SIMULATION OF IDENTIFYING SHORTEST PATH WALKWAY USING PARTICLE SWARM OPTIMIZATION (PSO)

YONG KAE SHIN

UNIVERSITI MALAYSIA PAHANG

SIMULATION OF IDENTIFYING SHORTEST PATH WALKWAY USING PARTICLE SWARM OPTIMIZATION (PSO)

YONG KAE SHIN

Report submitted in partial fulfillment of the requirements
for the award of the degree of
Bachelor of Computer Systems & Software Engineering
(Graphic & Multimedia Technology) with honors

Faculty of Computer Systems & Software Engineering
UNIVERSITY MALAYSIA PAHANG

JUNE 2012

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this project and in my opinion, this project is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Systems and Software Engineering (Graphics & Multimedia Technology) with honors.

Signature	:	
Name of Supervisor	:	ZALILI BINTI MUSA
Position	:	LECTURER
Date	:	10 JUN 2012

STUDENT'S DECLARATION

I hereby declare that the work in this project is my own except for quotations and summaries which have been duly acknowledged. The project has not been accepted for any degree and is not concurrently submitted for award of other degree.

Signature	:	
Name	:	YONG KAE SHIN
ID Number	:	CD09024
Date	:	10 JUN 2012

ACKNOWLEDGEMENTS

I am grateful and would like to forward my appreciation to my supervisor, Miss Zalili binti Musa for her guidance and constant support that helping me a lot in this project. She have give me a lot of encouragement while doing this project and also have instill me the knowledge that required for this project. I appreciate her hard work to guide me from the first day I approach to her to have her as my supervisor to these concluding moments. I am truly grateful for her progressive vision about my training in writing good documentation and she often impressed me with the idea on how to design the system flow. I often make naïve mistakes but she able to tolerance with me and lead me to back to the correct way.

My sincere thanks go to the lecturers who help me directly or indirectly during the period to complete this project. I would like to express a special thanks to the evaluators of my project for their suggestions and opinions.

ABSTRACT

The purpose of this research is to present the development of the system of Simulation of Identifying Shortest Path Walkway using Particle Swarm Optimization (PSO). This application system is use to solve the shortest path problem in the simulated environment. The suggested simulation environment is a restaurant. In the real world restaurant, if the waiter unable to deliver the food within short period of time, this may reduce the customer satisfaction and downgrade the work performance. This will not only happen in the restaurant, since people nowadays prefer the effective services. Therefore, this project is develop to find the shortest path which means to find a path with minimum distance between two locations by apply the algorithm of Particle Swarm Optimization technique. This system is expected to solve the shortest path and identify the optimal path. Due to this, it is tend to reduce the time consume in specific situation with the determining of shortest path. The development of this system could bring convenience and effectiveness for the work in progress.

ABSTRAK

Tujuan penyelidikan ini adalah untuk membentangkan pembangunan bagi sistem Simulasi Mengenalpasti Laluan Terpendek dengan menggunakan Particle Swarm Optimization (PSO). System aplikasi ini adalah digunakan untuk menyelesaikan masalah laluan terpendek dalam persekitaran simulasi. Kawasan persekitaran simulasi yang dicadangkan adalah restoran. Di restoran dunia sebenar, jika pelayan tidak dapat untuk menghantar makanan dalam tempoh masa yang singkat, ini akan menyebabkan ketidakpuasan pelanggan dan juga menurunkan taraf prestasi kerja. Keadaan ini bukan hanya akan berlaku di restoran, ini disebabkan orang pada masa kini suka perkhimatan yang berkesan. Oleh demikian, projek ini dibangunkan untuk mendapatkan laluan terpendek, dengan maksud lain ialah untuk mencari laluan yang berjarak minimum di antara dua lokasi dengan menggunakan algoritma teknik Particle Swarm Optimization (PSO). Sistem ini dijangka untuk menyelesaikan laluan terpendek dan mengenal pasti laluan yang optimum. Dengan ini, ia cenderung untuk mengurangkan masa yang digunakan dalam situasi tertentu dengan menentukan laluan terpendek. Pembangunan sistem ini boleh membawa kemudahan dan keberkesanan untuk kerja-kerja dalam pemprosesan.

TABLE OF CONTENTS

		PAGE
SUPERVISOR'S DECLARATION		**i**
STUDENT'S DECLARATION		**ii**
ACKNOWLEDGEMENTS		**iii**
ABSTRACT		**iv**
ABSTRAK		**v**
TABLE OF CONTENTS		**vi**
LIST OF TABLES		**ix**
LIST OF FIGURES		**x**
LIST OF APPENDIX		**xi**
CHAPTER 1	**INTRODUCTION**	
1.1	Introduction	1
1.2	Problem Statement	3
1.3	Objectives	4
1.4	Scope of the study	4
1.5	Thesis Organization	5
CHAPTER 2	**LITERATURE REVIEW**	
2.1	Introduction	6
2.2	Shortest Path Problem	6
	2.2.1 Type of Shortest Path Problem	7
	2.2.1.1 Robot Motion Planning	7
	2.2.1.2 Scheduling	8
	2.2.1.3 Travelling Salesman Problem	9
	2.2.2 Algorithm for related work	9
	2.2.2.1 Path planning of Inspection Robot based on	9

Ant Colony Optimization Algorithm
2.2.2.2 Timetable Scheduling Problem by using Genetic Algorithm 12
2.2.2.3 Solving Travelling Salesman Problem by using Genetic Algorithm 14
2.3 Particle Swarm Optimization (PSO) 16
2.4 Related work for Particle Swarm Optimization (PSO) 18
2.4.1 Application of Particle Swarm Optimization for Robot Path Planning 18
2.4.2 Application of Particle Swarm Optimization for Timetable Scheduling Problem 20
2.4.3 Traveling Salesman Problem using Particle Swarm Optimization (PSO) 23
2.5 Conclusion 25

CHAPTER 3 METHODOLOGY

3.1 Introduction 26
3.2 Project Methodology 26
3.3 Planning 27
3.3.1 Consultation with Supervisor 28
3.3.2 Planning the Project Milestone 28
3.4 Requirement 29
3.4.1 Requirement and scope derivation 29
3.4.2 Hardware and Software Requirements 29
3.4.2.1 Hardware specification 30
3.4.2.2 Software specification 30
3.5 Analysis 30
3.5.1 Search for relevant information 31
3.5.2 Define types of shortest path and techniques use to solve 31
3.5.3 Analysis on Implementation Technique 32
3.5.4 Interview with restaurant’s customers 32
3.5.4.1 Constraints 33
3.6 Design 34

3.6.1 Flow of system 34
3.6.1.1 Specify the number of table 35
3.6.1.2 Calculate distance 35
3.6.1.3 Choose table as checkpoint 39
3.6.1.4 Search for shortest path 39
3.6.1.4.1 Definition of fitness function 39
3.6.1.4.2 Particle Swarm Optimization (PSO) 40
3.6.1.5 Reset 41

CHAPTER 4 IMPLEMENTATION AND TESTING

4.1 Introduction 42
4.2 Implementation 42
4.2.1 Heuristic Approach 42
4.2.2 Particle Swarm Optimization 44
4.3 Testing 46
4.3.1 Black-box Testing 46
4.3.2 Alpha Testing 47

CHAPTER 5 CONCLUSION

5.1 Introduction 48
5.2 Conclusion of Chapter 1 48
5.3 Conclusion of Chapter 2 48
5.4 Conclusion of Chapter 3 49
5.5 Conclusion of Chapter 4 49
5.6 Overall Conclusion 49
5.7 Limitation 50
5.8 Future Research 51

REFERENCES 52

APPENDICES

A Gantt Chart 53

LIST OF TABLES

TABLE NO.	TITLE	PAGE
3.1	Hardware Specification	30
3.2	Software Specification	30
3.3	Arrangement of tables	33

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE
2.1	The avoidance obstacle path of robot.	10
2.2	The distribution of obstacles and checkpoints.	11
2.3	The result of path planning.	11
2.4	Uniform crossover of individuals.	13
2.5	Mutation of individual.	13
2.6	Flow chart of the steps of Genetic Algorithm	14
2.7	Basic PSO Algorithm	17
2.8	Standard situation in robotic soccer solve by different approach	19
2.9	Project overview and planning of timetabling scheduling	21
2.10	Flow Chart for assigning the timetable.	22
2.11	The data assign in timetable	22
2.12	Graph of iteration in PSO	22
2.13	DPSO algorithm for GTSP	23
2.14	Pseudocode of PSO for discrete optimization problem	24
2.15	Continuous-variable optimization problem	24
2.16	Outline of the proposed method	24
3.1	Flow of Development Model	27
3.2	Flow chart of the system	35
3.3	Flow chart of Heuristic approach	36
3.4	Arrangement of counter and table	37
3.5	The Heuristic approach	38
3.6	Flow chart of basic PSO Algorithm	40
4.1	Coding for Heuristic Approach	43
4.2	Coding for show the path and number of element for the path	44
4.3	Coding for update velocity and position of path	45
4.4	Potential paths	46
4.5	Number of element for each potential paths	47
4.6	Shortest path	47

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	Gantt Chart	53 - 54

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Generally, there is already a lot of effort that has been done for an optimization for certain purpose. The studies that based on optimization are implementing widely and it is become a common field to explore. Optimization is a process or methodology that to making something becomes functional, effective and fully perfects and it is normally involve in mathematical procedures. There are quite a lot of applications that use optimization to achieve their purpose. For instance, it involves the sector of science, economics, engineering and industry. Optimization may also require some extra technique to implement it since at the last, the application is need to implement into a system.

Optimizations are concern on the findings of highest achievable performance under several constraints for various aspects. The method of optimization is applied to wide variety of problems from scheduling to path planning. For example, the staff scheduler of hospital, there is a finite set of staff members that available for each unit in the hospital. Therefore, in order to arrange the schedule, some regulations and rules also need to follow such as the require numbers of the staff for the rest days in a schedule. The development of this system is able to facilitate the work and management for some certain sectors.

Besides that, optimization also widely uses in various sectors such as optimize energy usage in industry sector. This optimization helps in control the

energy usage in the earth. Through a better utilization of energy resources, it is giving benefits to peoples and our planet.

