, the appropriate design will be automatically loaded for each type of fuzzy inference system.

Phase 2: Optimization by the modified adaptive bats sonar algorithm (MABSA). The MABSA is employed to tune the FLC parameter. From the 4 structures of FLC which are fuzzification, membership function, rule based and deffuzification, membership function is chosen to be optimized by the algorithm. Since the parameters of membership function is fixed for a given range, the best position for the parameters need to be determined. MABSA will be used to get the best position for the value of parameters.

Phase 3: Simulation of proposed fuzzy logic controller (FLC) optimized by the modified adaptive bats sonar algorithm (MABSA) and particle swarm optimization (PSO) on DC servo motor position control. MATLAB Simulink software is required for simulation and research purposes which construct the FLC in sequential with transfer function of a DC motor. Additionally, the embedded MATLAB feature can be used to enforce MABSA to find an optimal FLC. William's (2019) transfer function for motor position control device will be utilized for evaluating the output of the latest designed of fuzzy logic controller optimized by the algorithm. In all simulations, the step response is used as an indicator of the controller performance. The simulation results will be evaluated to prove the capability of the proposed FLC optimized by MABSA to achieve best accuracy while minimizing time response. If the proposed design performs as desired, the research will proceed to hardware implementation. However, if the simulation results are not convincing, the proposed design will be adjusted and tuned back by MABSA.

Phase 4: Verification of proposed fuzzy logic controller (FLC) optimized by the modified adaptive bats sonar algorithm (MABSA) and particle swarm optimization (PSO) on DC servo motor position control by comparing with PSO algorithm. To verify the performance of the proposed controller, the PSO algorithm is used to compare in terms of rise time, settling time and percentage of overshoot. Two types of response will be taken into account which are response to sudden load increment and decrement. Finally, the result will be analysed. The process flowchart is shown in Figure 3.1.

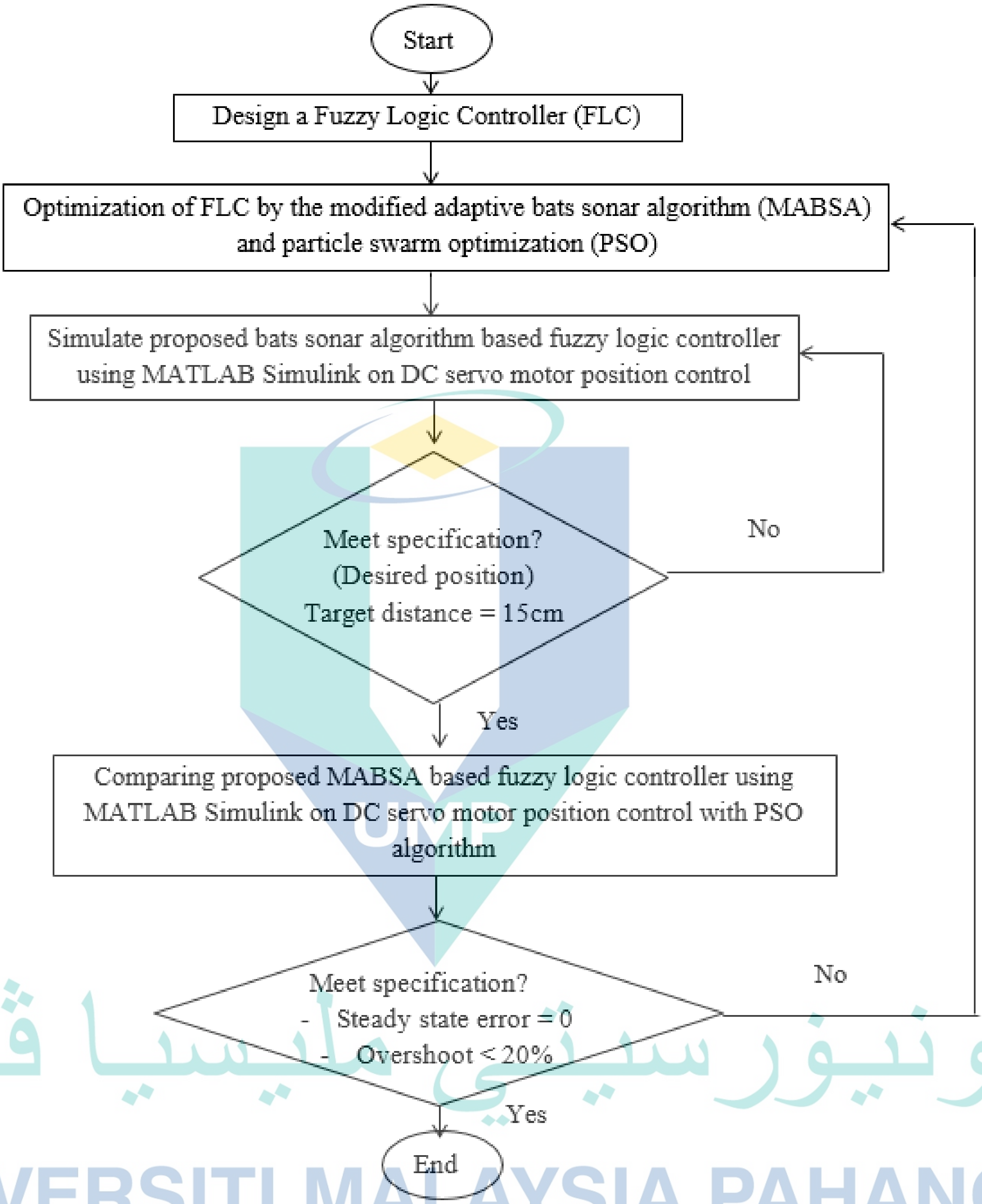

Figure 3.1 Flowchart of overall process

For the first specification that the target need to reach the 15cm in distance, the distance represent the maximum turning of the DC servo motor. However, for comparing between proposed optimization of MABSA and PSO, the distance will be reduced to 0.5cm since simulation take longer time to run. Therefore, the target specification of

15cm is only done with the optimization by MABSA for knowing whether the proposed design of MABSA-FLC can function properly. For the second specification, steady state error required to be zero since (Dai and Lin, 2015) (Komurcugil *et al.*, 2017) stated that a good control system will be the one that has a low steady-state error. And the overshoot need to be less than 20%. The reason is because researchers discovered that overshoot must below 20% from the final value to ensure that the signal receive will not be distortion (Ang *et al.*, 2005) (Morgan *et al.*, 2013).

PSO was inspired by having a population of candidate solutions which move around in the search space using a few pre-defined rules. The movements of these candidates are guided by their best position and the entire swarm's best known position. The movement of the swarm is then guided by improved positions. This process is a continuous one, until a reasonable solution is discovered.

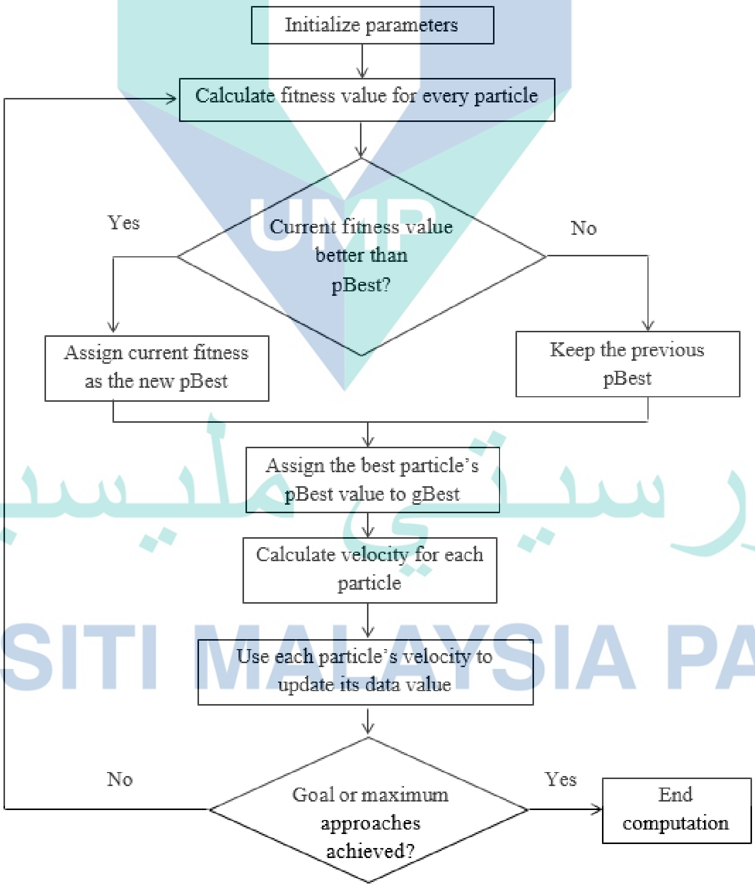

Figure 3.2 Flowchart of PSO implementation

The flowchart in Figure 3.2 shows the implementation of the PSO algorithm. This flowchart was then coded and used in the simulation. The parameters were initialized and the population that is generated works towards optimizing the objective function. Figure 3.3 illustrates the implementation of the MABSA. This flowchart was then adapted into coding for the simulation purposes. The parameters are initialized at the beginning of the algorithm. In the case of this research, it will work towards optimizing the controller parameters.

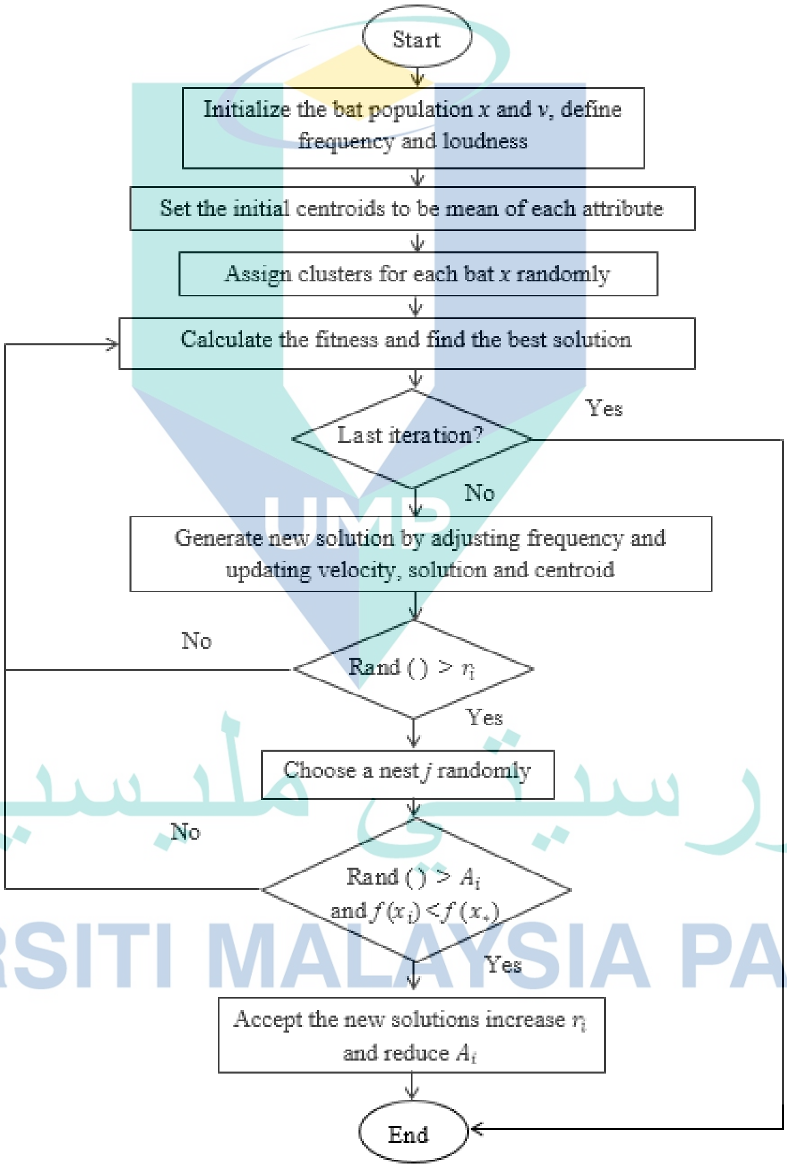

Figure 3.3 Flowchart of MABSA implementation

3.3 DC Servo Motor Modelling

First step for DC servo motor modelling is by mathematical calculation. The voltage supplied to the motor's armature is regulated in the armature control of independently excited DC servo motors, without varying the voltage added to the ground. Figure 3.4 presents an identical model of a separately excited DC servo motor.

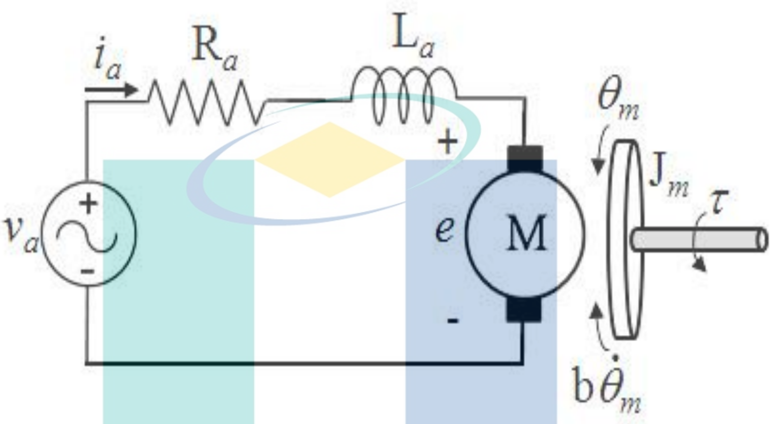

Figure 3.4 DC servo motor model

Source: William (2019).

$$v_a(t) = R_a . i_a(t) + L_a . \frac{di_a\ (t)}{dt} + e_b(t) \quad 3.1$$

$$e_b(t) = K_b . w\ (t) \quad 3.2$$

$$T_m(t) = K_T . i_a(t) \quad 3.3$$

$$T_m(t) = J_m . \frac{dw\ (t)}{dt} + B_m . w\ (t) \quad 3.4$$

Where v_a = armature voltage (V)

R_a = armature resistance (Ω)

L_a = armature inductance (H)

I_a = armature current (A)

E_b = back emf (V)

w = angular speed (rad/s)

T_m = motor torque (Nm)

θ = angular position of rotor shaft (rad)

J_m= rotor inertia (kgm^2)

B_m= viscous friction coefficient (Nms/rad)

K_T= torque constant (Nm/A)

K_b= back emf constant (Vs/rad)

Combining the upper equations together:

$$v_a(t) = R_a . i_a(t) + L_a . \frac{di_a\,(t)}{dt} + K_b . w\,(t) \qquad 3.5$$

$$K_T . i_a(t) = J_m . \frac{dw\,(t)}{dt} + B_m . w\,(t) \qquad 3.6$$

The relation between rotor shaft speed and applied armature voltage is represented by transfer function:

$$\frac{W(s)}{V_a(s)} = \frac{K_T}{L_a . J_m . s^2 + (R_a . J_m + L_a . B_m) . s + (R_a . B_m + K_b . K_T)} \qquad 3.7$$

The relation between position and speed is:

$$\theta\,(s) = \frac{1}{s} W(s) \qquad 3.8$$

Then the transfer function between shaft position and armature voltage at no load is:

$$\frac{\theta(s)}{V_a(s)} = \frac{K_T}{L_a . J_m . s^3 + (R_a . J_m + L_a . B_m) . s^2 + (K_T . K_b + R_a . B_m) . s} \qquad 3.9$$

The transfer function will be used in the Simulink motor model.

