SOCIAL SPIDER OPTIMIZATION ON PID CONTROLLER

NOOR DIYANA BINTI YAACOB

UNIVERSITI SAINS MALAYSIA

2017

SOCIAL SPIDER OPTIMIZATION ON PID CONTROLLER

by

NOOR DIYANA BINTI YAACOB

Thesis submitted in partial fulfillment

of the requirements for the degree of

Bachelor of Engineering (Mechatronics Engineering)

JUNE 2017

ACKNOWLEDGEMENT

Alhamdulillah, all praises to Allah for endowing me with health, strength, and knowledge to complete this research. Here, I would like to acknowledge some individuals and parties who had given me the opportunity to gain invaluable experience during my final year project.

Foremost, I would like to express my sincere gratitude to my advisor Dr. Wan Amir Fuad Wajdi Bin Othman for the continuous support of my study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and guidance for my study.

Finally, I must express my very profound gratitude to my beloved mother Faridah Binti Nordin as well as my siblings for the moral and eternal supports and to my friends for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

TABLE OF CONTENTS

APPENDIX

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF SYMBOLS

SYMBOLS

Kd	Differential gain
Ki	Integral gain
Kp	Proportional gain
Mp	Peak overshoot
e_{ss}	Steady-state error
dims	Number of decision variables
runtime	Number of runs
Tr	Rise time
Ts	Settling time
spidn	Number of spiders (swarm size)
fn	Number of female spiders
mn	Number of male spiders
itern	Number of iterations
rand	random number between [0,1].
ρ	Weighing factor of cost function

ABBREVIATIONS

ACO Ant Colony Optimisation

GA Genetic Algorithm

PID Proportional-Integral-Differential

PSO Particle Swarm Optimisation

SOCIAL SPIDER ALGORITHM PADA PENGAWAL PID

ABSTRAK

Pengawal PID adalah jenis yang paling banyak digunakan pengawal untuk aplikasi industri. Mereka adalah struktur mudah dan mempamerkan prestasi kukuh ke atas pelbagai keadaan operasi. PID adalah kawalan gelung maklum balas sistem mekanisme yang biasanya digunakan dalam sistem kawalan industri. Walau bagaimanapun, mencari optimum parameter pengawal PID adalah satu tugas yang rumit. Oleh itu, dalam usaha untuk menyelesaikan masalah fungsi pengoptimuman tinggi dimensi, banyak algoritma swarm perisikan telah dicadangkan. Antara pelbagai kecerdasan swarm, Sosial Spider Optimization (SSO) telah dipilih untuk kajian ini. Gabungan sempurna parameter kawalan yang dicadangkan boleh membawa kepada mengurangkan masalah yang dihadapi oleh kilang industri. Sambutan pengawal boleh digambarkan dari segi sejauh mana yang terlajak sistem puncak, Mp, keadaan mantap kesilapan, ess, masa naik, Tr dan masa penetapan, Ts. Di samping itu, prosedur yang sesuai telah ditubuhkan untuk menganalisis kualiti prestasi Sosial Spider Optimization (SSO) yang telah dilaksanakan secara meluas dalam masalah dunia sebenar. Algoritma ini berjaya dilaksanakan pada parameter mencari pengawal PID yang menunjukkan ketepatan dan kecekapan untuk menyelesaikan apa-apa pengiraan yang rumit.

SOCIAL SPIDER OPTIMISATION ON PID CONTROLLER

ABSTRACT

PID controllers are the most widely-used type of controller for industrial applications. They are structurally simple and exhibit robust performance over a wide range of operating conditions. PID is a control loop feedback mechanism systems which is commonly used in industrial control systems. However, finding optimal PID controller parameters is a complicated task. Thus, in order to solve high-dimensional function optimization problem, many swarm intelligence algorithms have been proposed. Among them, Social Spider Optimization (SSO) was chosen for this research. The perfect combination of control parameters proposed may lead to reducing the problems encountered by the industrial plants. The response of the controller can be described in terms of the degree to which the system peak overshoot, *Mp*, steady-state error, *ess*, rise time, *Tr* and settling time, *Ts.* In addition, appropriate procedures have been established to analyze the quality of Social Spider Optimization (SSO) performance that has been implemented widely in the real world problems. This algorithm is successfully applied in parameters searching for PID controller that shows the accuracy and efficiency to solve such complicated computations.

CHAPTER 1

INTRODUCTION

1.1 Project Overview

Chapter 1 introduces the overview of PID controllers that widely used as the chosen controller strategy due to their design simplicity and its reliable operation. The adjustment process of the values Kp, Ki and Kd is called 'tuning' of PID controller. A simple PID structure consists of three terms which are Kp, Ki and Kd referring to proportional, integrations, and d derivative gains respectively. The tuning approaches can be divided into two categories which are the conventional and the alternative approaches. The conventional approaches include the empirical methods and the analytical methods which widely used by control designers. The alternative approaches are limited to methods that employ the stochastic process in the tuning rules. Stochastic process refers to one whose behaviour is non-deterministic, where any of its sub-system determined by the process of deterministic action and a random behaviour.

Optimization techniques typically encountered in engineering applications as well known to search for an alternative with the most cost effective or highest achievable performance under a given constraints, by maximizing desired factors and minimizing undesired ones. In addition, the fast growing and complexity of modern optimization applications, evolutionary computing appears increasingly attractive as an efficient tool for optimization. Evolutionary computing algorithms can be classified into two important groups such that evolutionary algorithms (EAs) and swarm intelligence based algorithms.