Therefore, optimization is an important method for measurement as to provide correct outcome. This is because the results that obtain from optimization may help in some events that are hard for human. For example, human is hard to measure or identify something correctly and objectively by just depends on their sense or simple estimation. This may cause to the happening of some mistaken or failure in some cases. In contrast, by using optimization with a certain technique can provide accurate solutions to solve the problems.

With the development of study in optimization, its show that optimization is important as it contains wide variety of techniques such as artificial intelligence and computer science that use it to improve certain process in many industry. Due to this, a system that uses to identify an optimum path is necessary. This is because the development of this system may bring convenient and effectiveness for the work in progress.

One of the subsidiaries in optimization is to find the shortest path walkway. As known, a shortest path problem is to find a path with minimum distance between two locations. It is also can be say as to find the quickest way to move from one location to another location. This is important to searching for the most optimize path for certain purpose and also to improve the effectiveness. The shortest path problems also cover from several different aspects such as robot motion planning, scheduling, travelling salesman problem, job-shop scheduling problem, vehicle routing problem and so on. There are also some existing application or technique that had been used to solve this shortest path problem, for example Dijkstra algorithm, Bellman-Ford algorithm, Ant Colony(ACO), Genetic Algorithm(GA) and so on. However, the techniques that will be use in this study are Particle Swarm Optimization (PSO) which is in Artificial Intelligence (AI) field.

Therefore in this project, a system is developed by using optimization method to identify the shortest path walkway in a simulated environment. Optimization is important for searching the shortest path walkway in this project since this project

will involves huge number of calculation for each of the path distances. The technique to use in this project, Particle Swarm Optimization (PSO) is the optimization technique that develops by Dr. Eberhert and Dr. Kennedy in 1995 and it is inspired from simulation of social behavior which is fish schooling and bird flocking. PSO optimizes a problem and search an optimal value through iterations and PSO can approach the optimal solutions with fast speed. As a result, the development of this system can identify shortest path walkway easily in order to reach destinations between two places with less time consume.

1.2 PROBLEM STATEMENT

Nowadays, peoples are prefer to have the quick and fast services because in this busy society, everyone is compete with each other and are try to avoid the delaying on time. Therefore, the speed of a service is very important to fulfill the requirement of the society. Due to this reason, the service of a restaurant is important as food is the general needs of human. Peoples that order food will hopes that it can be delivering faster in order to save their time. Besides that, customer services are important to let the customer have a good impression on the restaurant. Based on this situation, it's motivated the needs on identify the shortest path walkway and use by the waiter (mobile robot).

Mostly, the customers are preferred to have the top services which also means quick and accurate because they may not like to wait for a long time. This is because, the higher satisfaction of the customers, the best reputation can remain for the restaurant.

In addition, the waiter may unable to identify correctly that which path is more suitable to use for food delivering when there are several paths. Sometimes the waiter will take the long way to reach to the customer, and this will result in the time delaying. Therefore, shortest path should be implementing to the real restaurant especially the large scale café, so that waiter can move faster and can keep the satisfaction of the customer, even increase.

In some large scale restaurant or café, there are allocated with many tables. In this situation, two table of customer that willing to make order at the same time will be happen. Waiter has to determine which table should be go first, so that can manage the time well to save the time consuming and also maintain a good work performance. Other than determine the sequence of the tables, the shortest path also need to be concern. By having these two concepts, the satisfaction of the customer can be maintained in order to keep the reputation as well.

Besides that, the men power can be replace with robot by applying the application to search the shortest path. By using mobile robot, it can solve some problem such as saving human resources since mobile robot able to work for long period of time, solve the problem of lack of staff and save since no need to pay monthly salary to the mobile robot. However, there are also some difficulties for the small scale restaurant to using mobile robot. This is because after they are paid for a mobile robot, the robot needs to recharge for maintain their batteries life. Hence, the electricity fees also need to be considered.

1.3 OBJECTIVE

The objectives of this project are:

1. To study Particle Swarm Optimization (PSO) technique in simulation of identifying the shortest path walkway for a simulated environment.
2. To develop a shortest path walkway by using Particle Swarm Optimization (PSO) for a simulated environment.

1.4 SCOPE OF THE STUDY

The scopes of this project are:

1. The simulated environment with minimum 9 tables and maximum 25 tables.
2. This project using Microsoft Visual Studio for programming and simulation.
3. The C programming language is use to implement this system with Particle Swarm Optimization (PSO) concept.

1.5 THESIS ORGANIZATION

There are six chapters that included in this thesis and there are different issues to be discussed for each of the chapter. Below is the summary of content for each chapter.

a) Chapter 1 – Introduction
   - This chapter explains about the introduction of this project along with the problem statement, objectives and scope.

b) Chapter 2 – Literature Review
   - This chapter will presented and discuss the related research paper and literature review.

c) Chapter 3 – Methodology
   - Project analysis, design and methodology of the project development will be discussed in this chapter.

d) Chapter 4 – Implementation
   - This chapter explained about the implementation of the system using Particle Swarm Optimization (PSO).

e) Chapter 5 – Result and Discussion
   - This chapter will presented the testing result of the system and discussion on the result.

f) Chapter 6 – Conclusion
   - This chapter explains about the complete summary of the project.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

In this project, PSO is the technique to be use to solve the shortest path problem which it is defined as a computational method that uses to optimize shortest path problem by iteratively and approach the optimal solutions with fast speed [1]. In this chapter, the shortest path problem will be discuss by providing the example of the types of shortest path problem and the algorithm for related work which is done by using another technique such as Genetic Algorithm and Ant Colony. Besides that, the details about PSO also discuss in this chapter by providing the related work of PSO which solve the shortest path problem.

2.2 SHORTEST PATH PROBLEM

Shortest path problem is defined as a problem that to find a path with minimum distance between two locations. It is also can be say that to find the quickest way to move from one location to another [2]. The purpose that to search for a shortest path, basically is to save time consume in order to achieve some improvement. The shortest path problem also cover from various aspect such as robot motion planning, scheduling and travelling salesman problem which will be explain in the next section. In addition, the shortest path problem can be solving by various different techniques such as Dijkstra algorithm, simulated annealing, Bellman-Ford algorithm, Ant Colony and Genetic Algorithm.

2.2.1 Type of Shortest Path Problem

In this section, the different types of shortest path problem which commonly research by the other researcher will be explained and introduced. There are three popular types of shortest path problem which are Robot Path Planning, Scheduling and Travelling Salesman Problem. The technique which use to solve these three problem will also mention is this section.

2.2.1.1 Robot Motion Planning

Mobile robot has widely use in various sectors due to the development of technology. Robot path planning is one of the tasks for mobile robot [4]. The key factor for a mobile robot to perform the task well is definitely depending on smooth navigation of the robot itself. Due to this, the algorithm for the path planning is necessary since this is a task that to generate a safety path for the beginning node to the final destination which with unknown environment and need to consider on the obstacle avoidance [3].

The obstacle avoidance is also one of the important factor that will be consider in the mobile robot path planning in order to achieve some controlled objectives. The obstacle avoidance is along the way to find the shortest path from an actual position *S* of the controlled mobile robot to the specified goal destination *G*, which concerning with position and shapes of the obstacles *O*. All of this parameter will be the input of the algorithm [4].

Besides that, there are two main components that must be concern in the algorithm of the robot path planning which the first is time and the second is must make sure that the path or the distance is safe [4]. This is because robot will play the role as similar to human since the exits of robot mostly are use to replace the man power in order to reduce the use of human energy, so it is important that to ensure the robot is accord to the requirements.

There is no suspicion to view the path planning as optimization problem since the method that use to find optimal path for a robot to move from one point to

another include on some intelligent approach or computational methods such as Genetic Algorithm, Particle Swarm Optimization [3, 4], Ant Colony [5], neural network and so on.

2.2.1.2 Scheduling

Scheduling is fundamentally related with the problem of deciding on how to commit resources between a variety of possible task and the time is the aspect that needs to be considered. The common scheduling problems that have seen in previous research are flow-shop scheduling [6], job-shop scheduling [7, 8], timetable scheduling [9] and so on. In flow-shop scheduling problem, the processing routes are the same for all jobs and it is to optimize a certain objective by allocating the resources to tasks [8]. Besides that, job-shop and flow-shop scheduling is had been proved that is a NP-hard problem [6, 7, 8].

As state in document [6], the flow-shop system will consists of a set of n different jobs which the jobs need to be processes by a set of m machines. This means that each of the jobs may comprise of m operations while each of the operation is performed on different machine for a number of time. Moreover, only one job handles by the machine in a moment. During the process, each job in m operations must processed on machine j (i=1,…,m). Every job can only processed on at most one machine and every machine can only processed at most one job at a time. Besides that, if the job is complete on machine j-1, one job can start on a machine j and if the machine is free. By this way, the job must be completed without interruption once it had started [6].

For the timetable scheduling problem, there are mostly focus on the school or university for academic purpose such as course timetabling and examination timetable [9]. This types of timetabling problem usually concern with the class scheduling and instructor distributions under some constraints. Most of the academic timetables have the complex relationship between period of time, classroom and subjects. The objective for scheduling is same even for job-shop or academic timetabling problem since they are aim to generate an optimal schedule to complete the tasks by utilize all the given resources within a minimum use of time [7].

2.2.1.3 Traveling Salesman Problem (TSP)

Traveling salesman problem is a well known NP-hard combinatorial optimization problem [14] and it is also defined as a task which to find a shortest possible journey that visits each cities or points exactly once. This is done by given a list of the cities and their pair wise distances, for instance the distances between any pair of *n* point [11, 15]. Besides that TSP is also a well known problem for find an optimal path which can be solve by various methods [13] and it is also studied by many other approaches such as Genetic Algorithm (GA) [13], Ant Colony (ACO), Simulated annealing [13], Particle Swarm Optimization (PSO) [10, 12, 14] and the others.

2.2.2 Algorithm for related work

The techniques which use to solve the different type of shortest path problem which introduced in the previous section will explain detail in this section. This section will show the different algorithm such as Ant Colony and Genetic Algorithm according to some research paper on their effectiveness on solving this shortest path problem.