The parameters is determined by the type of DC motor used. Every motor type has different DC motor functions that need to be included for the simulation. Table 3.1 below shows the parameter values of the DC motor.

Table 3.1 The motor parameterization

Parameters	Values
Armature resistance (R_a)	4 Ohm
Armature inductance (L_a)	2.75e-6 H
Back emf constant (K_b)	0.0274 V/ (rad/s)
Rotor inertia (J_m)	3.2284e-6 kg*m^2
Moment coefficient (K_T)	1.28 Nm/A
Rotor damping (B_m)	3.5077e-6 N*m/(rad/s)

3.4 Fuzzy Logic Controller Modelling System

For the fuzzy logic controller (FLC) design, it can be divided into two parts. First one by designing the controller in MATLAB toolbox and then recalled it back to the Simulink block diagram.

3.4.1 MATLAB Fuzzy Logic Toolbox

The designed fuzzy logic controller (FLC) from MATLAB software will be using the MATLAB toolbox. A typical FLC toolbox as shown in Figure 3.5 consists of the basic elements, namely Fuzzy Inference System (FIS) Editor, the Membership Function Editor, the Rule Editor, the Rule Viewer, and the Surface Viewer.

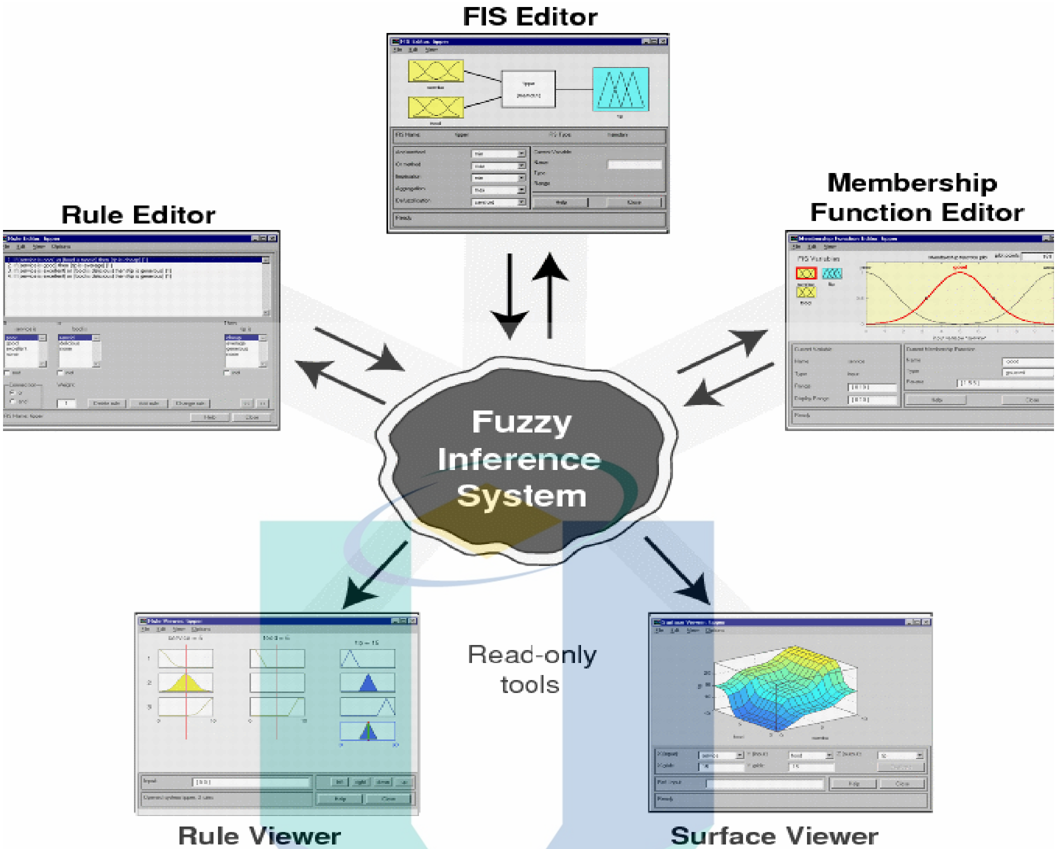

Figure 3.5 Fuzzy inference system

So, firstly add the variables for the input since two inputs will be used which are error and change of error. For the output will be the position of motor. The range for inputs and output is from -1 until 1. Equations 3.10 and 3.11 show the inputs for the FLC system.

$$e\,(t) = \frac{w_{ref} - w_{measured}}{w_{ref}} \qquad 3.10$$

$$\hat{e}\,(t) = \frac{\Delta\, e(t)}{\Delta t} = e(t) - e\,(t-1) \qquad 3.11$$

After deciding the input and output parameters, the value will be inserted in the FIS editor as shown in Figure 3.6. The reason to use FIS editor is because there is no limitation of the input numbers in the fuzzy logic toolbox software.

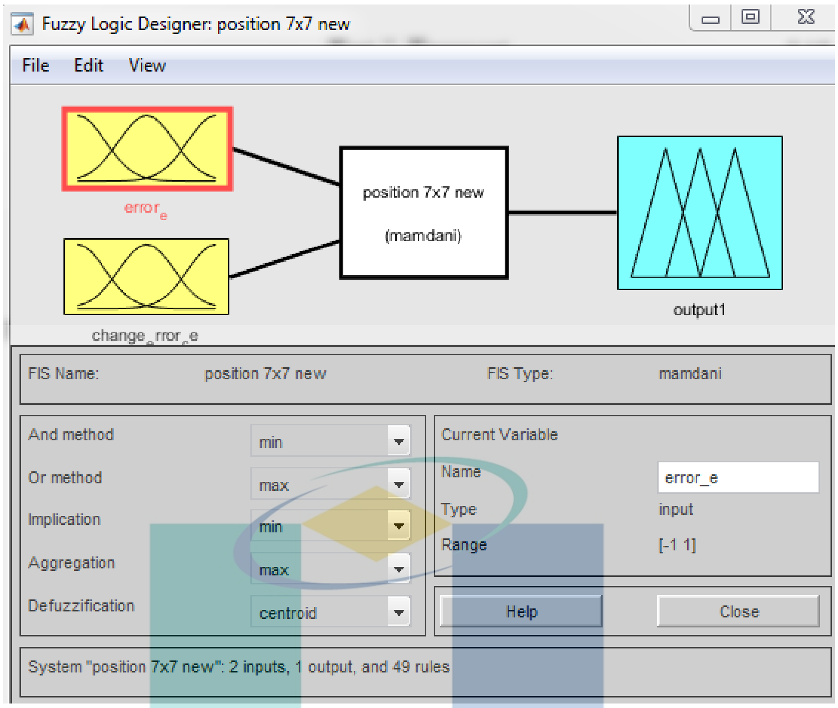

Figure 3.6 FIS editor for proposed design of fuzzy logic controller

The main function of membership function editor is for determining the formation of input and output variables. The membership function editor shares some features with the FIS editor. The membership function editor lets to arrange and rearrange all the membership functions that linked with all inputs and outputs variable. The 7x7 membership function is chosen for designing the FLC since the performance evaluated become more accurate and precision. Table 3.2 shows the membership function of designed controller.

Table 3.2 Rules table for fuzzy logic controller

Δe ce	NL	NM	NS	ZE	PS	PM	PL
NL	NL	NL	NL	NL	NM	NS	ZE
NM	NL	NL	NL	NM	NS	ZE	PS
NS	NL	NL	NM	NS	ZE	PS	PM
ZE	NL	NM	NS	ZE	PS	PM	PL
PS	NM	NS	ZE	PS	PM	PL	PL
PM	NS	ZE	PS	PM	PL	PL	PL
PL	ZE	PS	PM	PL	PL	PL	PL

From the membership function, the rule for the behaviour of the system will be identified. The fuzzy inference diagram can be viewed by using the rule viewer. Rule viewer displays a roadmap of the whole fuzzy inference process. By using this viewer, the individual form of membership function can be seen affecting the result. And for viewing the dependency of one of the output on any one or two of the inputs is by surface viewer. Surface viewer will generate and plots an output surface map of the system. After the rules are applied to both inputs and output, surface viewer will be obtained. Figure 3.7 shows the surface viewer that has been generated based on the parameters inserted.

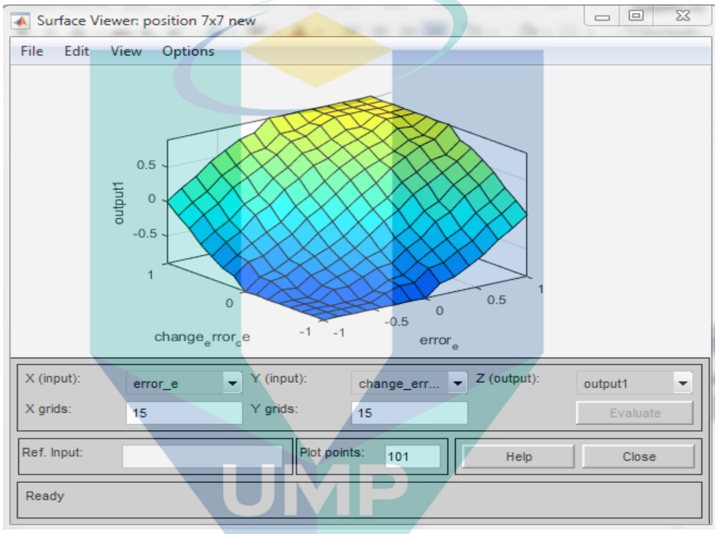

Figure 3.7 Surface viewer for designed fuzzy logic controller

After the surface viewer is generated, the block diagram for the overall process will be designed. The transfer function used is modelled and the FLC is called back in the block diagram that has been provided in the Simulink.

3.4.2 MATLAB Simulink

Following the complete design of the fuzzy logic controller (FLC) and also the findings produced by Surface Viewer, the MATLAB Simulink model is then built to simulate and test the system performance for the fuzzy logic controller. A transfer function will be linked to FLC output. This transfer function generally refers to the motor and the input to the FLC is the step input. The scope has been utilized for viewing the controller performance during the overall process. The resulted response will be in step response.

The block diagram for the system is shown in Figure 3.8. The block diagram is designed based on the closed loop system. There are three different subsystems in the block diagram that represent the system without the fuzzy logic controller, with the fuzzy logic controller, and also the system that has been optimized by MABSA.

The step block is placed in front of the design to provide a step for the system without FLC and also system with FLC. The Mux is needed since for FLC, two inputs will be entered into the fuzzification process. And at the last defuzzification, the output will go to the DC motor transfer function. The result from the scope will be compared by the different subsystem that has been designed. The output will be saved to the workspace in MATLAB for generating automatically all the needed parameters that need to be compared.

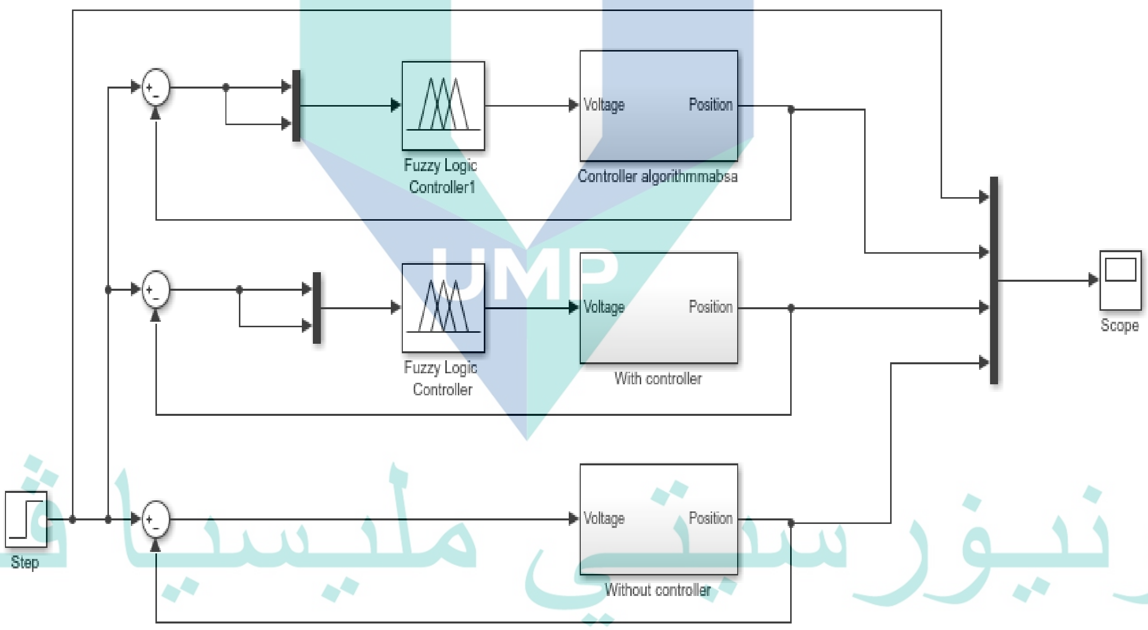

Figure 3.8 Block diagram for overall system

The comparison of all three different subsystems will be discussed to verify the assumption where the proposed FLC optimized by MABSA will have better step response performance compared to other subsystems. The comparison of performance will be validated in terms of rise time (T_r), settling time (T_s) and also the percentage of overshoot.

Figure 3.9 displays the transfer function of DC motor block diagram. The input will be the voltage, while the output for the DC block diagram is the position of the DC motor. The voltage is computed to the output position by setting the value of inductance, resistance and the final value is also integrated.