Social Spider Optimization (SSO) was chosen in terms of optimizing Proportional-Integral-Differential (PID) controller to evaluate the robust of this algorithm on this system. Practically, every activity in our daily life is influenced by sort of control systems which played an important role such as industrial processes, robotics, and power system. Application of algorithm to the PID controller makes it give an optimum output by searching for the best set of solutions for the PID parameters. In essence, SSO was modeled based on the mating behavior among the colony of spiders in the communal web. They interact with each other to share the information with following the predetermined procedures. Firstly, each female checked all the spiders in the search space then start finding for the presence of stronger vibration in the communal web.

During mating process, male spider being checked either it is good or placed above the median in order to generate the new generation similarly or even called offspring. This operation is iterated based on mating and movement occurs till convergence to the optimal solution. The computational speed of the algorithm is to be considered to analyze the effectiveness and robustness. Also, the validity of this algorithm is to be analyzed.

Relative to this study is finding the best solutions to tune PID controller. Basically, a proportional–integral–derivative controller (PID) is a control loop feedback mechanism which is commonly used in industrial control systems. A PID controller continuously calculates the error value obtained in the system as the difference between the desired set point and a measured variable. Also, appeal for a correction derived from the proportional, integral and derivative proportional, integral, and derivative return of phrase which usually denoted as P,I and D respectively. Equation 1.1 represent as the ideal version of proportional, integral and derivative is given by the formula as the output of the PID controller such that the small changes occur in the system plant less sensitive.

$$\mathrm{u(t)} = \mathrm{Kp}e(t) + \mathrm{Ki}\int_0^t \mathrm{e}(\tau)\mathrm{d}\tau + \mathrm{Kd}\frac{\mathrm{d}}{\mathrm{dt}}e(t) \quad (1.1)$$

where K_p=proportional parameter, $K_{i=}$ integral parameter and $K_{d=}$differential parameter

The efficiency and robustness of these algorithms are to be analyzed, thus discover the optimal solutions to make sure that achieve the minimal value of rising time, Tr, settling time, T_s, overshoot and lower steady-state error in the system which able to solve problems occurs in the industrial plants. Besides that, there are many techniques proposed and used for tuning of PID parameters to find a solution of most problems happen such as nonlinearities, high order and time delays.

1.2 Problem Statement

Proportional-Integral Derivative (PID) controllers have been widely used for various applications of control such as regulate speed, position and temperature. The conventional methods that commonly used for tuning of PID controller are Ziegler-Nichols (ZN) which may produced a higher overshoot, therefore several intelligent approaches of modern heuristics has been proposed such as genetic algorithm (GA) and particle swarm optimization (PSO) to enhance the capability of traditional techniques. The parameter settings of a PID controller for optimal control of a plant depend on the plant's behaviour [7].Due to the computational drawbacks of mathematical calculation for examples complex derivatives, a large amount of desired enumeration memory, thus researchers depend on meta-heuristic algorithms on simulations and stochastic optimization technique to achieve optimum solutions [13]. Nevertheless, these methods tend to obtain an inaccurate result and in any case, do not give the optimal result with less effort and time [1].

Other than that, in real world problems such as structural optimization problems, a single function evaluation may require several hours to several days of complete simulation. In fact, most industrial facilities no longer tune loops with manual calculation, but use tuning and loop optimization software. Therefore, there is a need for an effective and efficient global approach to optimize these parameters automatically. In recent year, there have been many new algorithms were developed in order to recoup the drawbacks of other algorithms. Thus, a more competent algorithm is to be proposed as to ease the process of data analysis in this research.

1.3 Objectives of Research

1. To study the performance of Social Spider Optimisation (SSO) to optimize tuning of PID controller by analysing the minimisation of cost function and step responses of closed-loop systems.
2. To study the affect of control parameters of Social Spider Optimization (SSO) to find the optimal solutions for closed- loop systems.

SCOPE OF RESEARCH

This study focuses mainly on tuning the parameters of PID controller to achieve an optimum result which closely resembles the step input. A growing number of workload nowadays that related to the application of PID controllers become significantly popular and demanded in various fields of science and engineering. Besides that, fundamental of PID control theory lead to easy optimizing plant performance with some efficient strategies for controller implementation and include a consideration of some constraints.

Referring to certain conditions, the existing of tuning PID method is not capable of tuning the combination of PID parameters when facing different conditions. Thus, this research proposed an algorithm that automatically gives the researchers or engineers the optimized PID parameters. Tuning of a PID controller refers to the tuning of its various parameters (P, I and D) to achieve an optimized value of the desired response. Different plants have different requirements of these parameters which can be achieved by meaningful tuning of the PID parameter.

In order to obtain whether the PID parameters display desirable step response, certain criteria were evaluated which is cost function. The suitable control parameters should be chosen so that would obtain accurate results. The best choice for each of the tuning parameters Kp, K_I, and K_D depends on the values of the other two as well as the behavior of the controlled process. Analysing the optimization algorithm with several test problems and comparing to a well-known algorithm.

1.4 Thesis Outline

This thesis consists of five chapters which include Introduction, Literature Review, Methodology, Result and Discussion, Conclusion and Future work. Chapter 1 mainly explains about the project overview, problem statement, and objectives of this study. Next, Chapter 2 consists of the basic concept of optimization, optimization algorithms and categories of optimization methods such as heuristic and meta-heuristic methods. This part elaborates the development and implementation of Social Spider Optimisation (SSO) Algorithm in tuning PID controller. In addition, explained the impact of control parameters to optimize the cost and time by utilizing SSO. Chapter 3 presents the methodology, design, and implementation of SSO in this problem. Also, includes the set of variables and control parameters used as well as procedures in order to design solution via MATLAB. Chapter 4 simply presents the result of simulations and analyses the optimum solution for this system. Besides that, this section discusses about the minimisation of cost and objective function. Finally, Chapter 5 concludes the thesis and suggests several further investigations of the optimization work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter, explain the basic concept for fundamental of optimization that was created by others in several research papers which related to basic of optimization, evolution and philosophical basis of optimization algorithms also the concepts of swarm intelligence are presented in this chapter. Section 2.3.3 described a brief overview of optimizing the control parameters used in Social Spider Algorithm. Section 2.4 is the summary of chapter 2.