2.2.2.1 Path Planning of Inspection Robot Based on Ant Colony Optimization Algorithm

Ant Colony Optimization is an idea which inspired by the observation of the real ant colonies [17] and it is also a kind of evolutionary algorithm which is based on the simulation of ants [5]. This is defined as the process of ant searching the shortest path for food and return to their nest. During this process, each of the ants will have different path and the route that passing by the ants will remain a substances named pheromone. Due to this, ants will able to discover the shortest path by following the path with strong pheromone concentrations. Moreover, the shortest path that found by the ants can reduce the time use to return the nest [5, 17].

Due to the development of robot technology, mobile robot has been widely using in various sectors nowadays especially the production of factory. In this

development, the key factor is path planning for the robot in the application and this is aim to find the optimal path for the robot to reach the goal destination form the start point. Besides that, the robot needs to be able to avoid the collision with the obstacles in order to reach the destination safely.

There are studies which implement path planning to a mobile robot, this path planning problem is applied in Ant Colony technique [5]. The research has clearly show that the necessary step of implementing the application which the algorithm is the most important. In the very first step, grid is made up by the suboptimal path to avoid the robot to turn frequently when moving on it. The unnecessary grids will then deletes to make the path straighten as to obtain the real solution [5]. In addition, the grid is made for avoid the obstacles and to search for the shortest path way.

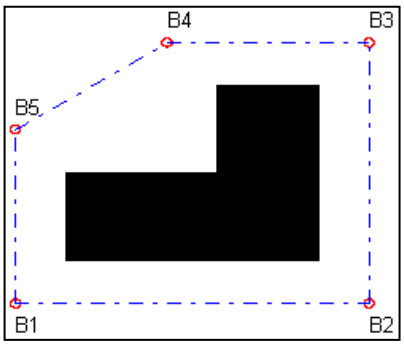


Figure 2.1: The avoidance obstacle path of robot

As seen in Figure 1, B1, B2, B3, B4 and B5 are the nodes which surrounded the obstacles and connect the line that robot will move along. There are also some limit length with the distances between the nodes and obstacles.

According to the research, a list has been constructed during the time the ants transfer between the two nodes repeatedly [5].

i. Look for the practical nodes from point T (starting point) and then transfer to the checkpoint P (target checkpoint). If there are no obstacles between them, proceed to the next step.

ii. Calculate the transfer probability between every two checkpoints with the following equations:

$$\tau_{i,j}(t+a) = \rho\tau_{i,j}(t) + \sum_{k=1}^{m} \Delta\tau_{i,j}^{k}$$

$$\eta_{i,j}(t) = \frac{1}{d_{i,j} + hd_{j,P}}$$

Choose the feasible path with this equation:

$$j = \begin{cases} \arg\max\{\tau_{i,j}^{\alpha}\eta_{i,j}^{\beta}\}, q \le q_o \\ S, otherwise \end{cases}$$

iii. Renew the local pheromone which determined by:

$$\tau_{i,j}(t+h) = \rho\tau_{i,j}(t) + \Delta\tau_{i,j}^{k}$$

$$\Delta\tau_{i,j}^{k} = \frac{Q}{L_{i,j}}$$

Where L denotes the length between point i and j.

iv. Lastly, store the nodes chosen and set the starting point T to a new node. Then, turn to step 1.

The simulation is getting in Matlab where the black components represent the obstacles and the small circuits are checkpoint which the robot will pass through.

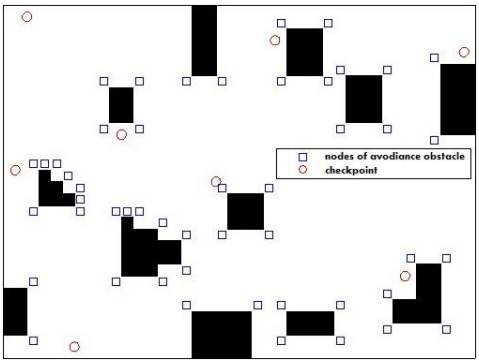


Figure 2.2: The distribution of obstacles and checkpoints

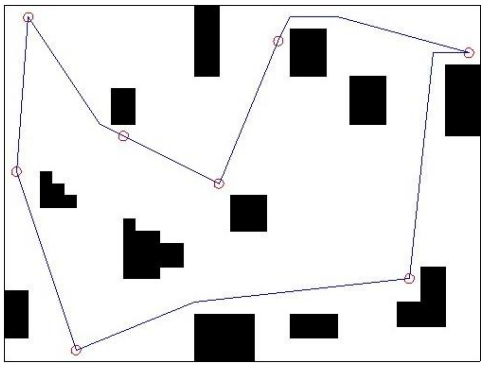


Figure 2.3: The result of path planning.

According to this research, it is successful to apply the ant Colony Algorithm optimization to path planning of inspection robot. Due to this method, the robot is possible to search for the optimal path and passes through all the checkpoints. Therefore, the results show that this method has obtained the optimal path and path planning for robot.

2.2.2.2 Timetable Scheduling Problem by using Genetic Algorithm

Genetic algorithm (GA) is introduced by John Holland in the early of 1970s. Genetic algorithm is inspired by natural evolution and this technique can be use for solving complex problems and search for large problem spaces [20]. For example, the NP problems which stand for nondeterministic polynomial. Besides that, it is working with the set of potential solution which is called population. Each of the individual in Genetic algorithm will measure by fitness function which the fitness value represents the quality of an individual. This is important for Genetic algorithm to select the best genetic material of individuals to produce new individual for further generations. Genetic operators are selection, crossover and mutation which are to maintain the process of evolution [20].

Timetable scheduling problem is common in education field which it presents a set of tasks and a set of resources. In the study, classes represent the tasks and rooms, groups and instructors represent resources. For the general situation, the resources are usually limited and also at the same time there should not have two tasks to occupy one certain resource [20]. The main purpose is to assign the tasks to the resources with no multiple tasks allocated to the resource at the same time.

There are some conditions that set in this study:

a) Group g can attend only one class at one time.
b) Instructor i can teach only one class at one time.
c) Room r only can have one taught at one time.
d) All lectures should be kept exactly once.

Since one class can may use more than one classroom, so the variable generation is using the pseudocode:

```
for each class c {
    generate all possible room combinations
    for each possible (day,time) pair {
        for each r in room combinations {
            create variable x_cdtr
        }
    }
}
```

Genetic algorithm is used in this study in order to achieve a feasible timetable since feasibility function will try to reduce the number of conflicts. The number of conflict will show the constraint which obstruct with the individual [20]. By using Genetic Algorithm, the individual is sorted in ascending sequence since the best individual having smallest fitness value.

In selection, the bad individuals will be eliminates form the population in order to have the better population for next generation. Then, the crossover operates on the individual which act as *parents* to make the new individual which is called *child* in order to fill up the empty place in population after the elimination. In this study, uniform crossover is chosen. Mutation acts to take one or more genes from an individual and changes its value [20].

```
for each gene in (parent1 , parent2){
   if(parent1[gene]==parent2[gene]){
      child[gene]=parent1[gene];
   }else{
      child[gene] =
      random(parent1 ,parent2)[gene];
   }
}
```

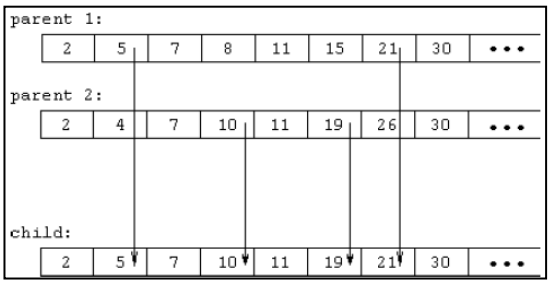


Figure 2.4: Uniform crossover of individuals

```
for each gene in individual{
   if(p(Random) < pm){
      gene = get random value from
             possible values list;
   }
}
```

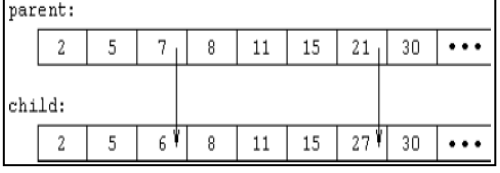


Figure 2.5: Mutation of individual.

The results of this study show that the small problem have solve without conflicts but when solving the large size problem, the algorithm reached about 20 conflicts. It is show that the algorithm needs further improvement to solve the full scale problem completely.

2.2.2.3 Solving Travelling Salesman Problem by using Genetic Algorithm

In this study, TSP is apply by the idea of Genetic Algorithm which in first to find the different solution and then combine the solutions. The solutions found are the fittest solution in between the population and this is to create a new and healthy solution [13]. TSP is a problem which to find the optimal path and in the study, it performs by the basic steps of Genetic Algorithm which are evaluation, crossover and mutation.

The steps of algorithm are as shown in below [13]:

a) Create the initial population randomly and create the matrix representation.

b) Assign a fitness of each individual in the population using criteria measure:

$$F(t) = \frac{value\ fo\ the\ assignment\ of\ the\ given\ problem}{value\ of\ the\ string}$$

c) Create new off-spring population from the existing string in parent population and apply crossover operation.

d) Apply mutation operation to the resultant offspring if required.
(Offspring population has the higher fitness than the parent after crossover and mutation.)

e) Call new offspring as parent population and repeat steps (c) and (d) until the optimal solution to the problem is achieve.