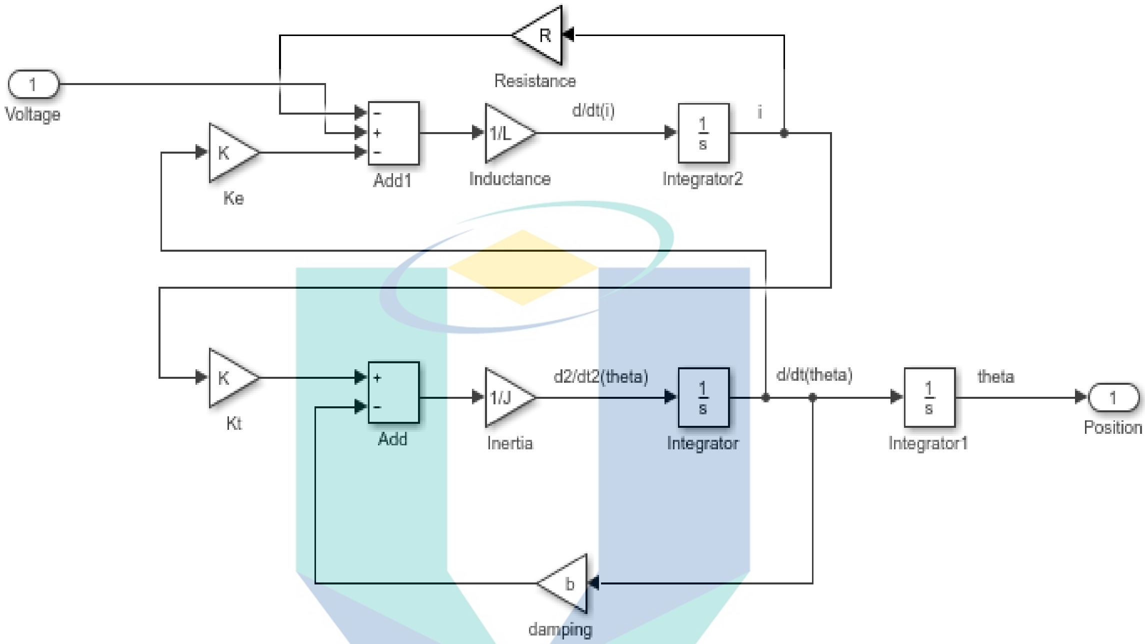

Figure 3.9 Block diagram for DC motor transfer function

3.5 Optimizing of Fuzzy Logic Controller by Modified Adaptive Bats Sonar Algorithm and Particle Swarm Optimization

A self-tuning method for fuzzy inference rules employing a descent method for Takagi-Sugeno fuzzy rules with constant outputs, and isosceles-triangular fuzzy numbers by Maeda and Murakami (1992) has been proposed. Besides that, the gradient descent approach for optimizing Takagi-Sugeno rules with symmetric and asymmetric triangular membership functions and output functions (Sugeno regular guidelines), suggesting "Takagi-Sugeno based rule" to prevent a particular class of local minima (Siarry and Guely, 1998).

Siarry and Guely (1998) suggested a computer-aided tuning method for gradient analysis, input variables with width around each side equivalent to the gap between the two adjacent peaks, and output variables utilizing symmetrical, triangular membership features of equal width.

A fuzzy logic controller's output relies upon its rules of control and membership functions. Thus, the application of these variables to the mechanism to be controlled is quite critical. There is still no FLC that have been optimized by bats echolocation-inspired algorithm. Therefore, an example to make the fuzzy logic control systems behave as closely as possible to the operator or expert behaviour in a control process is by tuning the fuzzy control rules.

The genetic algorithms are used for the tuning of fuzzy control rules. In (Herrera *et al*., 2005), training data (TRDs) is used for tuning fuzzy controllers. A set of TRDs is a pair of input-output data, wherein expected output values are the output data, and the input outcomes are applicable for the fuzzy input values. Such tuning data reflect the control behaviour of the skilled-operator. The tuning approach matches the membership functions of the fuzzy rules provided by the experts with the chosen inference system and defuzzification technique, achieving high-performance membership functions through reducing a defined error function using a collection of input-output measurement results.

For this project, the modified adaptive bats sonar algorithm (MABSA) and particle swarm optimization (PSO) will be employed to tune the FLC parameter. The approach employs MATLAB/M – file coding scheme in the Simulink/Embedded MATLAB Function block. Fuzzy inference system and scaling gains for the inputs and output signals are the parameters to be optimised. The fitness values for all particles are measured at each iteration dependent upon (Park *et al*., 2003).

Afterward, modifications are applied based on the updated values and next iteration starts. The running code will be run for 30 times. The best position is selected based on the set point that are nearer to zero.

3.6 Performance Validation

3.6.1 Comparing FLC-MABSA with FLC-PSO

As both PSO and MABSA algorithms are commonly ought to solve the DC motor issues, the performance of the algorithms should be tracked to understand precisely which is better in getting the system's step response performance, which used the fuzzy controller optimization in the DC motor position control. The FLC parameter will be optimized using a modified bats echolocation-inspired algorithm (MABSA) and particle swarm intelligence (PSO). The approach is to introduce a MATLAB / M-file coding scheme in the function block Simulink / Embedded MATLAB. The parameters that are significantly altered are the fuzzy inference system and the scaling gains for both inputs and output signals. The fitness values for all the particles are measured upon each repetition, based on the location of the particles. The modifications are then applied according to the updated values, and the next iteration begins.

The position function values are set, as the Fuzzy logic controller is just designed based on the minimum and maximum position of the boundary. Hence, the algorithm would be applied to optimize the controller for obtaining the best value to determine the position of the limit values. MATLAB Simulink software is implemented for evaluating and simulating the constructed FLC design when the transfer function of a DC motor is applied within the FLC. Additionally, the Embedded MATLAB function is implemented for ensuring the FLC can indeed be optimized for the algorithm. The simulations would be in step response form, in which the output of the controller may be shown. In order to validate the efficiency of the updated developments fuzzy logic controller designed by both algorithms a control model for motor position control system will be suggested.

To tune the FLC parameter, the modified bats echolocation-inspired algorithm (MABSA) and particle swarm optimization (PSO) are employed. The membership function is selected from the four structures of the Fuzzy Logic Controller (FLC), to be optimized by both algorithms. In both algorithms the m-file coding is changed depending on the transfer function. Figure 3.10 demonstrates the PSO algorithm for m-file application and Figure 3.11 demonstrates the MABSA algorithm. To get the best position, the four best values are chosen and inserted in the membership value function.

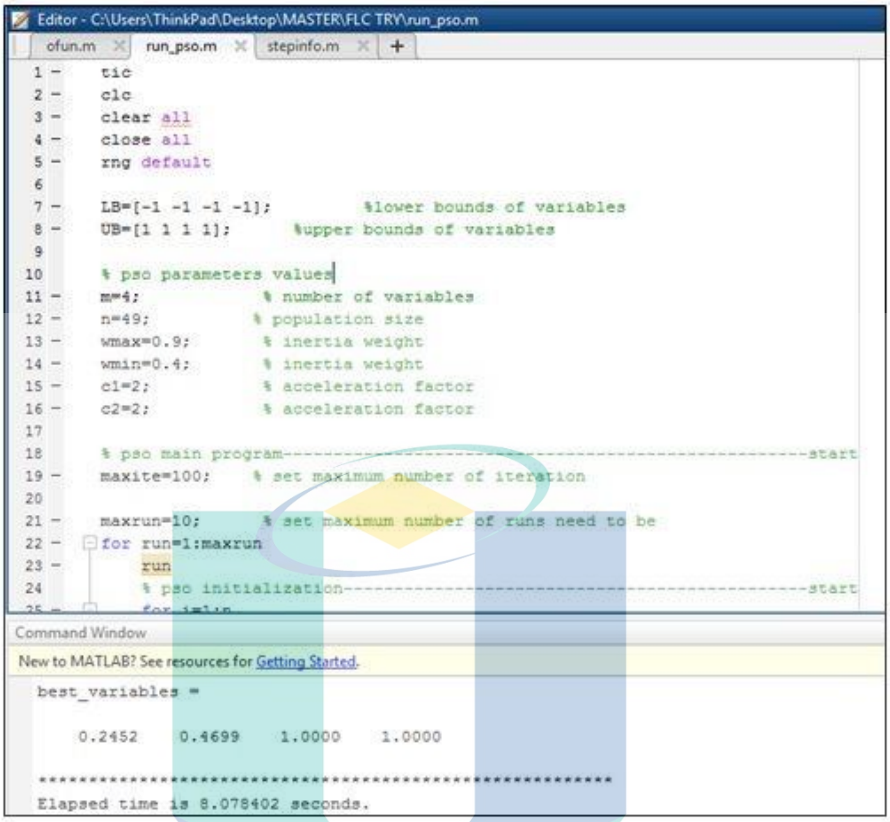

Figure 3.10 M-file coding for PSO algorithm

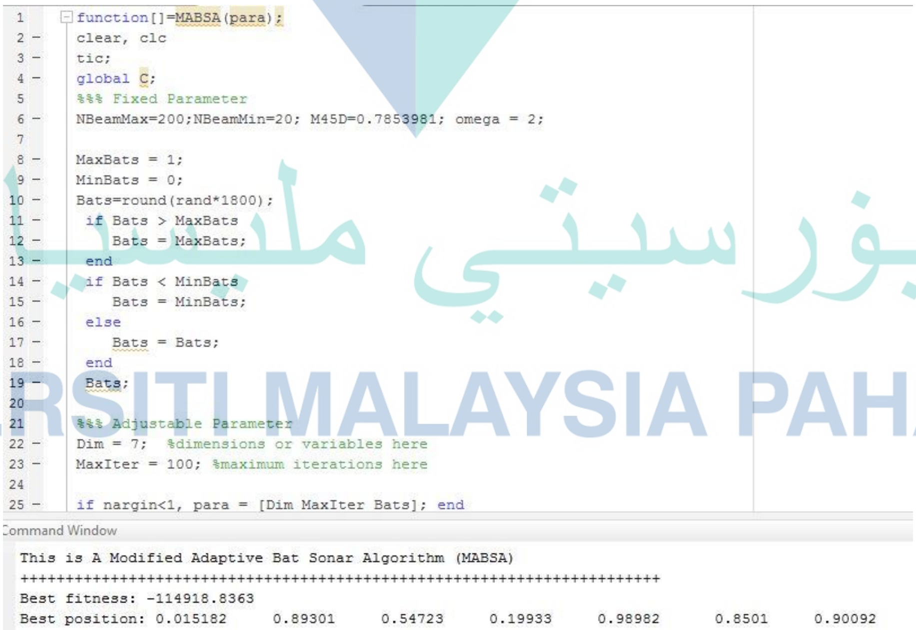

Figure 3.11 M-file coding for MABSA algorithm

After defining the four values, the values would then be incorporated into the membership function of the developed fuzzy logic controller. Each membership function has 3 parameters. The optimized values are placed in the membership function and then applied to obtain the optimized values. The following Figure 3.12 indicates the phase through which the value is inserted into the controller. The same measures for MABSA algorithm will proceed as shown in Figure 3.13.

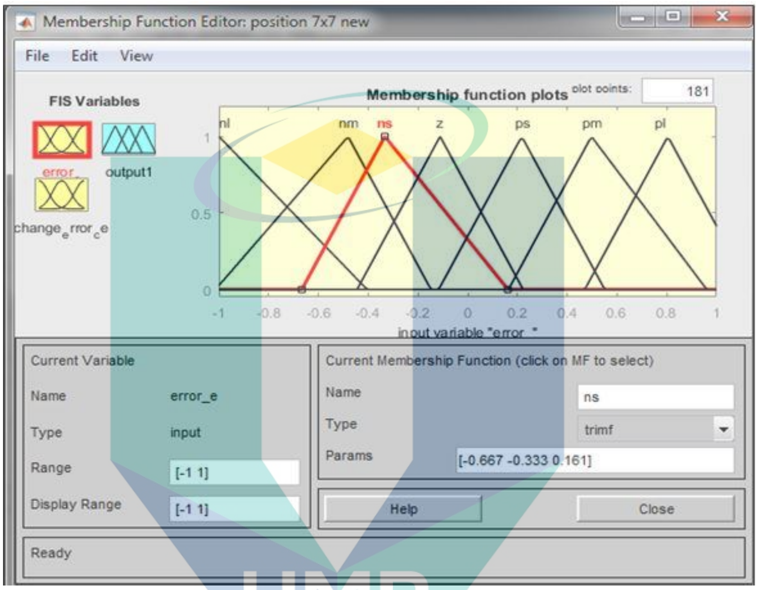

Figure 3.12 Membership function for optimization using PSO algorithm

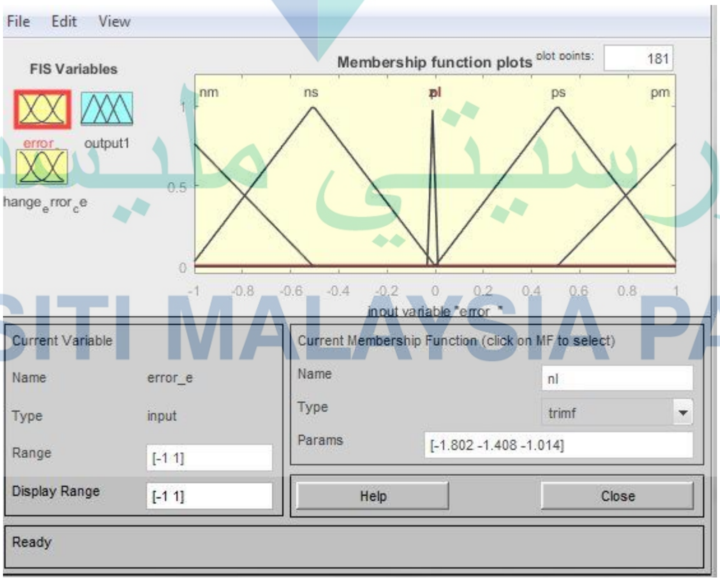

Figure 3.13 Membership function for optimization using MABSA algorithm

The new Surface Viewer will be created after all the procedure is followed. Finally, the fuzzy logic controller built for both algorithms may be utilized for controller in the Simulink block diagram. The FIS file for designing controller that have been optimized by modified adaptive bat sonar algorithm as shown in Figure 3.14

Editor - C:\Users\ThinkPad\Desktop\MASTER\FLC TRY\position 7x7 new.fis

MABSAtrysep.m | ltirepresentation.m | september15tryposmabsa.fis | position 7x7 new.fis

```
[System]
Name='position 7x7 new'
Type='mamdani'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=49
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='error_e'
Range=[-1 1]
NumMFs=7
MF1='nl':'trimf',[-1.333 -1 -0.6666]
MF2='nm':'trimf',[-1 -0.6666 -0.3334]
MF3='ns':'trimf',[-0.6666 -0.3334 0]
MF4='z':'trimf',[-0.3334 0 0.3334]
MF5='ps':'trimf',[0 0.3334 0.6666]
MF6='pm':'trimf',[0.3334 0.6666 1]
MF7='pl':'trimf',[0.6666 1 1.334]

[Input2]
Name='change_error_ce'
Range=[-1 1]
NumMFs=7
MF1='nl':'trimf',[-1.333 -1 -0.6666]
MF2='nm':'trimf',[-1 -0.6666 -0.3334]
MF3='ns':'trimf',[-0.6666 -0.3334 0]
```

Figure 3.14 The FIS file of fuzzy logic controlller

The input will be the error and change of error. The iteration runs for 30 times and the best fitness is chosen when the value is nearer to zero. Then 4 values for the best position will be chosen for creating the centre points of standardized, triangular membership functions to categorize the whole fuzzy partition of a variable. This property is preserved by a description which codes rather than its absolute position the distances between the adjacent fuzzy sets. After getting the new location of centre points, a new designing fuzzy logic controller will be built.