2.2 Concept of Optimisation

Optimization means searching for alternative options that promising most cost effective or better feasible performance under different constraints, either maximizing desired factors or minimizing the excess. In comparison, maximization trying to gain the highest outcome without regard the expense obtained. The practice of optimization is restricted by the insufficient information, and the lack of time to evaluate what information is available. Generally, optimization is attained by using linear programming techniques in a computer simulation (modelling). The practiceof optimization was restricted especially with either incomplete, imperfect information or limited computation capacity further lack of entire information, and take high elapsed time to evaluate the given information.

2.3 Optimisation Algorithms

Optimization has been applied in many fields of science, including engineering, economics, and logistics where optimal decisions need to be chosen in the presence of trade-offs between two or more multi-objectives functions. In essence, optimization means the selection of the best element with regard to some criterions or issues from variety set of available alternatives solutions. In addition, the algorithm would continuously search for the best solution until certain criteria are met. The main objective of the optimisation algorithm is to ensure that some subsequence of iterations converges to an optimal solution by finding values of the variables that minimize or maximize the objective function while satisfying the constraints. The feasible set of elements is generally defined by some constraint functions in the search spaces such that desire is to get as close as possible to the desired value of each objective. Nevertheless, in order to tend totally exploit and explore the search space, the optimization algorithm must have a system to balance between local search and global search [13]. Global optimum is a point presents as the highest fitness whereas local optimum is a point that presents higher than all its neighboring points on the fitness landscape but lower than of that global optimum. Unimodal fitness landscapes contain only exist one locals Optima which are global optimum while multimodal fitness landscapes consist of many local optima which may be the global optimum [14]. Local search algorithms are not able to find global minimum because of insufficient of information to direct search outside of the local minimum.

2.3.1 Deterministic Algorithms and Stochastic Algorithms

In essence, a deterministic algorithm consists of entire input and output relation where the model is conclusively determined whereas a stochastic algorithm generally included some cases with solved randomly [17]. In other words, in a deterministic algorithm normally calculate at each time a certain set of input is presented, such that it does the same computations and continuously gives the same results as any other time the set of input is presented [18].Mathematical methods based on the concept that future behavior can be predicted precisely from the past behavior of a set of data are usually based on the deterministic model. Besides that, in deterministic techniques usually developed by statistical which essentially the average system behavior of an equilibrium or steady-state relationship such as linear regression.

In substance, stochastic can be classified into two types of the group which is heuristics and meta-heuristic algorithm. Heuristics optimization algorithms usually start the process by creating one or set of random solutions and improve them such that obtain high ability to avoid local optima [19]. In the case of the Stochastic algorithm, the components such as random walks have become an intrinsic part of modern metaheuristic algorithms. In addition, the efficiency of a metaheuristic algorithm may essentially rely on the appropriate use of randomization. Nevertheless, it is quite tougher to calculate the convergence rate because of random nature type [15][20]. Normally, a random number is generated by some methods or to meet other requirements to execute trial.

Comparison between deterministic and stochastic optimization algorithm tend to show that deterministic algorithm performs less efficient and cannot practically deliver good solutions compared to stochastic especially with involving moderately high levels of noise. Thus, its demand large scale in reconstruction either through serial or parallel [15].Also, absolutely obvious that randomized algorithms usually provide better performance in the problem applications rather than deterministic algorithms [16].

2.3.2 Swarm-Intelligence Optimisation Algorithms

Depending on the nature of phenomenon simulated, evolutionary computing algorithms can be classified into two important groups which known as evolutionary algorithms (EAs) and swarm intelligence based algorithms. EAs, which mainly obtain based on inspiration from nature, have shown to be very efficient for optimization among all the methods devised by the evolutionary computation community [5].Swarm intelligence is the study of computational systems inspired by the 'collective intelligence' that emerges through the cooperation of colony of agents in the search space for examples include schools of fish, flocks of birds, and colonies of ants. In nature, such systems are commonly used to solve problems such as effective foraging for food, prey evading, or colony relocation [10]. The information memories are typically stored throughout the cooperate agents or communicate in the environment by itself for examples through the use of pheromones in ants, dancing for bees, and proximity to fish and birds.

There are variety types that present swarm intelligence phenomena in the natural world, which can give us countless inspiration to minimize the elapsed time and cost function. For instances, swarm models were inspired by the intelligent behavior of colony members of animals, such that many optimization computation methods have been proposed to solve complex problems that occurred [2] [4]. Besides that, in 1995, particle swarm optimization algorithm (PSO) was proposed by Kennedy which inspired through social behaviors and movement dynamics of birds [5]. In 1996, inspired by social division and foraging behavior of ant colonies, Dorigo M proposed the Ant Colony Optimization algorithm (ACO) [6]. In 2002, inspired by foraging behavior of fish schools, Li Xiaolei proposed the Artificial Fish Swarm Algorithm (AFSA) [7]. Other than that, in 2005, motivated by the specific intelligent behaviors of honey bee swarms, Karaboga proposed the artificial bee colony (ABC) algorithm [8]. Birds, fish, ants and bees do not have any human intelligence such as logical reasoning and synthetic judgment, however, under the same purpose, they emerge as robust swarm intelligence which constantly altersenvironment and mutual cooperation, which provide new ideas and techniques to solve complex application solution.