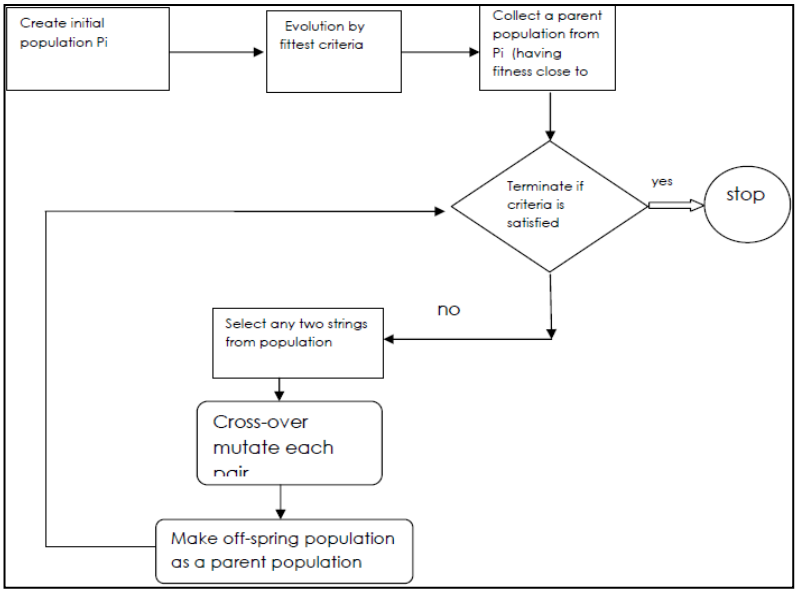
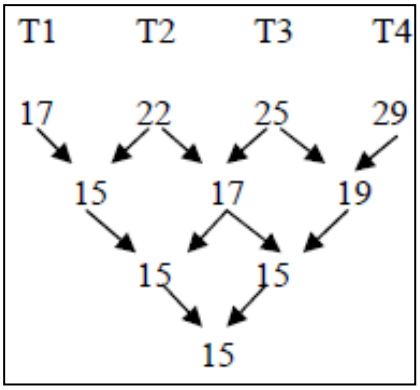


Figure 2.6: Flow chart of the steps of Genetic algorithm.



Initial Population:

After 1st crossover and mutation

After 2nd crossover and mutation

After 3rd crossover and mutation

The fitness criteria that have selected in this study are the tour which has values 17,22,23,24. Then, it will go to the crossover and mutation operators. After crossover and mutation, the resultant tour value is 15 [13].The study has showed that the proposed algorithm is useful to solve the TSP problem.

2.3 PARTICLE SWARM OPTIMIZATION (PSO)

Particle Swarm Optimization (PSO) is inspired by simulating social behavior of the movement of organisms in bird flocking and fish schooling and it is develop by Dr. Kennedy and Dr. Eberhert in 1995. PSO is a computational method that uses to optimize a problem by iteratively. PSO optimizes problem by having the population of candidate solutions and moving the particles in the search-space which means population represent the swarm and each solution is the organisms that represent particles. Each of the particles has their own position and velocity which calculated according to the mathematical formulae. In this method, it imitates the real creature's behavior [1, 10]. Velocity is important in PSO and it is assign to each of the particle since the velocity value is used to update the particle's position in each of the iteration. The new velocity of each particles will calculated during each iteration and the process will stop when reaching the stopping conditions [10].

According to document [10], each particle can define as below in the PSO algorithm.

i. Have own position and velocity.
ii. Know own position and its associated value.
iii. Know the best position it have reach previously and its associated value.
iv. Know its neighbors and its neighbors' best position and values.

Based on the statement above, the best solution or the fitness that achieve so far by the particle is call *pbest* (personal best). While, another value is the best value which achieved so far by any particle in the neighborhood of the particle is call *gbest* (global best). The particle will change their positions by the changing of their velocities towards *pbest* and *gbest* [10].

The basic algorithm of PSO is first to find the two best value, which is *pbest[]* and *gbest[]*. Then the particles update its velocity and position with following equation (a) and (b) [19].

$$v[] = v[] + c1 * rand() * (pbest[] - present[]) + c2 * rand() * (gbest[] - present[]) \quad \text{(a)}$$

$$present[] = persent[] + v[] \quad \text{(b)}$$

where *v[]* is the particle velocity, *present[]* is the current particle or called solution. *rand()* is a random number between (0,1). *c1*, *c2* are learning factors that usually *c1* = *c2* = 2.

The basic pseudocode for PSO procedure is as below:

```
For each particle
    Initialize particle
END

Do
    For each particle
        Calculate fitness value
        If the fitness value is better than the best fitness value (pbest) in history
            set current value as the new pbest
    End

    Choose the particle with the best fitness value of all the particles as the gbest
    For each particle
        Calculate particle velocity according equation (a)
        Update particle position according equation (b)
    End
While maximum iterations or minimum error criteria is not attained
```

The general algorithm of PSO also can be present in a simple flow chart as below:

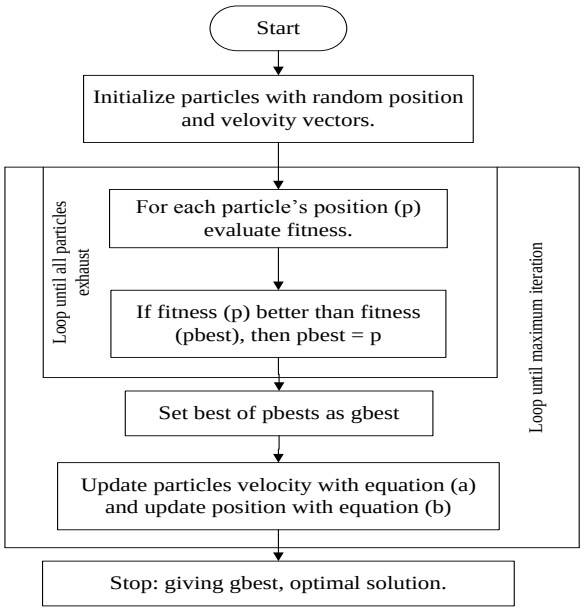


Figure 2.7: Basic PSO Algorithms

The advantage of using PSO as an optimization method is that PSO does not rely thoroughly on the problem that use to be optimized, therefore the method can readily employed for a host of optimization problems [18]. Besides that, another important advantage is that PSO technique can generate high-quality solution within shorter calculation time and it has stable convergence characteristic compared with other method [9].

2.4 RELATED WORK FOR PARTICLE SWARM OPTIMIZATION (PSO)

In this section, the related works which have solved by using PSO will be explained in detail. The shortest path problems are as introduce previously which are Robot Path Planning, Scheduling and Travelling Salesman Problem. Based on the papers, the way that the algorithm of PSO to apply in each of the problem is show and some of the algorithm is being advanced for improvement.

2.4.1 Application of Particle Swarm Optimization for Robot Path Planning

Robot path planning is one of the important tasks for the mobile robot which to generate an optimal shortest path walkway to complete their tasks in minimum time consume and most of the related research are also concern on obstacles avoidance while robot passing through the path [3, 4]. Obstacle avoidance is one of the important aspects for the mobile robot to achieve some controlled objective. This problem can solve by several algorithms but in these few studies is solved by Particle Swarm Optimization (PSO) by applying the algorithm. In the optimization function, it will strictly penalize the trajectories which cause the collision with the obstacles [4].

In the study, the obstacle avoidance is along the way find the shortest path from actual position named as *S* of the controlled mobile robot to the specified goal destination *G*, with respect to position and shapes of the obstacles *O* [4]. In addition, the input of the algorithm will be all of the parameters have stated previously.

For the particle information in this study are also similar with the general Particle Swarm Optimization (PSO) descriptions, which there are two most important

kinds of information to the particles. The first one is the best known position of particle or called their own experience which the choice have tried and know which is the best position achieved so far. The other information is the best position of the entire swarm or called by social knowledge which the performance of the other individuals in the neighborhood [4]. The velocity update rule and position update rule in this study are as below:

Velocity update rule:

$$\begin{aligned}\vec{v}_i(t) &= w\vec{v}_i(t-1) + \\ &+ \Phi_1(\vec{p}_i - \vec{x}_i(t-1)) + \\ &+ \Phi_2(\vec{p}_g - \vec{x}_i(t-1))\end{aligned}$$

Position update rule:

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t)$$

Besides that, the PSO algorithm in this study will count the fitness function with the following expression:

$$f = \frac{l}{l_{MIN}} + (\frac{\alpha}{d})^2$$

l_{MIN} = Euclidean distance between actual and desired robot position.

α = constant that determine the influence of the obstacles.

l, d = qualities of each particles

Moreover, computation the length of the trajectory is by:

$$l = \int_0^1 \sqrt{(f'(t))^2 + (g'(t))^2}dt$$

l= length of trajectory

d= minimum distance between trajectory and the closest obstacle

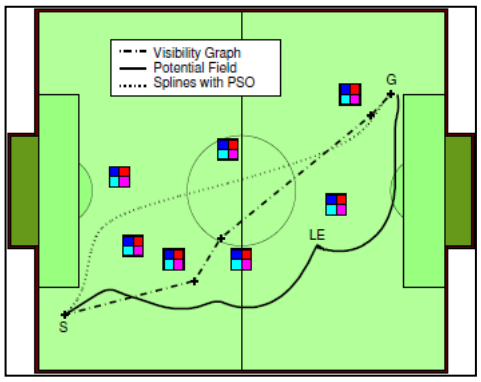


Figure 2.8: Standard situation in robotic soceer solved by different approachs [4].

In the result of the study, it is show that PSO method is success to avoid the obstacle wall by comparing with other approaches even there are no free path exists on the initial population. The study have been found that PSO algorithm is useful to find an optimal path for the mobile robot and the path discover is smooth and is easy to execute by the robot [4].

2.4.2 Application of Particle Swarm Optimization for Timetable Scheduling Problem

Nowadays, timetabling problem is very common problem that will be implementing in academic purpose. Timetabling is usually presents a set of tasks and resources such as classes, rooms, groups and instructors [20]. The parameters which concern in this problem are under some constraints and limited resources. Besides that, there are few types timetabling scheduling for academic purpose for example examination and course. In document [9], they have developed an application on the University Course Timetabling Problem (UCTP). The objective of the development is to solve the timetabling problem in term of university course and also to minimize the timetabling clash problem.

In this study, the software that use for simulation and programming is MATLAB 7.0 and this project follow the timetable structure of Falculty of Electrical Engineering, Universiti Teknologi Malaysia. Besides that, Particle Swarm Optimization is apply to solve this problem since it is a computational technique which is aim to find the point with minimum value of function [9].

Below shows a simple flow chart which present the flow of implementation and works of this project.