The output of the constructed fuzzy controller with the DC motor transfer function are being computed and evaluated in the MATLAB Simulink model, when the modelled fuzzy logic controller is completely generated as well as the Surface Viewer obtained the data. The rules contained in the fuzzy logic controller have been operating as a "brain"-controller throughout the whole program. The system's functionality as regards step response should be created via the Simulink model's output scope. MATLAB Simulink Library had Fuzzy logic controller blocks. The DC motor is the power output actuator used in the Simulink layout. Figure 3.15 shows the controller block diagram, DC motor and the overall system.

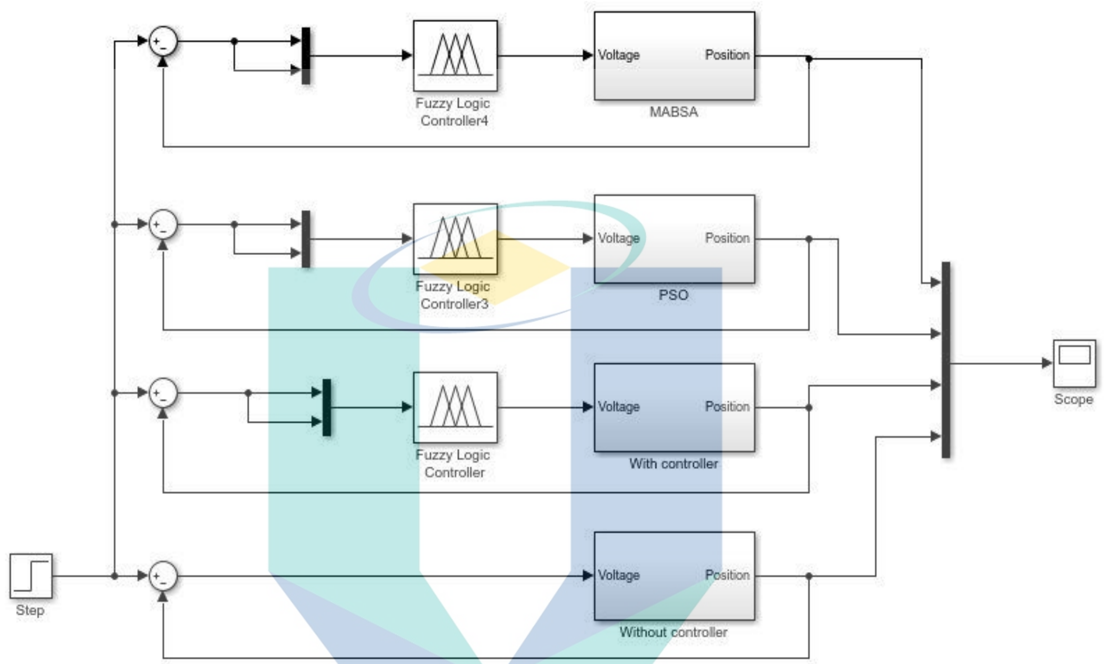

Figure 3.15 The block diagram for PSO-FLC and MABSA-FLC

For both algorithms, the output from scope is contrasted either with the controller or without the controller. The block diagram comprises four separate subsystems. Without the controller, the controller and even the mechanism which has been optimized by the PSO and MABSA algorithm reflect the DC motor mechanism.

And the fuzzy logic controller built for each circuit would also be different because the designed FLC is modified when the algorithm turned it on. The performance will be stored to MATLAB’s workspace for automatically generating the value for that rise time, settling time, and percentage of overshoot.

3.6.2 Robustness Test in Input Signal

Robust control is a branch of control theory which addresses uncertainty precisely in its solution with controller design. It also involves the processing of unknown plants subject to unknown disturbances with unknown dynamics. Disturbance signals describe undesired inputs that affect the output of the control system and cause system error to significantly rise. This is the control-system engineer's task to correctly construct the control device to partly remove the consequences of performance and device error disturbances. There are three disturbance groups which penetrate outside control loops: set point changes, load variations, and noise. Flow loops will mainly react to adjustments to the set point (input signal), load disturbances become especially normal when several users draw from a specific pump or compressor. For this research, set point changes is used to test the performance of the controller. There will be 2 sources that will be added which are sine-wave and pulse generator. Sine wave is usually used in order to get reliable results within the regions of the resonance frequencies. Figure 3.16 shows the block diagram of the overall system with sine-wave input signal.

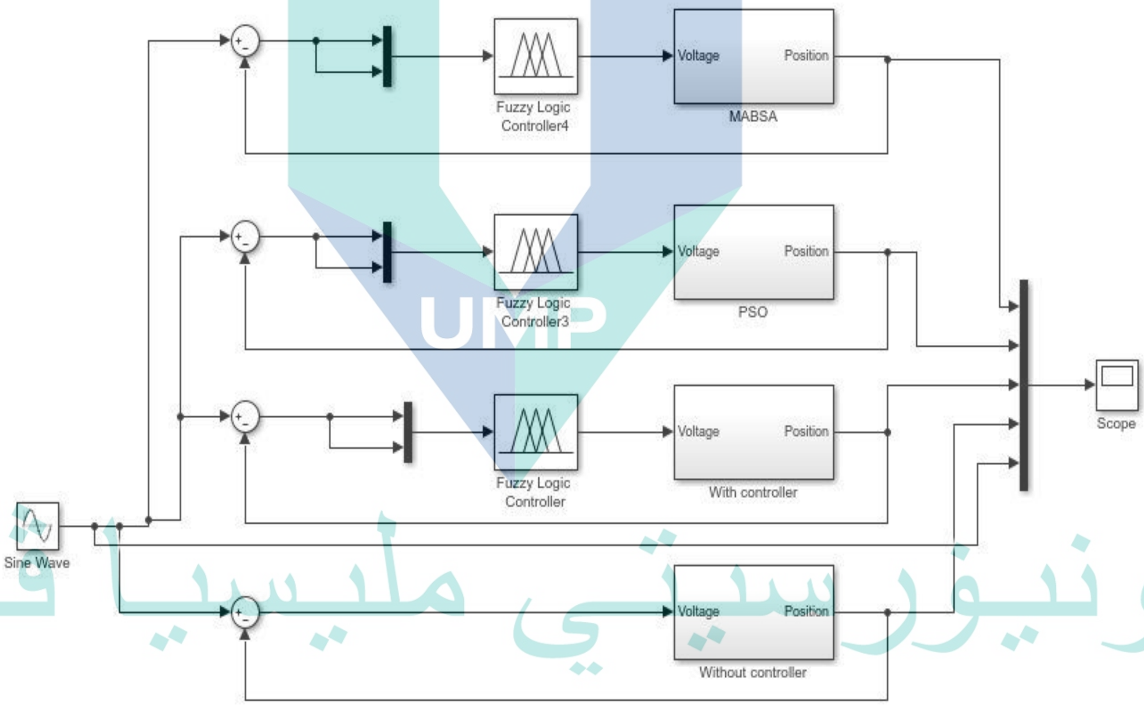

Figure 3.16 The block diagram for adding sine-waveas input signal

Either an electrical circuit or a piece of electrical research equipment used to generate rectangular pulses is a pulse generator. Pulse generators are mostly used to deal for digital circuits, the associated feature generators are predominantly used for analog circuits. Figure 3.17 shows the block diagram of the overall system with pulse generator input signal.

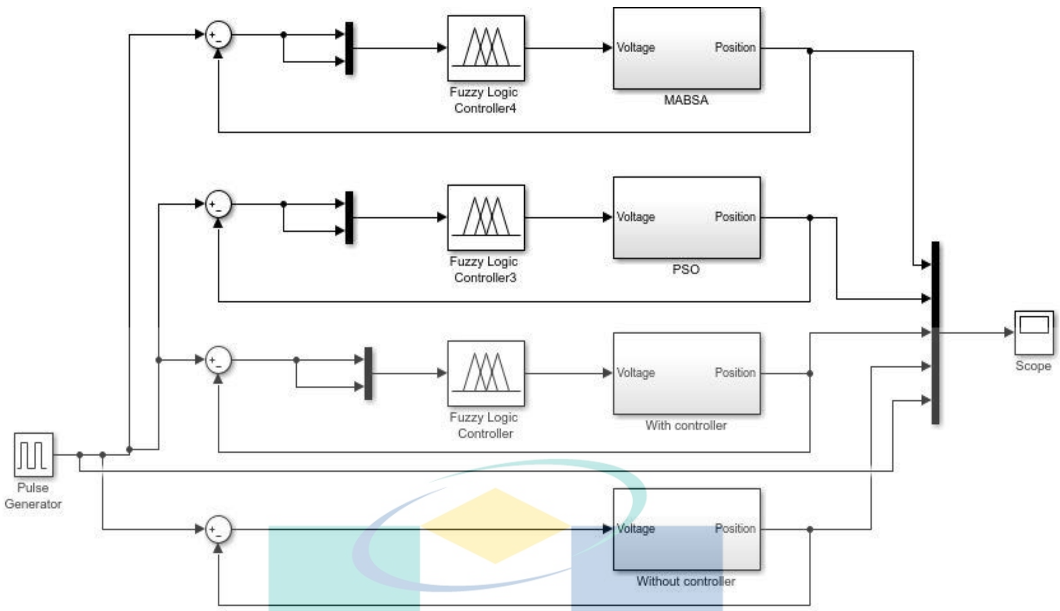

Figure 3.17 The block diagram for adding pulse generator as input signal

3.7 Summary

This chapter elaborated concisely on the procedure of designing FLC optimized by MABSA for DC motor position control. This chapter began with the DC servo motor modelling where the transfer function will be developed and inserted into the block diagram in Simulink model. Then, the chapter followed with designed fuzzy logic controller and continued by optimizing the proposed FLC design with MABSA. After that, the simulation and also the experimental of the proposed FLC optimized by MABSA are carried out.

In the next chapter, the verification of the results for simulation will be done. The results will then be compared to PSO algorithm to analyse the performance.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

Following the complete design of the fuzzy logic controller (FLC), and Surface Viewer acquired from the outcome, the MATLAB Simulink model is produced for the FLC to simulate and test the system performance. The rules that are placed in the FLC function like a "brain" – controller throughout the entire process. System efficiency is simulated in Simulink's output Scope model. MATLAB Simulink Library has sets of FLC that can be used directly. DC servo motor has the main actuator used in the Simulink device. Inside the Simulink model, the transfer feature of the DC servo motor is used to represent the output actuators. The scope of the outcome will be measured dependent on DC servo transfer function that used only the FLC, FLC that has been optimized by the modified adaptive bats sonar algorithm (MABSA) and also system without using controller for reference point.

Comparison between MABSA-FLC and PSO-FLC is done for the verification of the simulation performance. After the computer simulation is fully done, the result will be compared based on the transfer function of DC servo motor that used only the FLC, FLC that has been optimized by MABSA and also system without using controller for reference point. The comparative result for the computer simulation will also be going through robustness test for testing the controller's performance.

4.2 Overall Result

The result is divided into three categories which are the simulation result, the comparative result between PSO and MABSA and also the robustness test. For the simulation result, a Simulink model that reflects the system behaviour which includes the transfer function of DC motor is built to test the proposed FLC that has been optimized by MABSA. In order to validate above optimization method, comparison between algorithms of controller is introduced for controlling the direction of the motor and also robustness test for validating the FLC performance.

4.2.1 Result for Computer Simulation

The findings from the performance of the Simulink model which used a standard form of DC motor are shown in Figure 4.1. The red straight line from the graph reflects the input signal that is used as an indicator for benchmarking to validate other subsystem performances. The black straight line represents the fuzzy logic controller optimized by MABSA. The line describing the system that utilized the controller is quicker in stationary condition relative to the smoother and thinner dotted line reflecting the system without the controller being employed. The system that uses the controller has an overshoot, therefore the rise time of the response is quicker than without the controller as opposed to the system.

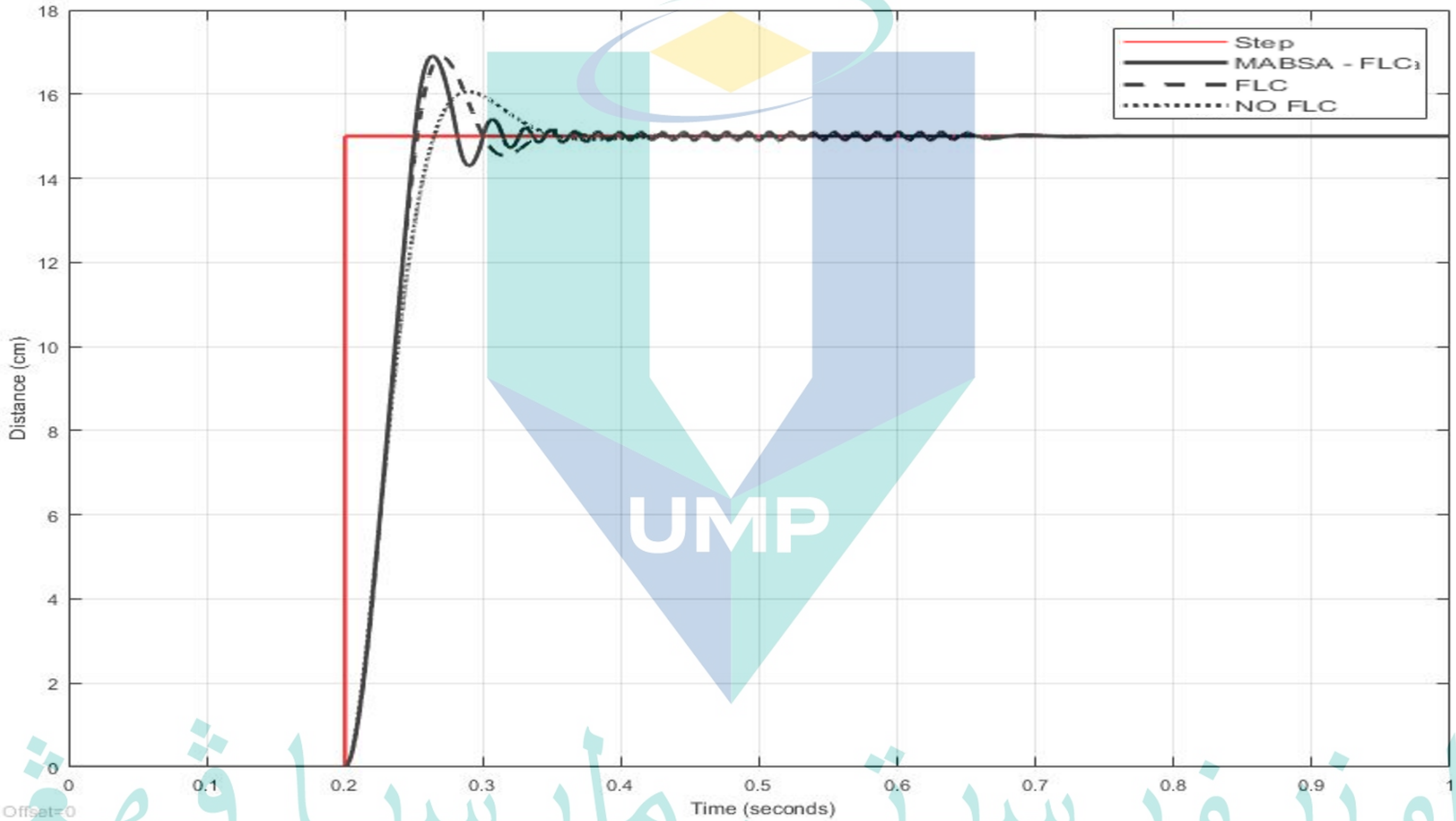

Figure 4.1 The simulation result for system without the fuzzy controller, with the fuzzy controller and with the fuzzy controller optimized by MABSA

The outcome which used the controller's MABSA provides greater outcome as opposed to the transfer function which used software only and without the controller. It is presented in the simulation findings as seen in Table 4.1. The important parameters' values such as rise time, settling time and also percentage overshoot are tabulated.