Initially, ants drift randomly around their communal web. Once the food is located, they will begin laying down pheromone in the environment. Numerous trips between the food and the colony are observed and if the same route is followed that leads to food then additional pheromone is laid down [23][24]. Other ants may recognize the similar path head to the located food and may stick to it and also lay down pheromone. Hence, more positive feedback process routes and more ants converge to valuable paths. In contrast, Bees algorithm illustrates the behavior of a swarm bees searching for a food source [25].

In comparison to the PSO population or swarm, converges on the optimum by using information gained from each individual, referred to as a particle as well as from the information gained by the swarm as a whole. This algorithm starts with an initial population that is randomly distributed among search space. Then, the position of each particle is updated to the next iteration. The solid points that PSO have a stable convergence characteristic and take less time consuming than other methods which make PSO can produce a high-quality result but it suffers from memory capability and computational burden [26].

The SSO has a special structure that able to find the solutions in the search space, so that it make the balance between exploration and exploitation and also, prevents falling into the local optimum. Actually, the communal web is the link that the spiders transform information through it and the information is transferred using the produced vibration by the spider.

In a nutshell, the objective is to exploit historic and heuristic information to construct solutions and fold the information learned from constructing solutions into the history. The compound solution designed with one discrete piece at a time in a probabilistic step-wise manner. Thus, a procedure for selecting process component is intent on by the heuristic contribution of the component to the overall cost of the solution and the quality of solutions at which the component has already stored [9]. History is updated proportional to the quality of the best-known solution and is decreased proportionally to the usage if discrete solution element.

2.3.3 Social Spider Optimisation

Social Spider Optimisation Algorithms (SSO) was proposed for global optimization applications which inspired by the intelligent behavior of the social spiders mainly based on the biological law of mating behavior, utilizing the vibrations on the spider communal web. The foraging behavior of the social spider can be described as the cooperative movement of the spiders towards the food source position in the communal web [3][4]. Social Spider Optimization (SSO) starts iterations to search for an optimal solution in the search space. Basically, the communal web is the link that the spiders transform information through it and the information is transferred using the produced vibration by the spiders.

Firstly, each female checked all the spiders in the search space then start finding for the presence of stronger vibration in the communal web. Since the strong vibrations depend on the spider characteristics which contain the memory of higher fitness and weight [21]. Lets someone more attractive, the Euclidean distances calculated. Next, the shortest distance between the spider's positions is calculated and indexed. An attraction is done based on the strong vibration and distance coming from the nearest spider. Furthermore, repulsion or dislike activity is done depend on the gender. The median of spiders is calculated, the male spiders above mean median is started looking for a female with short distance whereas spiders below the median, go to weighted mean.

During mating process, male spider being checked either it is good or placed above the median in order to generate the new generation similarly or even called offspring. Only the male spiders above the median are mating and the radio (range of mating) is calculated before start looking, whether have a best female nearby. The mating process occurs and produces new offspring’s which next step is evaluated and worst spider on the colony calculated. If the fitness of the offspring is better than the worst spider, so the worst spider removed from the colony. This operation is iterated based on mating and movement occurs till convergence to the optimal solution. This algorithm should be flexible enough to accommodate a variety of shapes and sizes. The computational speed of the algorithm is to be considered to analyze the effectiveness and robustness. The validity of this algorithm is to be analyzed.

The fitness values are evaluated once for each spider at each iteration. Depending on its gender, each spider tends to reproduce a specialized behavior. Since there were different between the global best and current positions of spiders lead to different searching behaviors. In addition, SSO has a strong capability to solving the multi-modal optimization problems with help in reducing premature convergence and local minimal problems [3]. In addition, the social spider algorithm is evaluated by a series of widely used benchmark functions and proposed algorithm has superior performance compared with other state-of-the-art metaheuristics [5].

2.4 Related research work on PID Controller

Referring to various research papers, researchers review multi-objective optimization problems from different viewpoints and thus exists different solution philosophies and goals when setting and solving them. Among the selected research paper which related to optimization of PID controller with utilizing a variety of methods that briefly explain the details thus able to intensify the knowledge of underlying objectives of optimization in a comparative study.

2.5 Chapter Summary

This chapter is to review the past research committed by a researcher in the field of engineering scopes that require optimization implementation in order to access the desired outcomes. Based on designed solution with respect to specific objective function and fitness function, set of procedures are determined in order to evaluate the validity performance of the system such that can obtain optimal results. The algorithm used in this comparative study basically in the range of stochastic algorithm which known as Social Spider Optimisation (SSO). Thus, the concepts and idealization of this algorithm were presented in detail in the next chapter. In addition, certain research papers were also reviewed for further study on fundamental concepts on choosing suitable algorithm and control parameters. Finally, the design problem and cost functions being there as the performance index which decides the performance of the system to be optimized.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this chapter, present the optimization of tuning PID Controller by implementing the Social Spider Optimization (SSO). The control parameters are set in the algorithm and methods proposed to achieve the objectives of this research were explained in details. Section 3.4 presents the techniques used to figure out the performance of algorithm which is explained in detail.

3.2 Optimization Problem Design

PID controllers have been widely used to control many systems [11]. Specific design problems are usually correlated to optimization problems. Different parameter setting and problem dimensions can influence the analysis of result or fitness. The controller design, the procedure proposed, and the controller performance approaches are presented in details.PID controller can be investigated into three categories. Each controller has different properties in term of controlling the whole system. In proportional control, adjustments are based on the current difference between the actual and desired speed. While, integral control, adjustments are based on recent errors. In derivative control, adjustments are based on the rate change of errors.