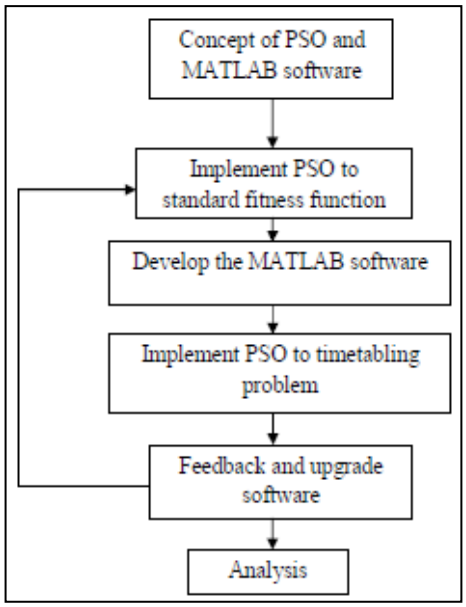


Figure 2.9: Project Overview and Planning of timetabling scheduling.

Based on the flow chart of this project, firstly, must have knowledge on MATLAB software before proceed to implement the approach to timetabling scheduling problem. The developer need to ensure that the objectives of the project are achieved, and then the performance on MATLAB is analyzed and to find better solution by updating the programming code.

By applying PSO to timetabling problem, the particle is the solution according to the constraint and the purpose is to find the minimum optimal distance. However, there are some constraints that need to be concern for this problem such as ensure that no clash exist in the timetable and so on [9]. In addition, there are two requirements that need to consider while writing the codes: a) the number of particle and b) ensure the particle choose the right value.

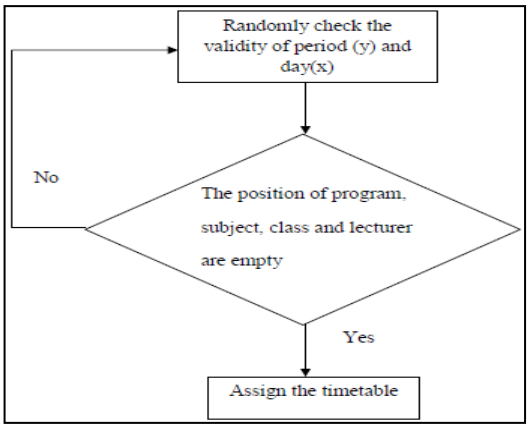


Figure 2.10: Flow chart for assigning the timetable.

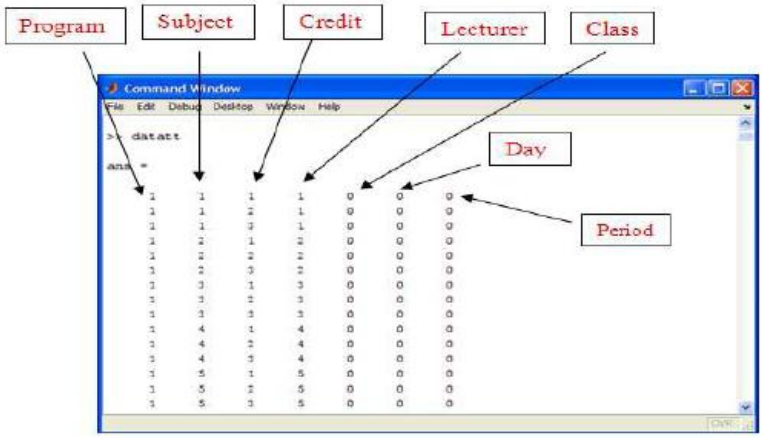


Figure 2.11: The data assign in timetable.

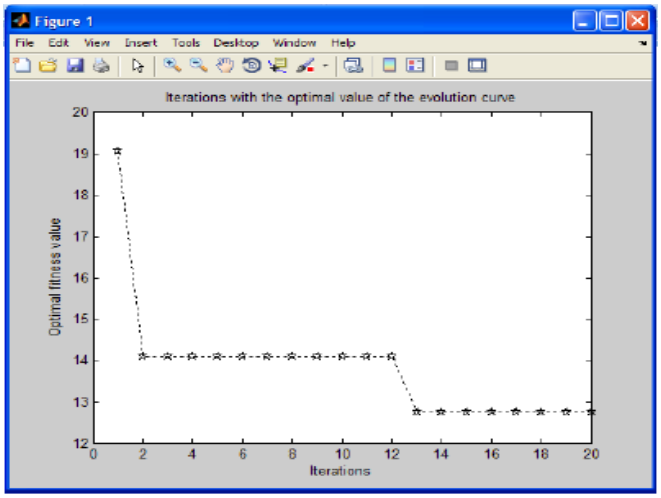


Figure 2.12: Graph of iteration in PSO

As a result, PSO is successfully applied in this project to solve the University Course Timetable Problem (UCTP) since the parameter in timetable fulfills the several constraints to obtain an optimal solution [9].

2.4.3 Travelling Salesman Problem using Particle Swarm Optimization

Travelling Salesman Problem (TSP) is well known as a NP-hard combinatorial optimization problem. Since Particle Swarm Optimization (PSO) has proven that it is success in many works which regarded to continuous-variable optimization problems, so it is use to apply its algorithm in the TSP problem which is a discrete problem [14].

In a paper, the researcher consider on how to apply the PSO to Travelling Salesman Problem and with the objective to find a tour with minimum length. In some study, the Hamilton cycle is use to represent the tour [16].

In another study, the generalized travelling salesman problem (GTSP) is a travelling salesman problem where a tour does not necessarily to visit all nodes [12]. Due to this, the set of *N* nodes is divided into *m* sets. Since the standard PSO equation is unable to be used to generate binary or discrete values [12], therefore the discrete PSO (DPSO) algorithm is used in the study where the DPSO algorithm is still using the same concept with standard PSO algorithm.

Below the DPSO algorithm:

```
Procedure DPSO
initialize parameters
initialize particles of population
evaluate particles of population
apply two_opt local search to personal best population
apply VND local search to personal best population
while (not termination) do
    find the personal best
    find the global best
    update particles of population
    evaluate particles of population
    apply two_opt local search to personal best population
    apply VND local search to personal best population
endwhile
return Global best
end
```

Figure 2.13: DPSO algorithm for GTSP.

Below is another study which applied by using PSO:

```
procedure Discrete_PSO
    /* Define initial probabilities for particles' moves:*/
    pr1 ← a1 /*to follow its own way*/
    pr2 ← a2 /*to go towards pbest*/
    pr3 ← a3  /*to go towards gbest*/
    /* a1+ a2+ a3=1 */
    Initialize the population of particles
    do
      for each particle p
          value_p ← Evaluate(x_p)
          if (value(x_p) < value(pbest_p)) then
            pbest_p ← x_p
          if (value(x_p) < value(gbest) ) then
            gbest ← x_p
      end_for
      for each particle p
          velocity_p ← define_velocity(pr1, pr2, pr3)
          x_p ← update(x_p,velocity_p)
      end_for
      /* Update probabilities*/
      pr1 = pr1×0.95; pr2 = pr2×1.01; pr3 = 1-( pr1+ pr2)
    while (a stop criterion is not satisfied)
```

Figure 2.14: Pseudocode of PSO for discrete optimization problems

While, in the continuous-variable optimization problem is the problem that find a real valued vector with the minimum objective function value which applied by PSO [16].

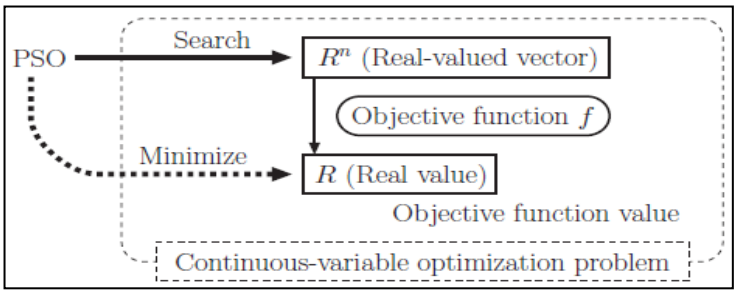


Figure 2.15: Continuous-variable Optimization Problem

Below is the simple outline for the method use in the study.

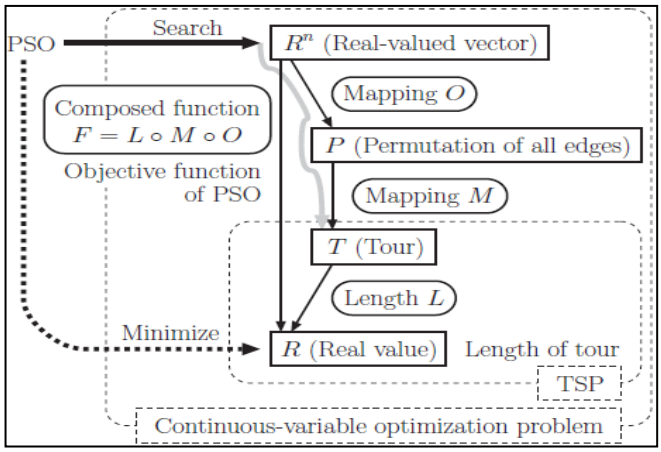


Figure 2.16: Outline of the Proposed Method

The study that presented DPSO to solve GTSP have been showed that DPSO algorithm having the best performing if compare with the other algorithm which implemented at the same time [12]. In the study which using continuous-variable optimization problem, it has show that the method able to reach the good solutions faster than other method on the early stage of the search process and also reach the better solutions than other at the end of the search process [16].

2.5 CONCLUSION

In this chapter, three main sections are have been discuss which are the type of shortest path problem, detail of Particle Swarm Optimization and the work related with Particle Swarm Optimization.

In the type of shortest path problem part, the three popular shortest paths have been explained. Besides that, the techniques other than Particle Swarm Optimization such as Ant Colony and Genetic Algorithm also have been explained along with the works which related to them.

Next, the detail about Particle Swarm Optimization have discuss by providing the basic algorithm and the flow chart regarding with this technique's process. In addition, the related work with Particle Swarm Optimization also discuss in this chapter.

CHAPTER 3

METHODOLOGY

3.1 INTRODUCTION

This chapter will explain about the methodology for developing the system which is Simulation of Identifying Shortest Path Walkway using Particle Swarm Optimization. This chapter will discuss the general method use in this project during the development process. The activities that conducted during each of the phases will be discussed in detail.

3.2 DEVELOPMENT METHOD

The development process for a system is depends on the method that uses to play a role as reference before the system can develop successfully. Due to this, a software process model is planned to be the system development method for this project.