Table 4.1 The simulation result for step response graph

Characteristics	Values		
	NO FLC	FLC	With MABSA-FLC optimized controller
Rise Time (s)	0.29	0.26	0.25
Settling Time (s)	0.40	0.38	0.35
Overshoot (%)	10.0	13.3	13.3
Steady state error	0	0	0

Based on the simulation results, MABSA-FLC shows that the subsystems have faster response in terms of rise time and settling time compared to the system without a FLC controller and system with FLC controller only. The reason for this behaviour is because the function of controller is adding an overshoot thus increasing the response time. Therefore, it is expected that system with FLC controller will give faster increment in the rise time. The rise time for MABSA-FLC is 0.01seconds faster compared to system with FLC only and 0.04seconds faster compared to the system without FLC. For the optimized algorithm, MABSA performs by 3.8% improvement in terms of rising time and also 7.5% better in settling time. In terms of settling time, MABSA-FLC takes only 0.35seconds to return to the final values. When without FLC controller is used as the reference point, the settling time for MABSA-FLC yields improvement of 12.5% in contrast to the FLC controller only which is 7.5%. The system without a controller, though, provides better performance than the system with a controller in the aspects of producing a lower overshoot. Observational studies, the system that has controller managed to shorten the rise time, raised the overshoot and reduced the settling time by a slight amount. The same goes to the system which is optimized by MABSA algorithm. The fuzzy logic controller optimized by MABSA provides greater contrast of performance than using the regular fuzzy controller without an optimization.

4.2.2 Result for Comparison of MABSA-FLC and PSO-FLC

The outcomes from the Simulink model's step response which was using a standard DC motor type are shown in Figure 4.2. Each subsystem is represented by 4 cases. The first one is the subsystem which did not use a fuzzy logic controller. The second one is the subsystem that used the fuzzy logic controller only without the optimization of algorithm. And then the last two are the system optimized by MABSA and PSO algorithm in the fuzzy logic controller.

A 0.05s delay is inserted because there must be some delay in turning on the DC motor, before it gets started. Form the graph, clearly MABSA-FLC becomes the fastest to reach the final value compared to other cases. Just like the result in simulation above, system without controller has provided slower step response in rise time and also settling time.

And the optimization by algorithm for controller gives better output performance compared to system that used controller only. Comparing between MABSA-FLC and PSO-FLC, MABSA-FLC is faster in stationary state compared to the PSO-FLC. The detail outcome is tabulated in the table.

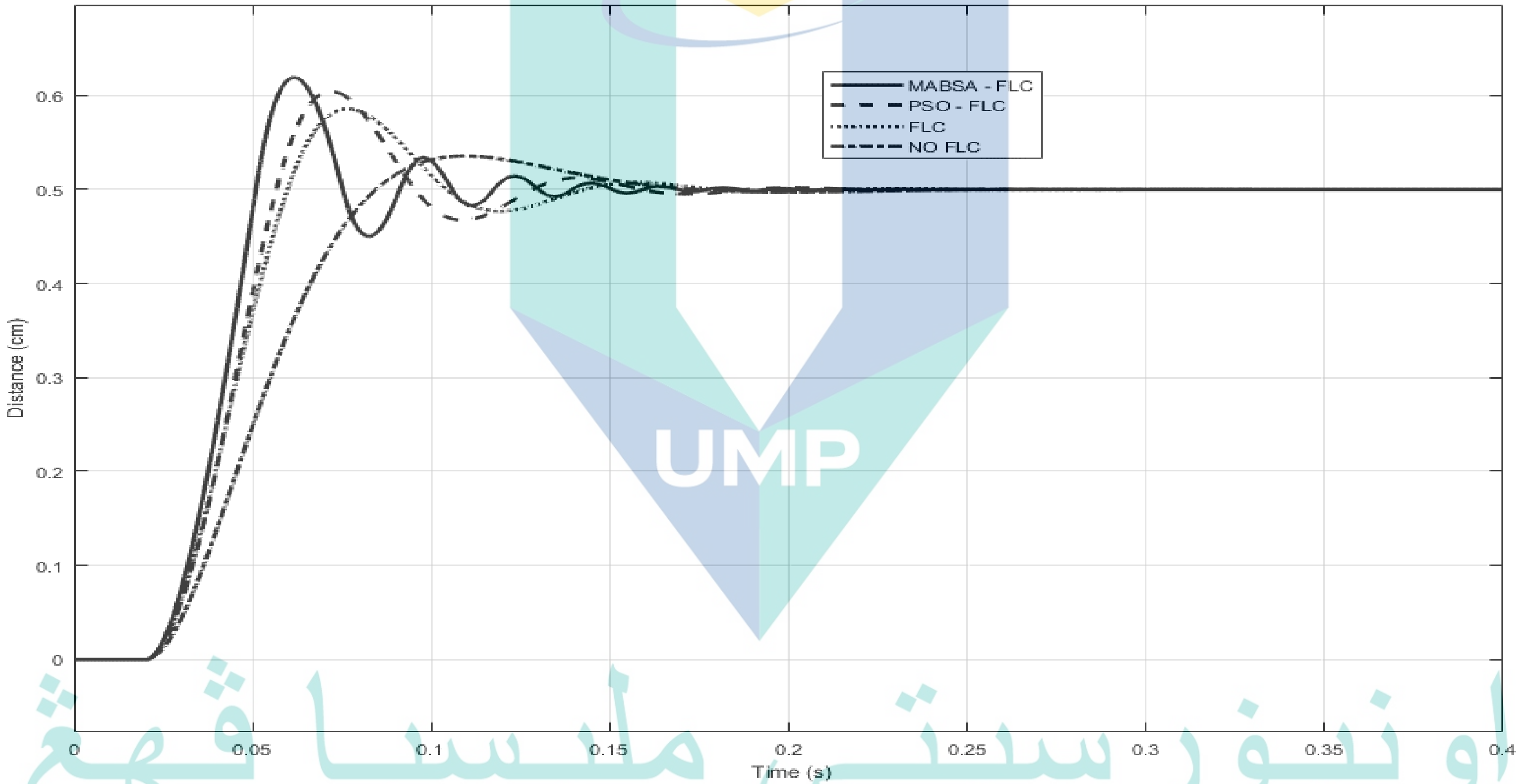

Figure 4.2 The comparison result from the MABSA-FLC and PSO-FLC

The comparison results are displayed as shown in Table 4.2. The values of the response parameters similar to the simulation result are also being analysed.

Table 4.2 The comparison result between MABSA and PSO for step response graph

Characteristics	Values			
	NO FLC	FLC	PSO-FLC	MABSA-FLC
Rise Time (s)	0.08	0.07	0.06	0.05
Settling Time (s)	0.20	0.18	0.15	0.13
Overshoot (%)	0.30	0.90	1.00	1.20
Steady State Error	0	0	0.6	0

As displayed in the Table above, almost 12.5% of the rise time by the system that used FLC optimized by MABSA compared with using PSO-FLC for the simulation result when without controller is used as an indicator. When FLC is used only as an indicator, MABSA-FLC still shows the best in rise time by the difference of 14.3%. The most difference in percentage is the parameter settling time. The difference is 0.02 when comparing the system with PSO-FLC. And of course the pattern is still the same where system that has been employed by FLC optimized by MABSA gives better result compared to system with FLC only. Although MABSA-FLC gives higher overshoot percentage which is 1.2% compared to other cases, the overshoot percentage requirement is still under limit. And the most important specification is that the steady state error is zero for the MABSA-FLC.

It was expected that from the simulation results, the subsystems that used fuzzy logic controller optimized by MABSA has a faster response time compared to the system without FLC, system with FLC only and system with FLC optimized by PSO algorithm. It is also clearly shown that for the optimization by algorithm, MABSA yields more improvement in terms of rising time and also settling time compared to the PSO algorithm. In summary, system with FLC optimized by MABSA has the best step response performance for all cases.

4.2.3 Robustness Test with Input Signal Comparative Results

There are two types of different sources used for analysing the performance such as sine-wave and pulse generator. Although the most sources used to evaluate the performance of step response is step, other sources will be added so that the performance of design MABSA-FLC can be discussed further. Figure 4.3 shows the output from sine-wave signal. The extended graph is shown in Figure 4.4. As can be seen by both figures, MABSA-FLC provides better direction with the sine wave input signal compared to other cases. The line MABSA-FLC is almost the same with the input reference which means that MABSA-FLC can proceed with input to output without much delay. The comparison results are displayed as shown in Table 4.3. The values of the response parameters similar to the simulation result are also being analysed.

Table 4.3 The comparison result between MABSA and PSO for sine wave graph

Characteristics	Values			
	NO FLC	FLC	PSO-FLC	MABSA-FLC
Rise Time (s)	0.9	0.8	0.7	0.6
Settling Time (s)	9.8	9.7	9.4	9.1
Overshoot (%)	0	0	0	0
Steady State Error	0	0	0	0

Overall, the rise time for MABSA-FLC is 0.1 seconds and 0.3 seconds higher than PSO-FLC and NO-FLC respectively. When FLC is used only as an indicator, MABSA-FLC still shows the best in rise time by the difference of 12.5% when comparing to PSO-FLC. The most difference in percentage is the parameter settling time. The difference is 0.3 when comparing the system with PSO-FLC. And of course the pattern is still the same where system that has been employed by FLC optimized by MABSA gives better result compared to system with FLC only. For all cases, there is no overshoot and steady state error.