3.21 System Plant

In this study, the closed-loop response model was created using MATLAB 2015 and executed using Intel (R) Core™ Pentium-4210u CPU@1.7GHz. Thus, both closed-loop PID controller design being analyzes based on the steady state response obtained which is generally the final value achieved by the system output. In a closed-loop system, it is necessary to have of transfer plant function. Hence, refer to research paper reviewed by Pareek et al. in [14], Gp_1 (System A) was chosen for this studied. Thus, the equation below illustrates as transfer plant function. The integrated PID controller proposed in these design, use the equation (3.2) which is in the Laplace form as the transfer function, $G_c(s)$.

Equation (3.1) below represent as a transfer plant functions for System A.

$$\text{Gp1(s)} = \frac{1}{S^4+6S^3+11S^2+6s} \qquad (3.1)$$

Equation (3.2) below represent as a transfer plant functions for System B.

$$\text{Gp2(s)} = \frac{0.01}{0.005S^3+0.06S^2+0.1001s} (3.2)$$

$$\text{Gc(s)} = K_p + \text{K}_\text{d}\text{S} + \frac{Ki}{S} (3.3)$$

The general transfer function, T(s) of the system are as follows:

$$\text{T(s)} = \frac{\text{Gp(s)Gc(s)}}{1+\text{Gp(s)Gc(s)H(s)}} \qquad (3.4)$$

Where H(s) is a unity feedback. Thus, the transfer function of System A is stated as below.

$$T1(s) = \frac{KdS^2 + KpS + Ki}{S^4 + 6S^3 + (11 + Kd)S^2 + (6 + Kp)S + Ki} \quad (3.5)$$

$$T2(s) = \frac{0.01KdS^2 + 0.01KpS + 0.01Ki}{0.005S^3 + (0.06 + 0.01Kd)S^2 + (0.1001 + 0.01Kp)S + 0.01Ki} \quad (3.6)$$

The diagram below shows the block diagram used for this system. This system consists of PID controller, system plant with a step input. The transfer function was coded in MATLAB using suitable format. It is rather a difficult task to randomly select the PID controller parameters and test the combinations of the parameters to ensure an optimum performance. Thus, the algorithm will search for the best parameters so that good transient response can be performed. For this case of study, Social Spider Optimization (SSO) was used to find the best parameters of *Kp, Ki,* and *Kd*. This study mainly focuses on an algorithm which can tune the PID controller better which supposedly towards desirable performance.

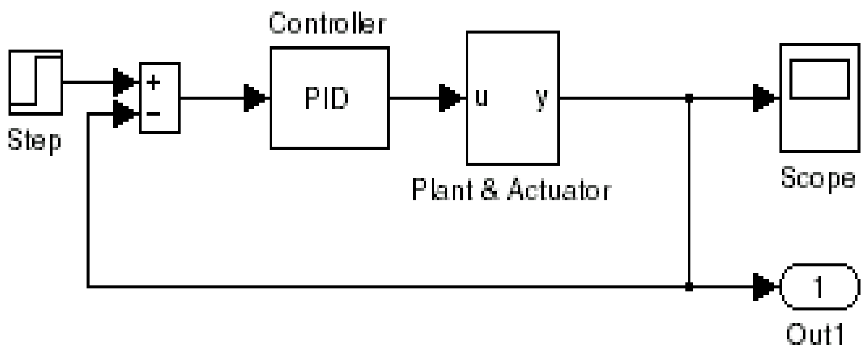


Figure 3.1: Closed-loop system with PID controller

3.2.2 Concept of Objective Function

To obtain the optimum PID controller parameters, design variables were selected to produce minimize the maximum percent overshoot, the rise time, the settling time and the steady-state error of terminal voltage step response. The best parameters were determined by the minimum value evaluated by cost the function, F [28]. The combination of the parameter as stated which is Kp, Ki Kd were represented by a point for which the step response is obtained.

In addition, the performance of the control system is usually evaluated based on its transient response behavior. This response is the reaction when subjecting a control system to inputs or disturbances [29].The response of the controller can be described in terms of the degree to which the system peak overshoot, *Mp,* steady-state error, *ess,* rise time, *Tr* and settling time, *Ts*. When using an optimization algorithm to find the PID gain parameters, such as the Social Spider Optimization (SSO) algorithm, these objective functions are combined in a single weighted sum objective function also known as cost function. Below shows the cost function that chosen because of display good dynamic performance to criteria stated in the function.

$$\mathrm{F} = (1 - e^{-\rho})(\mathrm{Mp} + \mathrm{ess}) + (\mathrm{e}^{-\rho})(Ts - Tr) \qquad (3.7)$$

Where ρ is the scaling factor of designer's choice. Based on review from research paper, according to Zwe-Lee in [12], if the value larger than 0.7 was set to ρ it tend for a lesser overshoot and steady-state error while if we used value smaller than 0.7 it tend to get the shortest rise, Tr and settling time, Ts. Thus, for this research, two choices of values were chosen *i.e.* ρ=0.5 and ρ=1.5

3.3 Social Spider Algorithm

3.3.1 Idealisation of SSO

Social-spider optimiser (SSO) is chosen to fine tune the proposed proportional-integral-derivative (PID) controllers by generating their optimal settings. The integral time multiplied the summation of absolute deviations thus the gains of PID controllers define the fitness function and control variables respectively [27]. Basically, SSO is inspired by mating behavior and movement in nature that occurs till convergence to the optimal solution are met. Using the SSO algorithm where each Spider location is a candidate solution to the Proportional-Integral-Derivative parameters.

The computational procedures for the propose Social Spider Optimization [22].