The model for this project consists of six phases which are:

a) Planning
b) Requirement
c) Analysis
d) Design
e) Implementation
f) Testing

Each of these phase will proceeds to the next phase once a development is complete, each of the phase also may turn back to the previous phases. This means that the model is reversible. Since the model is reversible, if there is any changes happen during the involve phase, it can turn back to the previous phase for the modification. Due to this, this model use is adaptable and flexible.

The activities in each phase are described clearly using this development method, this is important for the development process of this project.

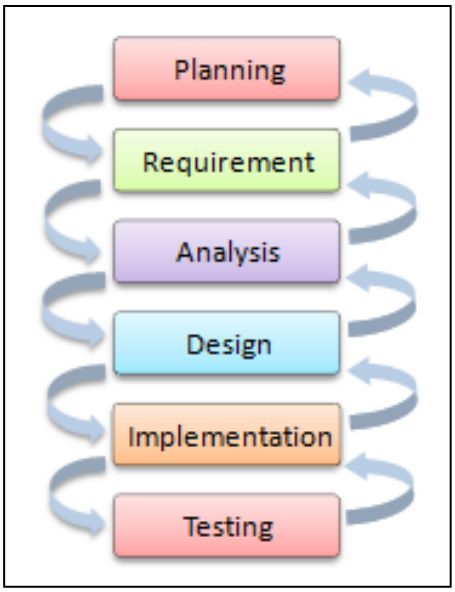


Figure 3.1: Flow of Development Model

3.3 PLANNING

Planning is the early stage of the system development and the task to creating a software program is important in extracting the requirements. Planning is the phase that important for a project before proceed to the next phase since this phase determine the basic for the whole project. All of the vital information such as goals, scope and needs of the project are obtained from the discussion with supervisor Miss Zalili binti Musa.

In this project, there are some events involve in this phase. Meeting with supervisor is purposely to discuss on the important information and decided some important issues.

3.3.1 Consultation with Supervisor

There are several important things that relevant with this project have been discussed during the meeting.

a) The project title is discussing in detail which is concern on shortest path.
    - The title of the project is being discussed as to understand clearly on the concept of the whole project which is to identify the shortest path for certain purpose.

b) The purpose for finding the shortest path is decided.
    - To find shortest path walkway for a simulated environment in order to reduce time consume. The simulated environment that suggested is a restaurant.
    - The simulated environment will be design by own self for this project.

c) A suitable technique is considered.
    - Particle Swarm Optimization (PSO) is choose as the technique to implement in this project.

d) The title of this project is decided.
    - The project title is named as Simulation of Identifying Shortest Path Walkway using Particle Swarm Optimization (PSO).

3.3.2 Planning the Project Milestone

Gantt chart is plan and creates as to summarize the schedule of the system flow. This is to ensure that each of the development process is follow and also to avoid out of time of the development process. Please refer to Appendix A for the Gantt chart.

3.4 REQUIREMENT

The requirement means to captures all of the possible requirements of the system which to be develop. Without the good requirements, the project cannot be developing successfully. The requirements are specified by depend on the system satisfaction.

There are two main activities in this phase.

i. Requirement and scope derivation
ii. Hardware and software requirements

3.4.1 Requirement and scope derivation

In this project, the scope is defined after discussed with supervisor and the requirements are derive from the scope of the system. Below show the scope of the project:

a) Set the simulated environment as a restaurant.
b) Number of tables with minimum 9 tables and maximum 25 tables.
c) Microsoft Visual Basic 2010 for programming and simulation.
d) C programming language is implementing in this project.
e) Apply Particle Swarm Optimization (PSO) algorithm.

3.4.2 Hardware and Software Requirements

There are several different software and hardware to be used in this system for the development of Simulation of Identifying Shortest Path Walkway using Particle Swarm Optimization (PSO). Hardware is the devices that to be use which means the physical aspect while software is the program that used to operate such devices or computer. The following section will display all of the hardware and software specifications for the system development of this project.

3.4.2.1 Hardware Specification

The hardware required used to develop this system is listed as table below.

Item	**Required**
Processor	Intel Core i5 2.30GHz
RAM	4GB

Table 3.1: Hardware Requirements

3.4.2.2 Software Specification

The software required used to develop this system is listed as table below.

Item	**Name**	**Purposes**
Operating System	Windows 7 Home Premium	As a operating system
Software	Microsoft Visual Studio 2010	To develop the system coding.
	Microsoft Office Word 2007	To do the thesis documentation.
	Microsoft Office Project 2007	To do the project Gantt Chart documentation.
	Microsoft Office PowerPoint 2007	To do the project presentation slide.
	Microsoft Office Excel 2007	To record the testing result.
	Adobe Reader 9	To read pdf file.

Table 3.2: Software Requirements

3.5 ANALYSIS

The analysis is the phase before proceed to the design stage. In this phase, the information that related with this project is gathering for referring purpose. The source of information such as book, thesis, online article, online journal and the other materials which related to the project is gathering and analyse on the part that is needed for the project.

In this phase, there are some activities have done.

i. Search for relevant information
ii. Define types of shortest path and techniques use to solve
iii. Analysis on implementation technique
iv. Interview with restaurant's customers

3.5.1 Search for relevant information

The information that have found is analyse and the purpose of this is to gain more knowledge from the source have found which it can help in the development of project. During this moment, shortest path and Particle Swarm Optimization (PSO) is the keyword to search the information about the types of shortest path problems and the basic algorithm of Particle Swarm Optimization (PSO), which this is the basic knowledge needed to develop this project.

From the article and journal, the researches that have done by other researcher can be use as a guideline in this project, for example:

a) The way to implement the technique.
b) The way to apply the algorithm.
c) The flow of the system development.
d) The existing system for the shortest path problem.

After gather the information, it is helps in the literature review that has done previously. The journal and article which concern on shortest path problem and Particle Swarm Optimization has discussed in chapter 2. By referring to the research paper, the existing system referred give ease to more understand on this system.

3.5.2 Define types of shortest path and techniques use to solve

According to some research paper, shortest path problem is a problem that to find the quickest way to move from one location to another location.

The types of shortest path problem are includes:

a) Robot motion planning
b) Scheduling
c) Travelling salesman problem
d) and others

The research paper also states that those types of shortest path can solve by using various different techniques such as:

a) Dijkstra algorithm
b) Ant Colony
c) Genetic Algorithm
d) Particle Swarm Optimization
e) and others

3.5.3 Analysis on Implementation Technique

For this project, Particle Swarm Optimization (PSO) is use as the techniques to implement to the system as state in the previously. According to the research paper, Particle Swarm Optimization is readily employed for optimization problems and this technique can generate high-quality solution within shorter calculation time. Based on those research papers, it has proved that Particle Swarm Optimization can solve the problem effectively and this is a strong reason for choosing this technique.

3.5.4 Interview with restaurant's customers

Other than the information that search from book or journal, an interview session also held with the restaurant's customers by randomly which purposely to collect their opinions and comments about the services.

In this interview session, the summary of the opinions and comments that collect from the customers are:

a) The waiter takes long time to deliver food to the customer.
b) The waiter sometime needs to serve more than one customer in one time if there are many customers.

Due to this, there is time consume or time delaying for the waiter to deliver food without shortest path and this is prove that shortest path is needed to solved in order to reduce time consume.

3.5.4.1 Constraints

The constraints for identifying the shortest path walkway:

a) Size of the area
   - The size of the area may affect the distance of the path.
   - The bigger the area's size, the larger the distances of the objects.
   - Therefore, the area size of this system is set to fit with minimum 9 tables and maximum 25 tables.

b) Arrangement of tables
   - The arrangement of table is important since table represent the node for the waiter to pass through for reaching the destination.
   - The tables will arrange in grid form in term of row and column.
   - The number of tables that decided in row and column, and there is 1 counter by default.

Type	**Tables in row x column**	**Counter**
a	3 x 3	1
b	3 x 4	1
c	3 x 5	1
d	4 x 3	1
e	4 x 4	1
f	4 x 5	1
g	5 x 3	1
h	5 x 4	1
i	5 x 5	1

Table 3.3 Arrangement of tables

c) Distance between tables

- Distance is defined as the fitness function in the algorithm and this is the key to identify the shortest path.
- The distances between tables are set to be 1 meter in row and also 1 meter in column for this system.

d) Size of tables

- The tables in this system are all in equal size.

e) Number of waiter

- There are set to be 1 waiter.

f) Number of counter

- There is only 1 counter in this system.

The other constraints for this system is that this system cannot solve the path with large number of tables since the number of table and number of waiter is set as mention in above. This system only can solve the shortest path in the small scale environment of simulated restaurant.

3.6 DESIGN

In this phase, the system outlook is starting to create based on the requirements and planning have done previously. The system design is prepared and it is help in the overall system architecture. In the other words, design phase is purposely to transform requirements into design.

3.6.1 Flow of system

The first steps are to set the arrangement of tables for the system which fits with the requirements. The requirements will be the input to be inserting by the user, which is number of tables. Secondly, apply Particle Swarm Optimization algorithm to the system. Therefore, after key in the entire requirement, the result will show the generated shortest path. The flow of system is show in the figure below.

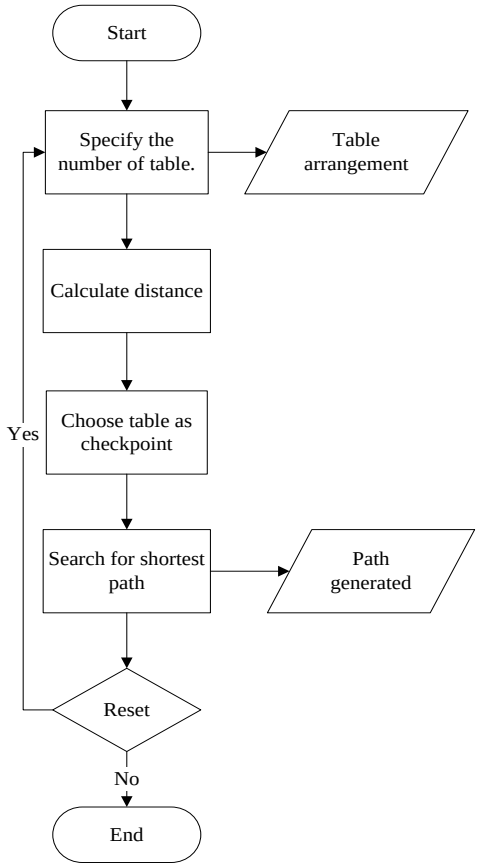


Figure 3.2: Flow chart of system

3.6.1.1 Specify the number of table

The number of table is set to minimum with 9 tables and maximum with 25 tables in this system. While determine the table arrangement, the user needs to specify it in term of row and column. The number of waiter and number of counter are automatically set as 1.