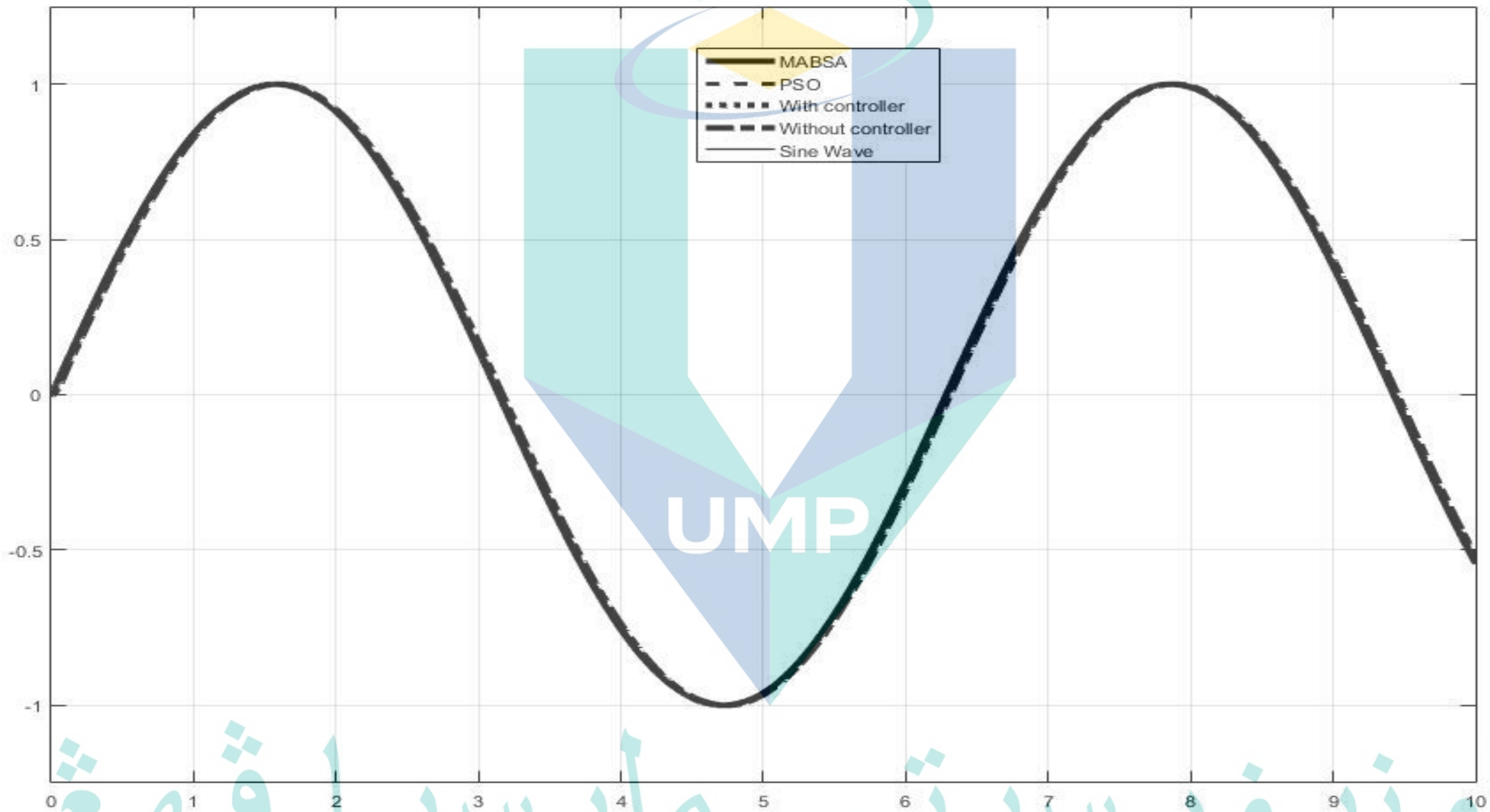

Figure 4.3 Sine wave graph for DC servo motor position control

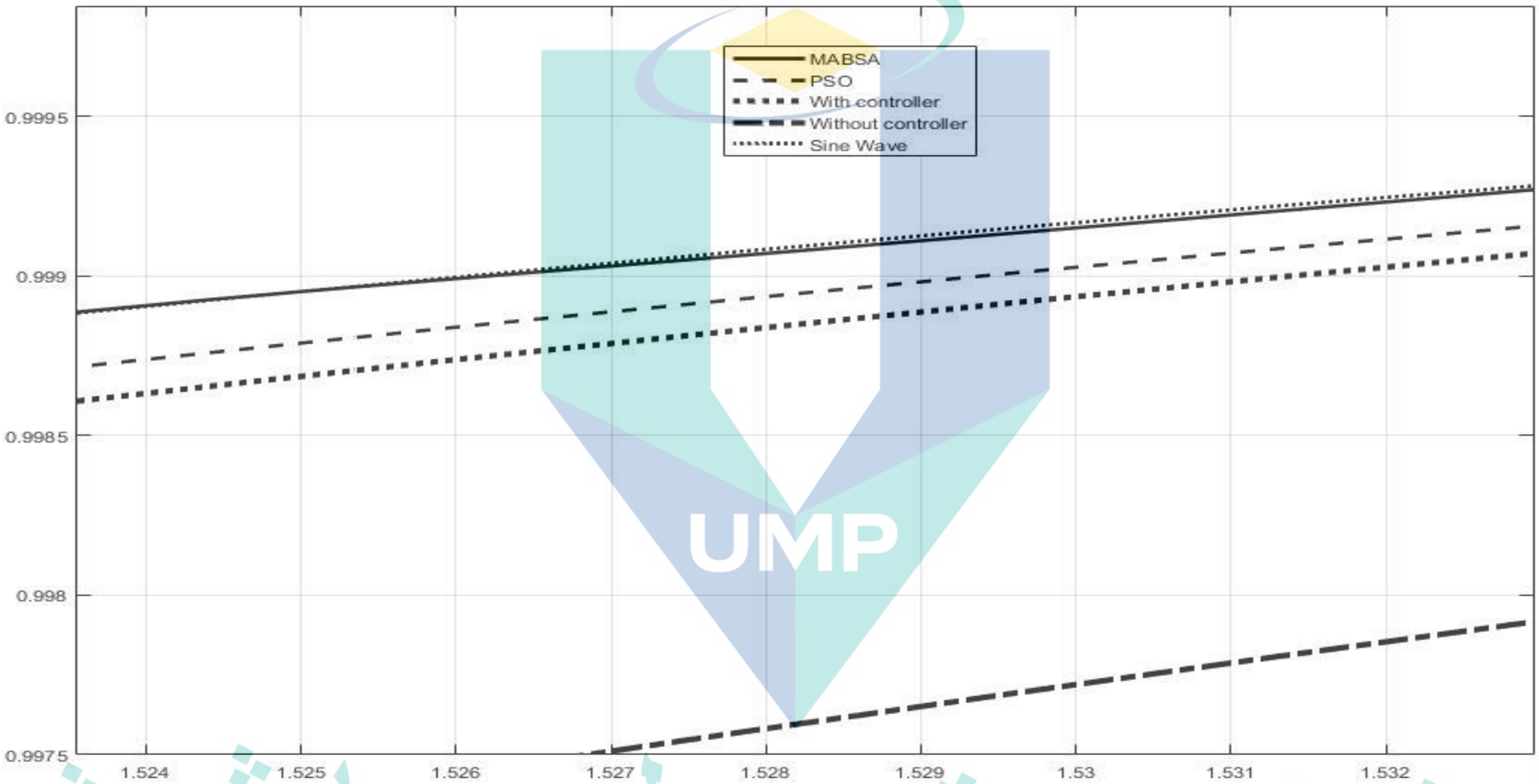

Figure 4.4 Extended sine wave graph for DC servo motor position control

Pulse generator has also been added to see the performance of MABSA-FLC. Figure 4.5 shows the output from pulse generator signal. The extended graph is shown in Figure 4.6 and Figure 4.7. As can be seen in Figure 4.5, all cases have no steady state error and returned back normally when disturbance is applied to it. However, MABSA-FLC gives better performance in rise time and settling time for both during increment and decrement of the input reference compared to other cases. Compared to PSO-FLC, MABSA-FLC is way faster in response.

The details comparison results are displayed as shown in Table 4.4. The values of the response parameters similar to the simulation result are also being analysed.

Table 4.4 The comparison result between MABSA and PSO for pulse generator graph

Characteristics	Values			
	NO FLC	FLC	PSO-FLC	MABSA-FLC
Rise Time (s)	0.09	0.07	0.06	0.05
Settling Time (s)	0.21	0.19	0.18	0.16
Overshoot (%)	5	10	12	14
Steady State Error	0	0	0	0

From the Table 4.4 above, almost 17% of the rise time by the system that used FLC optimized by MABSA compared with using PSO-FLC for the simulation result when without controller is used as an indicator. For the settling time, MABSA-FLC give the best time by only 0.16 seconds compared to 3 other cases. Although like the result from the other cases, MABSA-FLC shows the highest percentage in overshoot but as the specification that has been set, the percentage still under limit.

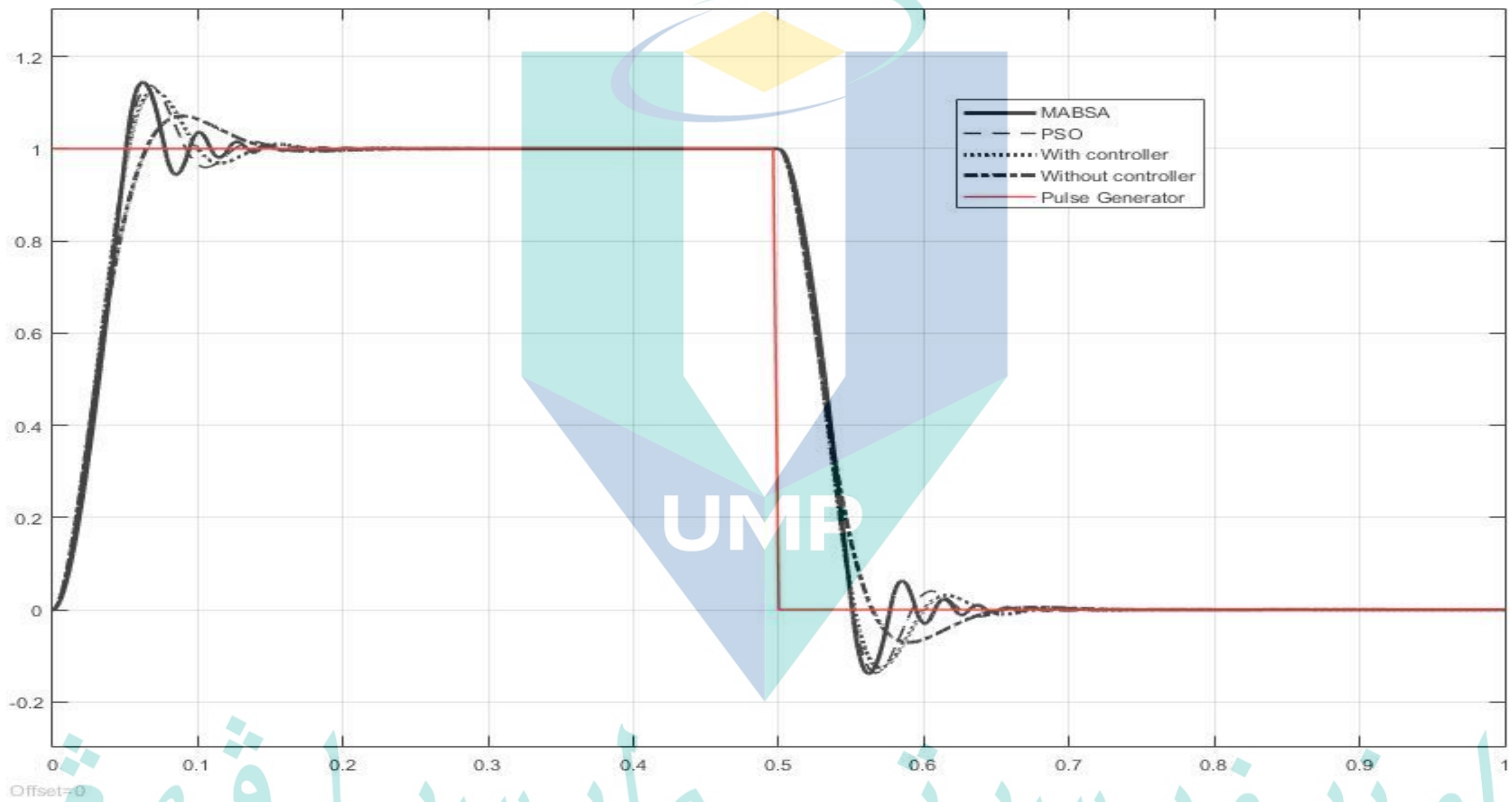

Figure 4.5 Pulse generator graph for DC servo motor position control

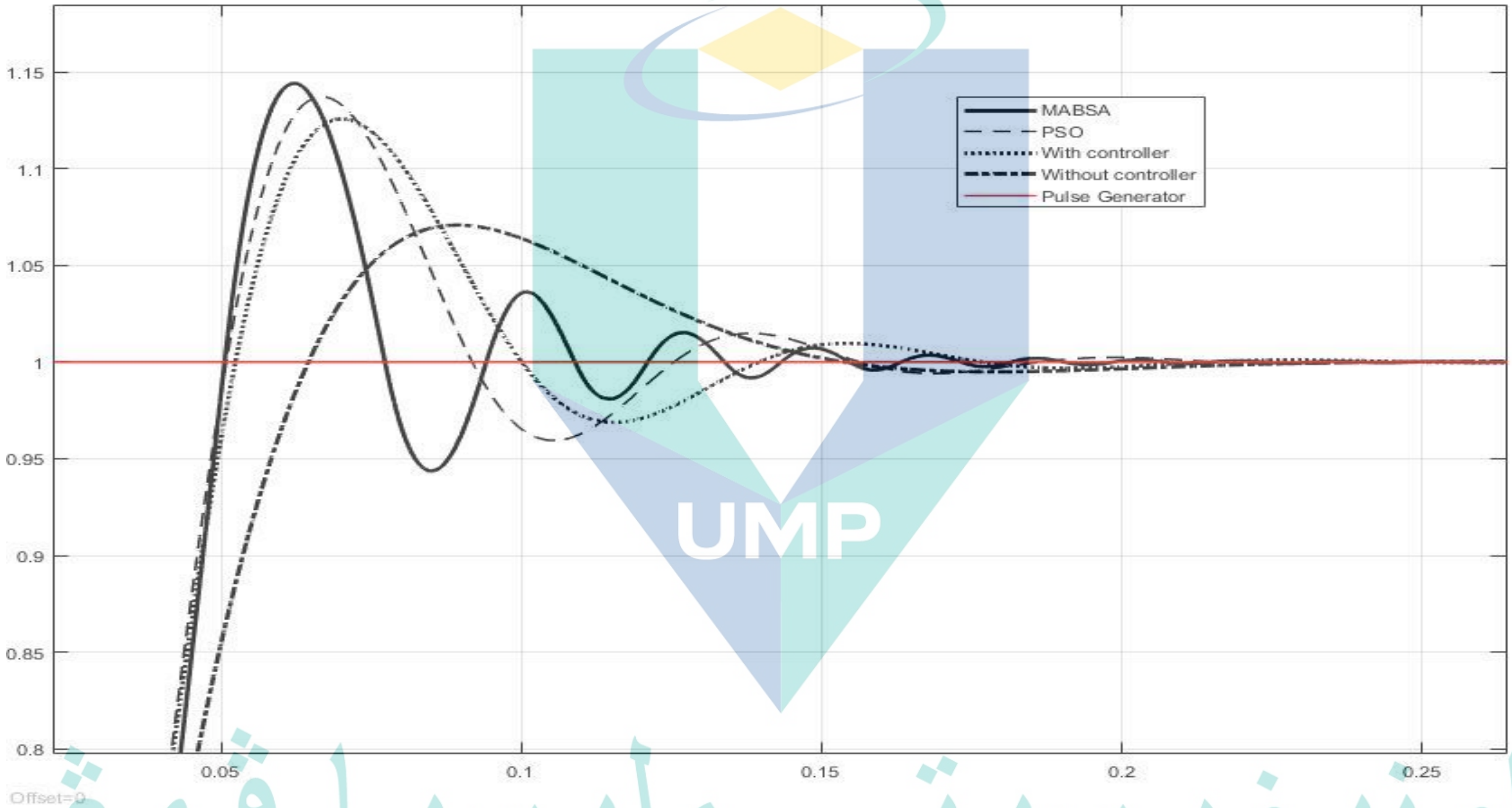

Figure 4.6 Extended pulse generator graph for DC servo motor position control during increment