Step 1: Initialize N as the total number of n-dimensional colony members, define the number of male (Nm) and females sides (Nf) in the entire population (S).

$$Nf = \text{floor}\,[(0.9\text{-}rand*0.25)*N],\; Nm = N\text{-}Nf$$

Where a *rand* is a random number in the range [0,1], whereas floor(.) maps a real number to an integer number.

Step 2: Initialize randomly the female members ($F=\{f_1, f_2,\ldots,F_{Nf}\}$), male members ($M=\{m_1, m_2,\ldots, mNm\}$) and calculate the radius of mating.

$$r = \frac{\sum_{j=1}^{n} \rho j^{high} - \rho j^{low}}{2n}$$

Step 3: Calculate the weight of every spider of S

For (i=1; i<N +1; i++)

$$wi = \frac{J(si) - worsts}{bests - worsts},$$ where $best_s = \max (J(S_k))$ and $worts_s = \min (J(S_k))$

End For

Step 4: Move female spiders according to the female cooperative operator

For (i=1; i<N_f +1; i++)
Calculate $Vibc_i$ and $Vibb_i$
If (r_m <PF), where $r_m \in$ rand (0,1)
$$f^{k+1}_i = m^k_i + \alpha * Vibci * (S_c - f^k_i) + \beta * Vibbi * (S_b - f^k_i) + \delta * (\text{rand} - \frac{1}{2})$$

Else If
$$f^{k+1}_i = m^k_i - \alpha * Vibci * (S_c - f^k_i) - \beta * Vibbi * (S_b - f^k_i) + \delta * (\text{rand} - \frac{1}{2})$$

End If
End For

Step 5: Move the male spiders according to the male cooperative operator

Find the median male individual (w_{Nf}+ m) from M.

For (i = 1; i< Nm + 1; i++)
Calculate *Vibfi*
If (wNf + i> wNf + m)
$$m^{k+1}_i = m^k_i + \alpha * Vibfi * (S_f - m^k_i + \delta * (\text{rand} - \frac{1}{2})$$
Else If
$$m^{k+1}_i = m^k_i + \alpha * (\frac{\sum_{h=1}^{Nm} m^k_i * wNf+h}{\sum_{h=1}^{Nm} wNf+h} - m^k_i)$$
End If
End For

Step 6: Perform the mating operation

For (i = 1; i< Nm + 1; i++)
If ($m_i \in$ D)
Find E^i
If (Ei is not empty)
From S_{new} using the roulette method
If ($w_{new} > w_{wo}$)
$S_{wo} = S_{new}$
End If
End if
End if
End For

Step 7: If the stop criteria are met, the process is finished; otherwise, go back to Step 3.

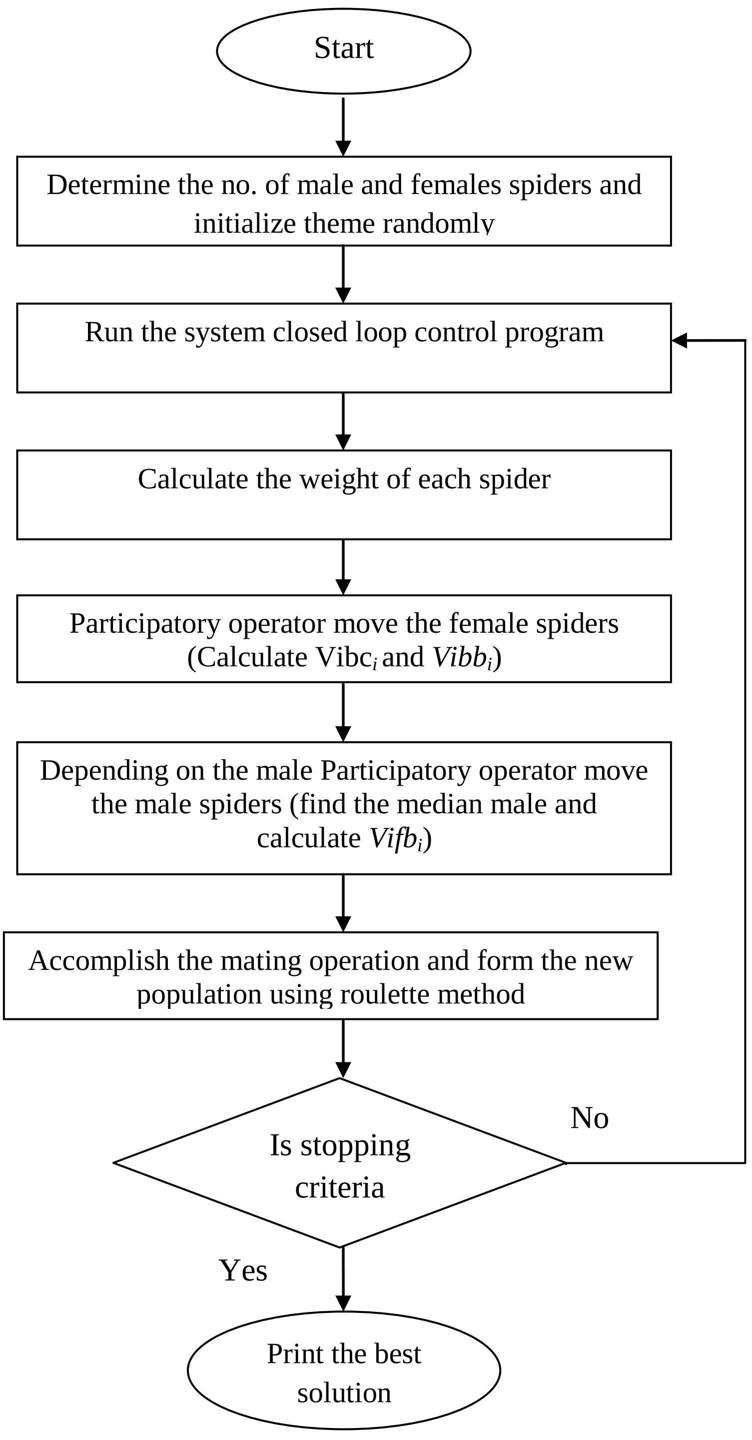


Figure 3.2: Flow chart of Social Spider Optimization (SSO)