3.6.1.2 Calculate distance

After the user have insert the number of table, the distance for all of the potential paths will be calculated in this step according to the number of table that insert by user. This calculation will do with heuristic approach.

a) Heuristic approach

Heuristic is use in this system for the calculation of the potential paths. This approach will determine which will be the next nodes to proceed. Besides that, some rules are requiring for this heuristic approach.

i. The node only can move in three directions which are down, right or left.

ii. The node cannot cross to the other node.

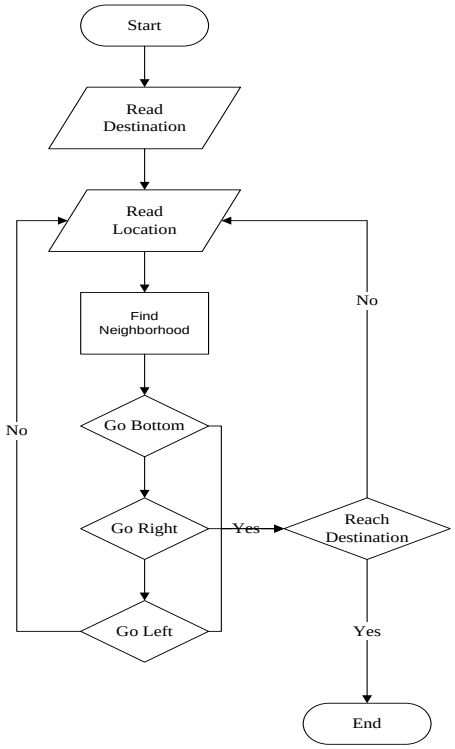


Figure 3.3: Flow chart of Heuristic approach

The flow chart above indicates the flow for the heuristic approach that use in this project.

i) Read destination

- In this project, the destination is the counter.

ii) Read Location

- The location is the tables involve which means the value inserts by the user.

iii) Find neighbourhood

- Based on the location, the location will be the first node, and then the first node will find the path to go to the destination.
- As shown in the flow chart above, the node will first determine the direction to go.
- Firstly, the node will determine whether to go bottom, right or left.
- If there is either one direction to go, it will proceed to the next step which is to determine is it reaching the destination. If there is no direction to go, it will back to Read Location and repeat the process.
- In Reach Destination steps, if "No", it will back to the Read Location and repeat the process. If "Yes", it will end the process.

b) Design of Heuristic approach

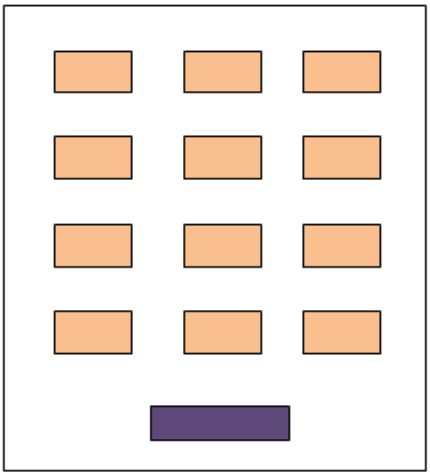


Figure 3.4: Arrangement of counter and tables

The figure above shows the arrangement of counter and tables, which the purple colour represent the counter and the pink colour represent the tables.

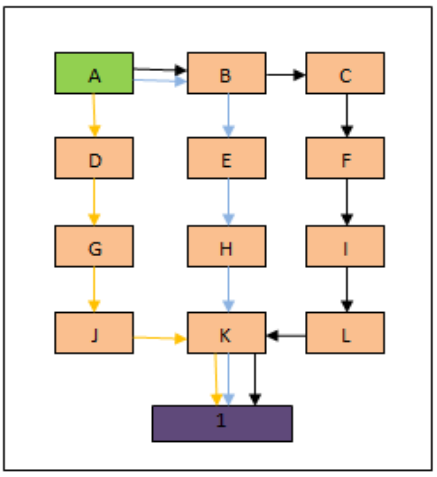


Figure 3.5: The Heuristic approach

As shown in figure above, the purple colour box represent the counter with label 1 and the tables is represent by pink colour box with label A to L. To find the potential paths, the only green colour box with label A is the starting point and the only counter is the destination point. The figure has show the path which can pass by the waiter from the starting point to destination point. The paths have show by arrows with different colours.

There are several different paths have show in the figure to reach the destination 1 from starting point A. These paths is solving by using Heuristic approach and follow its rules, which the nodes will not going upward and they will just go to bottom, right or left. The paths have show in the interface are:

i) ABCFIL1
ii) ABEHK1
iii) ADGJK1

Besides the paths have shown in the figure, there are also the other path that may generated, for example:

i) ADEHK1
ii) ABCFEHK1
iii) ADGHK1

3.6.1.3 Choose table as checkpoint

According to the number of tables that specify by the user, user can choose any table to be the checkpoint to generate the shortest path.. The counter will automatically become the checkpoint since there is only one counter and it is set as default.

3.6.1.4 Search for shortest path

The shortest path is search according to the check point that chooses by the user in previous steps. In this part, it involves the calculation in the algorithm to find any potential path and then determine the shortest path. The path that has been generated will be show which indicates the shortest path can take by waiter to perform their task. The algorithm of Particle Swarm Optimization is applied in this step.

3.6.1.4.1 Definition of fitness function

In this system, the fitness function is defined as the distance of the path in Particle Swarm Optimization. The fitness function is to determine the shortest path by calculate and compare among those potential path.

3.6.1.4.2 Particle Swarm Optimization (PSO)

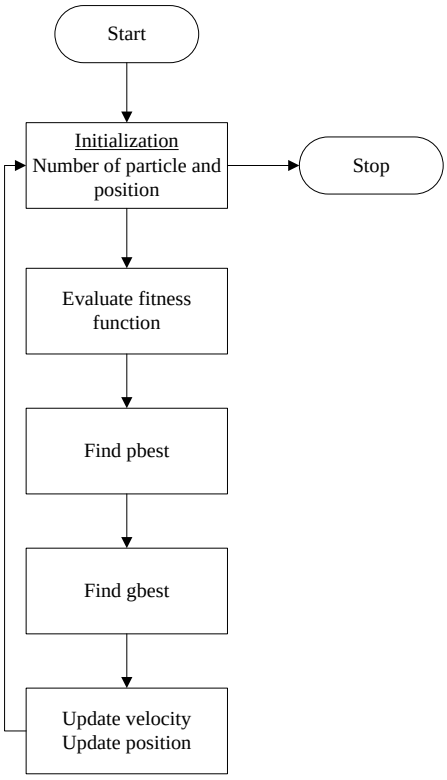


Figure 3.6: Flow chart of basic PSO Algorithm

a) Initialization
   - In this step, the number of particle and iteration is set and also initialize the particles with random position.

b) Evaluate fitness function
   - The fitness of each particle is calculated based on the fitness function.

c) Find pbest
   - Analyze the trajectory of single particle, for example if fitness for the particle position (p) is better than fitness of current pbest, then the pbest = p.

d) Find gbest
   - Gbest is the best solution between the pbest. Therefore, the particle will compare all of the pbest and set the best of pbest as gbest.

e) Update velocity and position

- Update position and velocity based in the two equations.
- *v[] = v[] + c1 * rand() * (pbest[] - present[]) + c2 * rand() * (gbest[] - present[])* (a)

  present[] = persent[] + v[] (b)

f) Stop

- It will stop after giving the gbest and the optimal solution.

3.6.1.5 Reset

The system can reset to search for another different check point after all the previous steps is done. Otherwise, user can choose to quit this system.

CHAPTER 4

IMPLEMENTATION AND TESTING

4.1 INTRODUCTION

This chapter will explain and discuss about the implementation of Heuristic method and Particle Swarm Optimization (PSO) in this project. Testing part also will be discussed in the next section.

4.2 IMPLEMENTATION

This implementation phase is to transform the design and the overall system flow into an actual code after the transform of requirement into design. The coding is develop using Microsoft Visual Studio 2010 since it is supported the C programming language. This phase is important since the development of code must be complete before the system testing.

This project is about indentifying shortest path by using Particle Swarm Optimization (PSO) and the implementation of this technique will be explained. Besides that, the Heuristic approach which is the method to find out all the potential paths also will explain.

4.2.1 Heuristic Approach

Heuristic is use to calculate the potential path in this system. This method plays the important role to determine the next node to be proceeding for one path. Therefore, Heuristic has some rules during searching the potential paths.

a) The node only can move in three directions which are down, right or left.

b) The node cannot cross to other node.