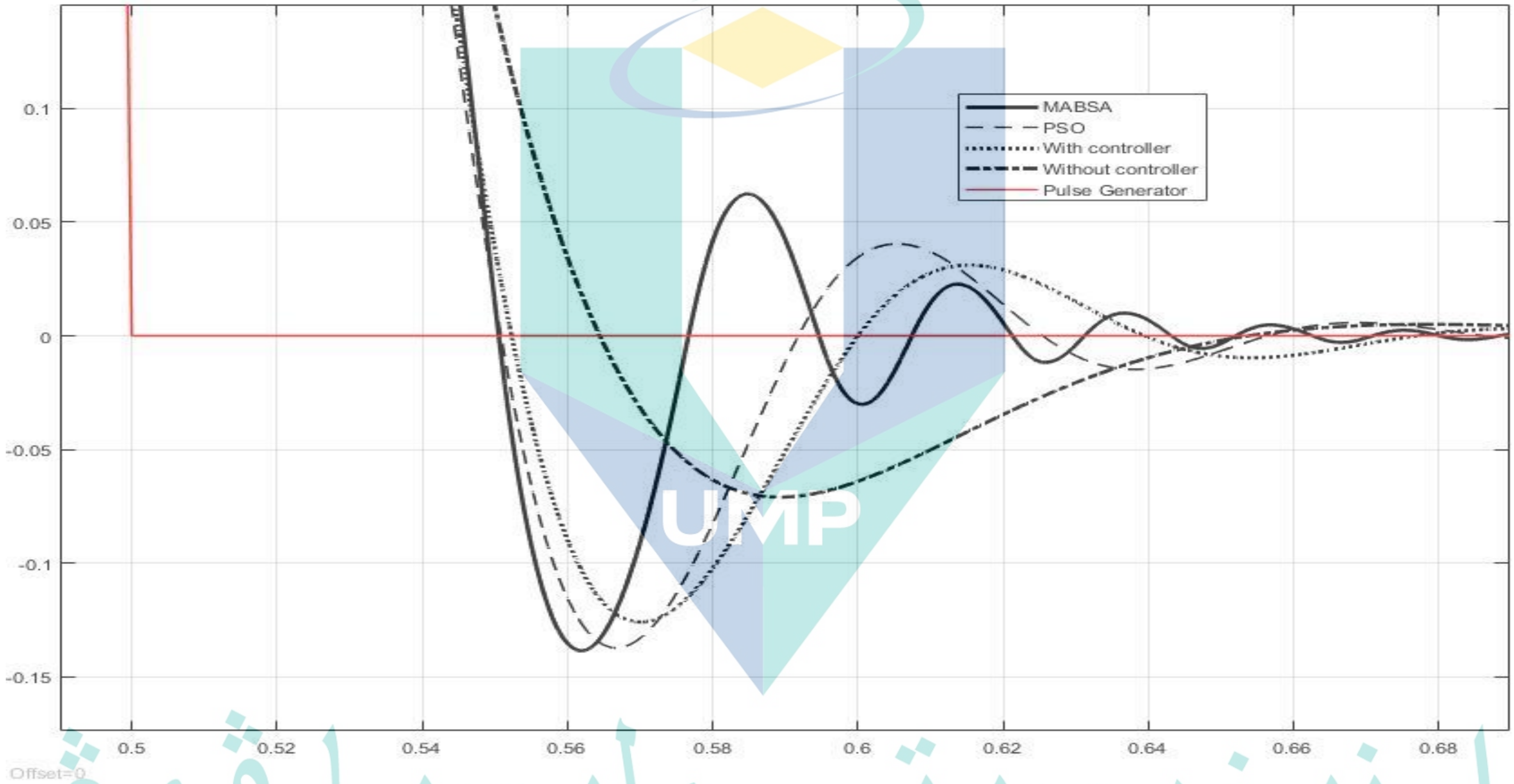

Figure 4.7 Extended pulse generator graph for DC servo motor position control during decrement

4.3 Summary

This chapter has presented the overall results for the simulation. The chapter continued with the comparative results that were divided into 2 cases which were without FLC being used as a reference point and with only FLC being used as a reference point. The performance criteria being compared were the rise time, settling time and percentage of overshoot. The proposed FLC optimized by MABSA clearly produced better response for both simulation and experiment validation.

The next chapter is the thesis summarization. The chapter concludes the research that has been conducted and presents the future works.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Introduction

This chapter presents the overall research summarization, conclusion and recommendations for future works.

5.2 Research Summary and Conclusions

A study of the improvement DC servo motor position control using fuzzy logic controller (FLC) optimized by a modified adaptive bats sonar algorithm (MABSA) has been presented. The work is focused on the designing of FLC and the modification of FLC by inserting the optimal parameters using MABSA.

The main reason for this research to be carried out is because the fuzzy inference system still lacking in terms of the accuracy and time performances. Therefore, the purposes of this research are to improve the fuzzy inference system performance by utilizing the DC servo motor. The research proposed a system model using a Simulink block diagram consisted of FLC block and DC servo motor transfer function block. For a newly proposed FLC with MABSA, optimization process has first taken place by adding the optimal value of parameters in the defuzzification process. The research has also developed a system model to simulate and experiment the actual system behaviours.

In relation to the objectives of the research: The research had successfully developed FLC optimized by MABSA that can operate efficiently and delivered the result depend on the requirement of a user (fuzzy rules). Objective 1 had successfully achieved by developing the FLC optimized by MABSA that can operate efficiently and delivered the result depend on the requirement of a user (fuzzy rules) and the MABSA had successfully obtained the optimum value for the use in designing the FLC.

Objective 2 had successfully achieved since the simulation results verified that proposed FLC optimized by MABSA improved 14.3% of rise time for simulation compared to the system that used FLC only without the optimization. For the pulse generator result, almost 17% of the rise time by the system that used FLC optimized by MABSA compared with using PSO-FLC for the simulation result when without controller is used as an indicator.

Objective 3 also successfully done by the simulation results that verified the proposed FLC optimized by MABSA improved the system performance compared to PSO-FLC in robustness test. The simulation results verified that the proposed FLC optimized by MABSA improved the system performance compared to PSO-FLC.

5.3 Future Direction of Research

The new design of FLC optimized by MABSA can be extended for future work by including the following which is the proposed FLC optimized by MABSA can be tested in different types of application that requires positional accuracy and time minimization such as cutting machines.

Secondly, the optimization by MABSA can be verified with other swarm intelligence algorithms to determine whether the value of parameters produced is at optimum level.

And lastly, the design of FLC can be compared with other available existing controllers such as PID controller or Neutral network controller. This comparison can show whether FLC can give better or poor performance in the position control system application.

5.4 Research Contribution and Publications

The scientific contributions to knowledge of this research include the following:

1. Fuzzy logic controller (FLC) optimized by MABSA for solving accuracy and time response problem.

2. Fuzzy logic controller (FLC) optimized by MABSA that can be used for DC servo motor position control application.

3. Fuzzy logic controller (FLC) optimized by MABSA that can solve fuzzy inference system problem with 2 input parameters.

Several publications have been produced through the research course that include the following:

1. Elias, N., Yahya, N. M., and Sing, E. H. (2018). Numerical Analysis of Fuzzy Logic Temperature and Humidity Control System in Pharmaceutical Warehouse Using MATLAB Fuzzy Toolbox. In Intelligent Manufacturing & Mechatronics (pp. 623-629).

2. Elias, N., and Yahya, N. M. (2018). Simulation Study for Controlling Direct Current Motor Position Utilising Fuzzy Logic Controller. International Journal of Automotive and Mechanical Engineering, 15(4), (pp. 5989-6000).

3. Elias, N., and Yahya, N. M. (2019). Comparison of DC Motor Position Control Simulation using MABSA-FLC and PSO-FLC. In 2019 IEEE 15th International Colloquium on Signal Processing & Its Applications (CSPA) (pp. 39-42).

4. Elias, N., and Yahya, N.M. Fast Response Fuzzy Logic Controller Optimized by Bats Sonar Algorithm. In SN Applied Sciences (submitted on 3th December 2019, under review).

REFERENCES

Abonyi, J., Babuska, R., & Szeifert, F. (2002). Modified Gath-Geva fuzzy clustering for identification of Takagi-Sugeno fuzzy models. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 32(5), 612-621.

Afjei, E., Ghomsheh, A. N., & Karami, A. (2007). Sensorless speed/position control of brushed DC motor. In 2007 International Aegean Conference on Electrical Machines and Power Electronics (pp. 730-732).

Akbar, M. A. (2014). Simulation of fuzzy logic control for DC servo motor using Arduino based on MATLAB/Simulink. In 2014 International Conference on Intelligent Autonomous Agents, Networks and Systems (pp. 42-46).

Akhtar, S., Ahmad, A. R., & Abdel-Rahman, E. M. (2012). A metaheuristic bat-inspired algorithm for full body human pose estimation. In Proceedings of the 2012 9th Conference on Computer and Robot Vision, CRV 2012 (pp. 369–375).

Ali, J. A., Hannan, M. A., Mohamed, A., & Abdolrasol, M. G. (2016). Fuzzy logic speed controller optimization approach for induction motor drive using backtracking search algorithm. Measurement, 78, 49-62.

Al-Odienat, A. I., & Al-Lawama, A. A. (2008). The advantages of PID fuzzy controllers over the conventional types. American Journal of Applied Sciences, 5(6), 653-658.

Angelov, P., & Buswell, R. (2002). Identification of evolving fuzzy rule-based models. IEEE Transactions on Fuzzy Systems, 10(5), 667–677.

Ang, K. H., Chong, G., & Li, Y. (2005). PID control system analysis, design, and technology. IEEE transactions on control systems technology, 13(4), 559-576.

Bachache, N. K., & Wen, J. (2013). Design fuzzy logic controller by particle swarm optimization for wind turbine. In International Conference in Swarm Intelligence (pp. 152-159).

Bhatt, P., Mehar, H., & Sahajwani, M. (2019). Electrical Motors for Electric Vehicle–A Comparative Study. Proceedings of Recent Advances in Interdisciplinary Trends in Engineering & Applications (RAITEA).

Bhushan, B., & Singh, M. (2011). Adaptive control of DC motor using bacterial foraging algorithm. Applied Soft Computing, 11(8), 4913-4920.

Bindu, R., & Namboothiripad, M. K. (2012). Tuning of PID controller for DC servo motor using genetic algorithm. International Journal of Emerging Technology and Advanced Engineering, 2(3), 310-314.

Blej, M., & Azizi, M. (2016). Comparison of Mamdani-type and Sugeno-type fuzzy inference systems for fuzzy real time scheduling. International Journal of Applied Engineering Research, 11(22), 11071-11075.

Cao, Y. Y., & Frank, P. M. (2001). Stability analysis and synthesis of nonlinear time-delay systems via linear Takagi–Sugeno fuzzy models. Fuzzy sets and systems, 124(2), 213-229.

Chan, C. C., & Wong, Y. S. (2004). The state of the art of electric vehicles technology. In The 4th International Power Electronics and Motion Control Conference, 2004. IPEMC 2004. (Vol. 1, pp. 46-57). IEEE.

Chang, F. J., & Chang, Y. T. (2006). Adaptive neuro-fuzzy inference system for prediction of water level in reservoir. Advances in water resources, 29(1), 1-10.

Chen, C. W., Yeh, K., Chiang, W. L., Chen, C. Y., & Wu, D. J. (2007). Modeling, H∞ control and stability analysis for structural systems using Takagi-Sugeno fuzzy model. Journal of Vibration and Control, 13(11), 1519-1534.

Chen, T. C., & Sheu, T. T. (2002). Model reference neural network controller for induction motor speed control. IEEE Transactions on Energy Conversion, 17(2), 157-163.

Cheng, P. C., Peng, B. R., Liu, Y. H., Cheng, Y. S., & Huang, J. W. (2015). Optimization of a fuzzy-logic-control-based MPPT algorithm using the particle swarm optimization technique. Energies, 8(6), 5338-5360.

Coello Coello, C. A. (2000). Constraint-handling using an evolutionary multiobjective optimization technique. Civil Engineering Systems, 17(4), 319-346.

Dai, Z., Lin, W., & Lin, H. (2015). Estimation of single-phase grid voltage parameters with zero steady-state error. IEEE Transactions on Power Electronics, 31(5), 3867-3879.

Dashti, H. T., Simas, T., Ribeiro, R. A., Assadi, A., & Moitinho, A. (2010). MK-means-Modified K-means clustering algorithm. In The 2010 International Joint Conference on Neural Networks (IJCNN) (pp. 1-6).

Denny, M. (2004). The physics of bat echolocation: Signal processing techniques. American Journal of Physics, 72(12), 1465–1477.

Dimeas, I., Petras, I., & Psychalinos, C. (2017). New analog implementation technique for fractional-order controller: a DC motor control. AEU-International Journal of Electronics and Communications, 78, 192-200.

Din, W. I. S. W., Yahya, S., Jailani, R., Taib, M. N., Yassin, A. I. M., & Razali, R. (2016). Fuzzy logic for cluster head selection in wireless sensor network. In AIP Conference Proceedings 1774 (1), (pp 050006).

Dorigo, M., & Di Caro, G. (1999). Ant colony optimization: a new meta-heuristic. Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406), 2, 1470–1477.

Du, H., & Zhang, N. (2008). Application of evolving Takagi–Sugeno fuzzy model to nonlinear system identification. Applied soft computing, 8(1), 676-686.

Espin-Andrade, R. A., Gonzalez, E., Pedrycz, W., & Fernandez, E. (2016). An Interpretable Logical Theory: The case of Compensatory Fuzzy Logic. International Journal of Computational Intelligence Systems, 9(4), 612–626.

Gandomi, A. H., Yang, X. S., Alavi, A. H., & Talatahari, S. (2013). Bat algorithm for constrained optimization tasks. Neural Computing and Applications, 22(6), 1239–1255.

George, M. (2008). Speed control of separately excited DC motor. American journal of applied sciences, 5(3), 227-233.

Hamam, A., & Georganas, N. D. (2008, October). A comparison of Mamdani and Sugeno fuzzy inference systems for evaluating the quality of experience of Hapto-Audio-Visual applications. In 2008 IEEE International Workshop on Haptic Audio visual Environments and Games (pp. 87-92).

Hernández-Guzmán, V. M., Silva-Ortigoza, R., & Muñoz-Carrillo, D. (2015). Velocity control of a brushed DC-motor driven by a DC to DC Buck power converter. International Journal of Innovative Computing, Information and Control, 509-521.

Herrera, F., Lozano, M., & Verdegay, J. L. (2005). Tuning fuzzy logic controllers by genetic algorithms. International Journal of Approximate Reasoning, 12(3–4), 299–315.

Hidrue, M. K., Parsons, G. R., Kempton, W., & Gardner, M. P. (2011). Willingness to pay for electric vehicles and their attributes. Resource and energy economics, 33(3), 686-705.

Huh, J. H., & Lee, H. G. (2018). Simulation and Test Bed of a Low-Power Digital Excitation System for Industry 4.0. Processes, 6(9), 145.

Ibrahim, B. S. K. K., Tokhi, M. O., Huq, M. S., Jailani, R., & Gharooni, S. C. (2011). Fuzzy modelling of knee joint with genetic optimization. Applied Bionics and Biomechanics, 8(1), 85-99.

Janga Reddy, M., & Nagesh Kumar, D. (2007). An efficient multi-objective optimization algorithm for engineering design. Engineering Optimization, 39(1), 49-68.

Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. Journal of Global Optimization, 39(3), 459–471.

Kaur, A., & Kaur, A. (2012). Comparison of mamdani-type and sugeno-type fuzzy inference systems for air conditioning system. International journal of soft computing and engineering, 2(2), 323-325.

Kelner, V., Capitanescu, F., Léonard, O., & Wehenkel, L. (2008). A hybrid optimization technique coupling an evolutionary and a local search algorithm. Journal of Computational and Applied Mathematics, 215(2), 448-456.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In Proceedings of ICNN'95-International Conference on Neural Networks (pp. 1942-1948).