Based on the flow chart illustrated in Figure 3.2, SSO works initially with generating the initial population of spiders, spidn including two agents, which is females, f_{ps}, males, m_{ap}. Next, initialize randomly the females and males member and calculate the radius of mating. Calculate the weight if each spider in the entire population. Move female spiders according to the female cooperative operator to emulate the cooperative behavior female spider, considers the position change of the female spider (i) at each iteration process. Whereas male move according to the male cooperative operator with the nearest female individual to male member (i). Next, the mating operator usually occurs between dominant male and female. The iterative process updates the best solution continuously by evaluating the fitness function. Next, the fitness spider was ranked based on the fitness function with resulting in current best quality solution and fitness value. This process continues till the termination criterion is achieved such as reaching the maximum number of iterations.

3.3.2 Setting of SSO

The algorithm owns specialised control parameter that can be set based on the optimization problem in order to achieve the cost and objective function. The control parameters are listed in Table 1. Based on this research, the set number of agents, spidn, 50 while the number of iterations, itern,200. Since, in this study used 3 different parameters, thus dims represent as the decisions variables respectively whereas variables are limited by the parameters xd and xu lower bound and upper bound of decisions variables or in other words, represent as the lower and upper bounds of Kp, Ki,K_d parameters respectively.

Descriptions	Parameters	Initial Value
No. of Spider	spidn	50
No. of Iterations	itern	200
Lower space dimension	xd	0
Upper space dimension	xu	10
Lower Female percent	fpl	0.65
Upper Female percent	fpu	0.9
Dimension space	dims	3
Attraction or dislike	PF	0.7
Random number	rand	[0,1]

Table 3.1: Sets of control parameters

The objective function in this system was saved as *trackls.m* file. Then, it will directly interface with the m. file of Social Spider Optimization (SSO). Hence, the variables can directly read by the algorithm.

3.4 Comparative study on cost functions and step response of PID controller

After all the best set of control parameters was selected, the m.file was executed to extract data of cost values and parameters of Kp, Ki and Kd. The number of runtimes or *runtime,* also included as one of stopping criteria in this algorithm. In this research, 20 runtimes were set. Each runtime is an independent experiment which does not affect the other experiment. Runtime which returned lowest cost value with the best optimal solutions was chosen from all the runtimes.

Apart from that, 20 runtimes were set, so that to check the robustness of SSO. Each runtime will run for 200 iterations as we set number of iteration, itern is 200. The last iteration usually returns least cost value of the objective function. The data of cost values and solutions obtained were saved in *.csv* file format for analysis purposes. The last cost function value obtained for each iteration and runtime was compared to get least cost value. The minimize cost value obtained was chosen and the combination of the PID controller parameters were tested.

Hence, the step response produced by the combination from this research was analyzed to discover the performance either closely as step input or vice versa. The combination of parameters continues for second smallest cost value if the first one does not produce desirable output.

Referring to cost function in equation 3.7 (refer page17), there are four important criteria that correspond to cost value evaluated. The smaller the value obtain, the better the solution produced. From the first iteration, the best cost value was kept and transferred to next iteration for comparison. The solutions were evaluated and continued until the termination conditions achieved. To calculate the average cost value, the cost values from *runtime* 1 to runtime 20 will be added, next divided with the total number of *runtime,* 20.

In addition, the solutions improved in each iterations as the cost values were significantly reduced through iterations and lastly converged toward final best value. It will be continued until optimal solution was found. An average cost function was plotted against iterations in order to analyze the convergence of cost values. In order to analyze the trend of variance between the data, standard deviation versus iterations was plotted. Other than that, high standard deviation shows that the data has alarge variance from the average value while low standard deviation shows that the data obtain asmall variance from the average value.

Finally, errors bars were plotted to indicate the variability of data. The performance of SSO is studied in terms of computational time, cost function values and convergence. The step response wasanalyzed based on the data obtained for the combination of parameters for this algorithm.

3.5 Chapter Summary

In Chapter 3, represent the methodology on how to achieve the objective of this research. Existing metaheuristic tuning methods have been proven to be quite successful but there were observable areas that need improvements especially in terms of the system's gain overshoot and steady state errors. In short, the objective function consists of four criteria which are overshoot, $Mp_{,}$ settling time, Ts, rise time, Tr and steady-state error,e_{ss} which returns lowest cost value to specify the best parameters. The set of control parameters were selected from several research papers and information are listed in detail on the methods used. It contains two different system plants which are set as System A and System B. Finally, analyses were made based on the dynamic performance of each system.

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

This section presents the results obtained from this research project using the method as stated before. The analysis made based on average, the standard deviation of cost values and transient response of the system.