Khan, S. A., & Engelbrecht, A. P. (2007). A new fuzzy operator and its application to topology design of distributed local area networks. Information Sciences, 177(13), 2692–2711.

Koçak, S., Tóth-Laufer, E., & Pokorádi, L. (2018). Comparison of the Defuzzification Methods in Risk Assessment Applications. In 2018 IEEE 18th International Symposium on Computational Intelligence and Informatics (CINTI) (pp. 000229-000234).

Komurcugil, H., Bayhan, S., & Abu-Rub, H. (2017). Variable-and fixed-switching-frequency-based HCC methods for grid-connected VSI with active damping and zero steady-state error. IEEE Transactions on Industrial Electronics, 64(9), 7009-7018.

Kothamasu, R., & Huang, S. H. (2007). Adaptive Mamdani fuzzy model for condition-based maintenance. Fuzzy sets and Systems, 158(24), 2715-2733.

Krishnan, R. (2001). Electric motor drives: modeling, analysis and control. Prentice Hall.

Krishnan, R. (2017). Permanent magnet synchronous and brushless DC motor drives. CRC press.

Kusagur, A., Kodad, S. F., & Ram, B. V. S. (2010). Modeling, design & simulation of an adaptive neuro-fuzzy inference system (ANFIS) for speed control of induction motor. International Journal of Computer Applications, 6(12), 29-44.

Lee, D. C., & Kim, Y. S. (2007). Control of single-phase-to-three-phase AC/DC/AC PWM converters for induction motor drives. IEEE transactions on industrial electronics, 54(2), 797-804.

Li, G., Hou, Y., & Wu, A. (2017). Fourth Industrial Revolution: technological drivers, impacts and coping methods. Chinese Geographical Science, 27(4), 626-637.

Li, Y., Tong, S., & Li, T. (2013). Adaptive fuzzy output feedback control for a single-link flexible robot manipulator driven DC motor via backstepping. Nonlinear Analysis: Real World Applications, 14(1), 483-494.

Liu, J., Zhang, P., & Wang, F. (2009). Real-time dc servo motor position control by pid controller using labview. In 2009 International Conference on Intelligent Human-Machine Systems and Cybernetics (pp. 206-209).

Lope, H. S., & Coelho, L. S. (2005). Particle swarn optimization with fast local search for the blind traveling salesman problem. In Fifth International Conference on Hybrid Intelligent Systems (HIS'05) (pp. 245-250). IEEE.

Maeda, M., & Murakami, S. (1992). A self-tuning fuzzy controller. Fuzzy Sets and Systems, 51(1), 29–40.

Mahdavi, M., Fesanghary, M., & Damangir, E. (2007). An improved harmony search algorithm for solving optimization problems. Applied mathematics and computation, 188(2), 1567-1579.

Malakooti, B., Kim, H., & Sheikh, S. (2012). Bat intelligence search with application to multi-objective multiprocessor scheduling optimization. International Journal of Advanced Manufacturing Technology, 60(9–12), 1071–1086.

Mamdani, E. H., & Assilian, S. (1999). An experiment in linguistic synthesis with a fuzzy logic controller. International journal of human-computer studies, 51(2), 135-147.

Manikandan, R., & Arulmozhiyal, R. (2014). Position control of DC servo drive using fuzzy logic controller. In Advances in Electrical Engineering (ICAEE), 2014 International Conference (pp. 1-5).

Maraiya, G., & Ray, S. (2018). Control of a DC-Servo Motor Using PID Controller with System Indetification Technique. Journal of Control System and its Recent Developments, 1(3), 24-29.

Mavrovouniotis, M., Li, C., & Yang, S. (2017). A survey of swarm intelligence for dynamic optimization: Algorithms and applications. Swarm and Evolutionary Computation, 33, 1-17.

Mayr, A., Weigelt, M., von Lindenfels, J., Seefried, J., Ziegler, M., Mahr, A., & Franke, J. (2018). Electric Motor Production 4.0–Application Potentials of Industry 4.0 Technologies in the Manufacturing of Electric Motors. In 2018 8th International Electric Drives Production Conference (EDPC) (pp. 1-13).

Meike, D., & Ribickis, L. (2011). Energy efficient use of robotics in the automobile industry. In 2011 15th international conference on advanced robotics (ICAR) (pp. 507-511).

Meshram, P. M., & Kanojiya, R. G. (2012). Tuning of PID controller using Ziegler-Nichols method for speed control of DC motor. In IEEE-international conference on advances in engineering, science and management (ICAESM-2012) (pp. 117-122).

Mirjalili, S. (2016). Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Computing and Applications, 27(4), 1053-1073.

Monmasson, E., & Cirstea, M. N. (2007). FPGA Design Methodology for Industrial Control Systems - A Review. IEEE Transactions on Industrial Electronics, 54(4), 1824–1842.

Morgan, L. K., Stoeckl, L., Werner, A. D., & Post, V. E. (2013). An assessment of seawater intrusion overshoot using physical and numerical modeling. Water Resources Research, 49(10), 6522-6526.

Mosterman, P. J., & Zander, J. (2016). Industry 4.0 as a cyber-physical system study. Software & Systems Modeling, 15(1), 17-29.

Murray, R. M. (2017). A mathematical introduction to robotic manipulation. CRC press.

Nam, K. Y., Lee, W. T., Lee, C. M., & Hong, J. P. (2006). Reducing torque ripple of brushless DC motor by varying input voltage. IEEE transactions on magnetics, 42(4), 1307-1310.

Nanda, J., & Mangla, A. (2004). Automatic generation control of an interconnected hydro-thermal system using conventional integral and fuzzy logic controller. In 2004 IEEE international conference on electric utility deregulation, restructuring and power technologies. Proceedings (pp. 372-377).

Nasri, M., Nezamabadi-Pour, H., & Maghfoori, M. (2007). A PSO-based optimum design of PID controller for a linear brushless DC motor. World Academy of Science, Engineering and Technology, 26(40), 211-215.

Ozturk, S. B., & Toliyat, H. A. (2010). Direct torque and indirect flux control of brushless DC motor. IEEE/ASME Transactions on Mechatronics, 16(2), 351-360.

Park, K. H., Kim, T. S., Ahn, S. C., & Hyun, D. S. (2003, June). Speed control of high-performance brushless DC motor drives by load torque estimation. In IEEE 34th Annual Conference on Power Electronics Specialist, 2003. PESC'03. (pp. 1677-1681).

Passino, K. M., Yurkovich, S., & Reinfrank, M. (1998). Fuzzy control (pp. 15-21). Menlo Park, CA: Addison-wesley.

Peters, B. A., & Yang, T. (2007). Integrated facility layout and material handling system design in semiconductor fabrication facilities. IEEE Transactions on Semiconductor Manufacturing, 10(3), 360–369.

Polat, A., Ergene, L. T., & Firat, A. (2011). Dynamic modeling of the universal motor used in washer. In International Aegean Conference on Electrical Machines and Power Electronics and Electromotion, Joint Conference (pp. 444-448).

Prabu, M. J., Poongodi, P., & Premkumar, K. (2016). Fuzzy supervised online coactive neuro-fuzzy inference system-based rotor position control of brushless DC motor. IET Power Electronics, 9(11), 2229-2239.

Premkumar, K., & Manikandan, B. V. (2015). Speed control of Brushless DC motor using bat algorithm optimized Adaptive Neuro-Fuzzy Inference System. Applied Soft Computing, 32, 403–419.

Qureshi, M. S., Swarnkar, P., & Gupta, S. (2016). Assessment of DC servo motor with sliding mode control approach. In 2016 IEEE First International Conference on Control, Measurement and Instrumentation (CMI) (pp. 351-355). IEEE.

Rad, M. K., Torabizadeh, M., & Noshadi, A. (2011). Artificial Neural Network-based fault diagnostics of an electric motor using vibration monitoring. In Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE) (pp. 1512-1516).

Raghav, G., Farooq, M., Khan, A. A., & Muzammil, M. (2011). Ergonomic modification and evaluation of chain saw handle in wood cutting. Canadian Acoustics, 39(2), 90–91.

Rahmani, R., Mahmodian, M. S., Mekhilef, S., & Shojaei, A. A. (2012). Fuzzy logic controller optimized by particle swarm optimization for DC motor speed control. In Research and Development (SCOReD), 2012 IEEE Student Conference (pp. 109-113).

Rini, D. P., Shamsuddin, S. M., & Yuhaniz, S. S. (2011). Particle swarm optimization: technique, system and challenges. International journal of computer applications, 14(1), 19-26.

Rong, H. J., Sundararajan, N., Huang, G. B., & Zhao, G. S. (2011). Extended sequential adaptive fuzzy inference system for classification problems. Evolving Systems, 2(2), 71-82.

Scott, J., McLeish, J., & Round, W. H. (2008). Speed control with low armature loss for very small sensorless brushed DC motors. IEEE Transactions on industrial electronics, 56(4), 1223-1229.

Shanmugasundram, R., Zakariah, K. M., & Yadaiah, N. (2012). Implementation and performance analysis of digital controllers for brushless DC motor drives. IEEE/ASME transactions on mechatronics, 19(1), 213-224.

Siarry, P., & Guely, F. (1998). A genetic algorithm for optimizing Takagi-Sugeno fuzzy rule bases. Fuzzy Sets and Systems, 99(1), 37–47.

Song, J. H., & Choy, I. (2004). Commutation torque ripple reduction in brushless DC motor drives using a single DC current sensor. IEEE Transactions on Power Electronics, 19(2), 312-319.

Subramanian, K., Das, A. K., Sundaram, S., & Ramasamy, S. (2014). A meta-cognitive interval type-2 fuzzy inference system and its projection based learning algorithm. Evolving Systems, 5(4), 219-230.

Subramanian, K., Savitha, R., & Suresh, S. (2014). A complex-valued neuro-fuzzy inference system and its learning mechanism. Neurocomputing, 123, 110-120.

Subramanian, K., Suresh, S., & Sundararajan, N. (2013). A metacognitive neuro-fuzzy inference system (McFIS) for sequential classification problems. IEEE Transactions on Fuzzy Systems, 21(6), 1080-1095.

Szabat, K., & Orlowska-Kowalska, T. (2007). Vibration suppression in a two-mass drive system using PI speed controller and additional feedbacks—Comparative study. IEEE Transactions on Industrial Electronics, 54(2), 1193-1206.

Tawfeeq, M. A. (2012). Intelligent Algorithm for Optimum Solutions Based on the Principles of Bat Sonar, 10(10), 11–19.

Teimouri, R., & Baseri, H. (2015). Forward and backward predictions of the friction stir welding parameters using fuzzy-artificial bee colony-imperialist competitive algorithm systems. Journal of Intelligent Manufacturing, 26(2), 307–319.

Thomas, N., & Poongodi, D. P. (2009). Position control of DC motor using genetic algorithm based PID controller. In Proceedings of the World Congress on Engineering (Vol. 2, pp. 1-3).

Tomar, A., Sood, Y. R., & Tomar, D. S. (2014). An efficient scheme of automation and control for conventional cable manufacturing industry. IEEE Transactions on Industrial Informatics, 11(1), 132-140.

Tsai, P. W., Pan, J. S., Liao, B. Y., Tsai, M. J., & Istanda, V. (2012). Bat algorithm inspired algorithm for solving numerical optimization problems. In Applied mechanics and materials (Vol. 148, pp. 134-137). Trans Tech Publications Ltd.

Uraon, K. K., & Kumar, S. (2016). Analysis of defuzzification method for rainfall event. International Journal of Computer Science and Mobile Computing, 1(1), 341-354.

Vazquez-Sanchez, E., Gomez-Gil, J., Gamazo-Real, J. C., & Diez-Higuera, J. F. (2011). A new method for sensorless estimation of the speed and position in brushed DC motors using support vector machines. IEEE Transactions on Industrial Electronics, 59(3), 1397-1408.

Vyas, D., Misra, Y., & Kamath, H. R. (2015). Comparison and analysis of defuzzification methods of a fuzzy controller to maintain the cane level during cane juice extraction. In 2015 International Conference on Signal Processing and Communication Engineering Systems (pp. 102-106). IEEE.

Wang, Y., & Chen, Y. (2014). A comparison of Mamdani and Sugeno fuzzy inference systems for traffic flow prediction. Journal of computers, 9(1), 12-21.

Wang, H., Qian, G., & Feng, X. Q. (2013). Predicting consumer sentiments using online sequential extreme learning machine and intuitionistic fuzzy sets. Neural Computing and Applications, 22(3-4), 479-489.

Wieczorek, D. (2018). Fuzzy risk assessment in the life cycle of building object-Selection of the right defuzzification method. In AIP Conference Proceedings (pp. 240005).

Williams, J. K. (2009). Introduction to fuzzy logic. In Artificial Intelligence Methods in the Environmental Sciences (pp. 127–151).

Xu, L. D., Xu, E. L., & Li, L. (2018). Industry 4.0: state of the art and future trends. International Journal of Production Research, 56(8), 2941-2962.

Xue, X. D., Cheng, K. W. E., & Cheung, N. C. (2008). Selection of electric motor drives for electric vehicles. In 2008 Australasian Universities Power Engineering Conference (pp. 1-6).

Yadav, S. K. (2015). DC motor position control using fuzzy proportional-derivative controllers with different defuzzification methods. IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE) e-ISSN, 10(1), 2278-1676.

Yahya, N., & Osman Zahid, M. N. (2016). Application of Adaptive Bats Sonar Algorithm for Solving a Single Objective of Practical Business Optimisation. Algorithm for Solving a Single Objective of Practical Business Optimisation, (December), 28–29.

Yahya, N. M., & Tokhi, M. O. (2017). A modified bats echolocation-based algorithm for solving constrained optimisation problems. International Journal of Bio-Inspired Computation, 10(1), 12-23.

Yahya, N. M., Tokhi, M. O., & Kasdirin, H. A. (2016). A new bats echolocation-based algorithm for single objective optimisation. Evolutionary intelligence, 9(1-2), 1-20.

Yedamale, P. (2003). Brushless DC (BLDC) motor fundamentals. Microchip Technology Inc, 20, 3-15.

Yen, J., & Pfluger, N. (1995). A Fuzzy Logic Based Extension to Payton and Rosenblatt's Command Fusion Method for Mobile Robot Navigation. IEEE Transactions on Systems, Man, and Cybernetics, 25(6), 971–978.

Yin, Y., Stecke, K. E., & Li, D. (2018). The evolution of production systems from Industry 2.0 through Industry 4.0. International Journal of Production Research, 56(1-2), 848-861.

Zaher, H., Kandil, A. E., & Fahmy, R. (2014). Comparison of mamdani and sugeno fuzzy inference systems for prediction (with application to prices of fund in egypt). Journal of Advances in Mathematics and Computer Science, 3014-3022.