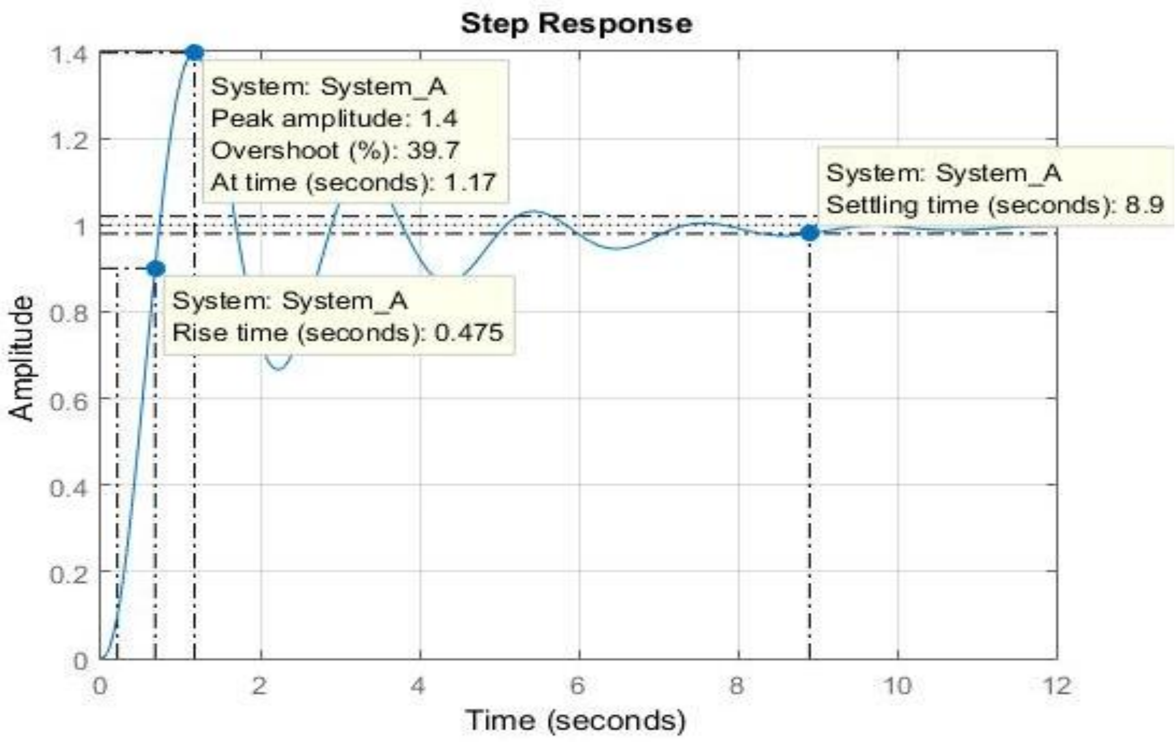
4.2 Closed-loop System

4.2.1 Transient response of the system without algorithm

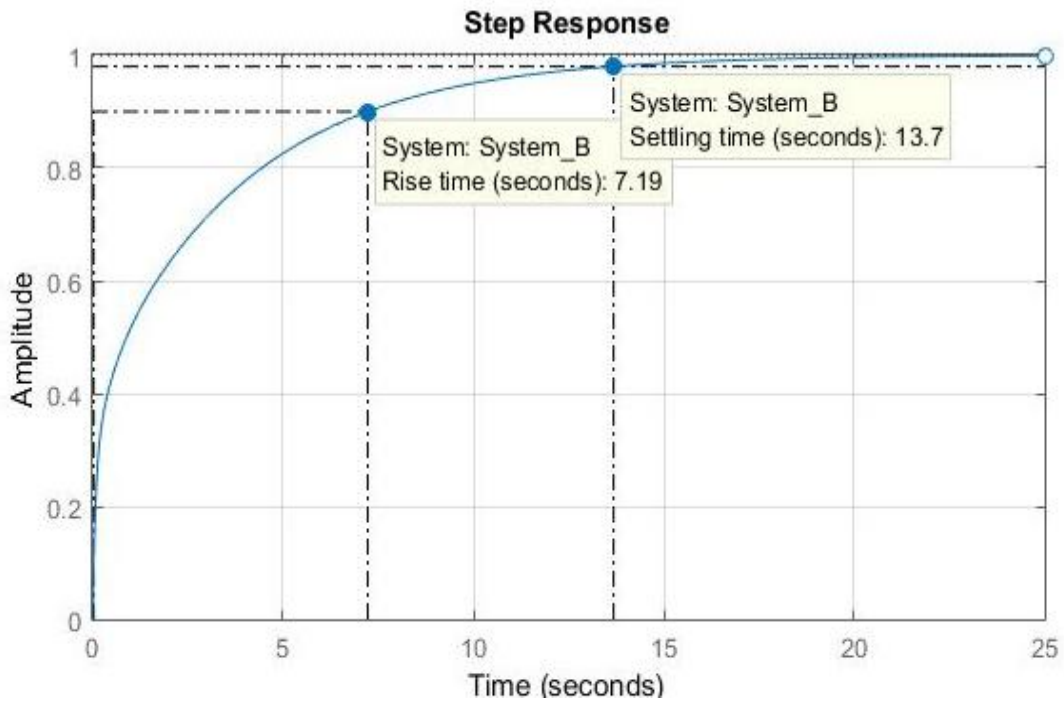
The systems used to test this algorithm were Gp_1: System A (equation 3.1) and $Gp_{2:}$ System B (equation 3.2) consists of step input, PID controller however without an algorithm to tune the parameters. The values of Kp, Ki and Kd parameters were selected randomly for analysis purpose (table 4.1). The step responses were obtained (refer Figure 4.1).The dynamic performance specifications of the system are tabulated in the Table 4.1 which include the values of percentage of overshoot, Mp, rise time, Tr, settling time, Ts and steady-state error, e_{ss}.

Table 4.1: Values of Kp, Ki and Kd parameters

Parameters	System A	System B
Kp	40	8
Ki	10	4
Kd	4	2



(a)



(b)

Figure 4.1: Step response of (a) System A and (b) System B without algorithm