Below is the code that has implement with Heuristic based in the rules:

```
prev[0]=i;
        prev[1]=j;
        while(count<50)
        {
                choice=rand()%3+1;
                switch(choice)
                {
                case 1:{
                        if(j==0) break;
                        else if ((prev[0]==i)&&(prev[1]==j-1))break;
                        else{
                                prev[0]=i;
                                prev[1]=j;

                                printf("-> %d",table[i][--j]);//go left
                                storePath[count][index]=table[i][j];
                                index++;
                                break;
                        }
                }
                case 2:{
                        if(j==2) break;
                        else if ((prev[0]==i)&&(prev[1]==j+1))break;
                        else{
                                prev[0]=i;
                                prev[1]=j;
                                printf("-> %d",table[i][++j]);//go right
                                storePath[count][index]=table[i][j];
                                index++;
                                break;
                        }
                              }
                case 3:{
                        printf("-> %d",table[++i][j]);//go bottom
                        prev[0]=i;
                        prev[1]=j;
                        storePath[count][index]=table[i][j];
                        index++;
                        break;
                        }
                default:printf("wrong\n");
                        break;
                           }

                //if (i>2)
                if (i==3) //last row
                {
                        i=j=0;
                        index=0;
                        printf("\n%d",table[i][j]);
```

```
                    //count++;
                    storePath[count++][index]=table[i][j];
                    index++;
                }
                }
```

Figure 4.1: Coding for Heuristic Approach

4.2.2 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is the technique to use in this project to identify the shortest path walkway among the potential paths that have generate by Heuristic approach.

a) Set the position

Before applying the Heuristic approach, the position is set by the user to find the shortest path. This is means that user choose the starting point which to pass through to reach the destination.

b) Find the possible potential paths

This is done by the Heuristic approach as explain in 4.2.1. All the potential path that can pass for the position that have chosen by user will be generate. However, Heuristic may generate the repeat path among those potential paths.

c) Count the number of element of each path (Fitness function)

After Heuristic generates all the potential paths, the number of element within the path is count. The number of element represents the distance of the path. The number of element will be display along with the path. The number of element represents the fitness function in Particle Swarm Optimization.

```
printf("\n");
        for (i=0;i<50;i++)
        {
        printf("\n StorePath %d: ",i+1);
        count=0;
        for (;;)
        {
```

```
                if (storePath[i][count]>0)
                {
                        printf(" %d",storePath[i][count]);
                        count++;
                }
                else break;
        }

        printf("\n Number of elements: %d \n",countArray(storePath[i]));
        }
```

Figure 4.2: Coding for show the path and number of element for the path.

d) Search for optimal shortest path

After generate the potential paths, the number of node that pass through by a path is counted and this become the number of element for a path which represent the fitness function in Particle Swarm Optimization (PSO). The last part is to apply Particle Swarm Optimization (PSO) technique to identify the shortest path by compare the number of element (fitness function) of each path. The shortest path is the path with smallest value in number of element.

In detail, the initial step for applying the Particle Swarm Optimization (PSO) is to randomly choose five potential paths for iteration and get the local best (*pbest*) and global best (*gbest*) for the iteration. The best value that obtain will bring to the next iteration to be compare and to get the latest *pbest* and *gbest*. Besides that, there is one important step while apply this techniques which are the velocity and position of the particle will be updated during each iterations. The method use to update the velocity and position is as shown in below.

```
for (int i=0;i<p_num;i++)
{
        particle_v[i] = w*particle_v[i] + c1*rand()*(particle_loc_best[i]-
particle[i]) + c2*rand()*(particle_glo_best-particle[i]);
}
particle_p[i] = particle_p[i] + particle_v[i];
```

Figure 4.3: Coding for update velocity and position of path.

4.3 TESTING

In testing phase, the code is compiled to check whether it is running or not and to detect the error. If there is any error occur in this phase, the error will be detected and to fix the code. This testing phase is to ensure that the project is run according to what it is expected. This phase also check the performance of this system to determine whether it provides the outcome as expected. In this system, the path obtains in this system is tested to check whether the path generate is the optimal path for the waiter. Due to this, there is some testing that will perform:

4.3.1 Black-box testing

Black-box testing is a method of testing software that to test in the aspect of system functionality of an application. This testing will be done by based on unit by unit, so when a specific function have done it will tested before proceeding to the next progress. There are several function have test by black-box testing.

a) Heuristic approach

The potential path are generated after apply the Heuristic Approach. In the code, it is set to show 50 potential paths. Below are some of the results that generate after applying the Heuristic Approach. However, as seen in the testing result, there consists of some repeated paths.

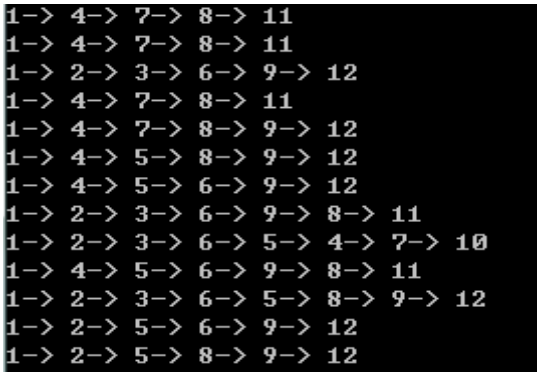


Figure 4.4: Potential paths

b) Count the number of element

The number of element for each of the potential path is then counted and display along with the path. The numbers of element show the total number of checkpoint too pass through for each path. Below are some of the results when compiled the code.

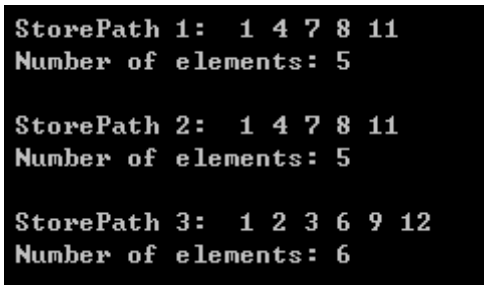


Figure 4.5: Number of element for each potential path

c) Identify shortest path

Identify the most optimal shortest path is the final step for this system. The generated paths will be comparing by using the number of element which represent fitness function through iteration to find the shortest path. In this system, the path which has small number of element is the shortest path.

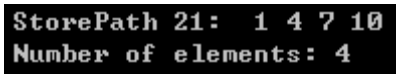


Figure 4.6: Shortest path.

4.3.2 Alpha Testing

Alpha testing will be conducted to test on the user interface and this testing will done by programmer. This alpha testing is to test on the overall function of the system to ensure that it can run smoothly and free of error.

Firstly, the tester needs to ensure that the table is shown in the grid form within the range have set. This is to check whether the system has fulfilled the requirements. After that, the starting point and destination is decide to generate the shortest path.

CHAPTER 5

CONCLUSION

5.1 INTRODUCTION

This chapter will conclude the conclusion for chapter 1, chapter 2, chapter 3 and chapter 4 for this project. Besides that, the overall conclusion also has made to conclude the whole project and the future research also have discussed in the last section.

5.2 CONCLUSION OF CHAPTER 1

In this chapter, the introduction of this project is discussed which is to explain the background of optimization, shortest path problem and Particle Swarm Optimization (PSO) technique that related with this project. It is also follow by the problem statement which is to figure out the problems that lead to do this project. Besides that, the objectives for doing this project have defined and also set the scope for this project. Lastly, the thesis organization has described the organization of the complete full report.

5.3 CONCLUSION OF CHAPTER 2

In this chapter, the review on the research paper is done based on the shortest path problem, Particle Swarm Optimization (PSO) and the algorithm of related work. By review to the shortest path problem, there are different types of shortest path problem have found in the research paper along with the different technique use to solve this different shortest path problem. For example, robot motion planning, scheduling and raveling salesman problem solve by using Genetic Algorithm, Ant

Colony or Particle Swarm Optimization. Besides this, the detail about Particle Swarm Optimization (PSO) also discuss in this chapter along with the related work since it is the technique to be use in this project.

5.4 CONCLUSION OF CHAPTER 3

In this chapter, the development model for this project is design and have discuss in the project methodology section. The development model that design for this project consists of six phases which are planning, requirement, analysis, design, implementation and testing, each of the phase have explain in detail in this chapter respectively along with the activities that have done during each phases. In addition, the expected results for this project also have explained in this chapter and the interface for expected result also has shown.

5.5 CONCLUSION OF CHAPTER 4

In this chapter, the part of implementation and testing of this project have been explained in detail. Firstly, the Heuristic approach is implementing to find all the potential paths by determine which node to pass through in the system. Heuristic is apply according to the rules that have been set which are the node just can move in three directions and cannot cross to another node. While implementing the Particle Swarm Optimization (PSO) technique, the number of element which count the node of potential path that have generate by Heuristic will be use as the fitness function of each potential paths. In testing part, the black-box testing is use to test in the aspect of system functionality while alpha testing is conducted by the programmer to test on the user interface. Alpha testing is to test the overall function of the system.

5.6 OVERALL CONCLUSION

In this project, a system is purposely developing to solve the shortest path problem which is named by Simulation of Identifying the Shortest Path Walkway using Particle Swarm Optimization (PSO). The shortest path walkway which generate by this application is use to help the waiter in the simulated restaurant to

deliver the food with less time consume since they will follow the shortest path compute by the system.

Particle Swarm Optimization (PSO) is the technique that uses to generate the shortest path by applying its algorithm in the system. PSO able to optimizes a problem and search an optimal value through iteration. Besides that, PSO can approach the optimal solutions with fast speed. Therefore, PSO is chosen to be the technique in this project.

This project will develop according to the model that has planned. The process to undergo while develop this project are planning, requirement, analysis, design, implementation and testing. By applying the Particle Swarm Optimization (PSO) techniques, the shortest path will be found by counting the number of element in the potential paths and display the shortest path in the system interface. The system is expected to generate the optimum shortest path with Particle Swarm Optimization (PSO) to achieve the objectives which state in this project. This system can help in reduce the time consume in order to help in increasing the work load.

5.7 LIMITATION

There are some limitations have found in this system of Simulation of Identify Shortest Path Walkway using Particle Swarm Optimization (PSO).

In this system, the first limitation is the arrangement of table in the system's environment. It is already set to a specific value which with minimum nine tables and maximum 25 tables. This means that the number of table could not be add or even modify. The area size is also fixed since this project aim to test on the smaller scope. Besides that, the arrangement of table is fix to be in grid form.

Secondly, the node which represents the table cannot be passing through with cross direction since this is following the rules that have set in the Heuristic Approach. This is also means that the waiter can walk in three directions only which are either right, left or down. Therefore, the result if the path generated is only in those three directions.

5.8 FUTURE RESEARCH

For future research, there are some improvements that can make from this current system. In this current system, the number of table has fix with minimum 3x3 and maximum 5x5. Therefore, the size of area has fixed and also the size of the table is all the same. For the further development, the size of area can set to be a big scale restaurant. With this, it can be apply to the place such as the big scale food café.

Beside that, the arrangement of the environment can be improve to the other design other than same with this current system. This can test the functionality of the system. Moreover, the application can be apply to one or more mobile robot in future to save the use of men power and also help in the large scale restaurant or café.

REFERENCES

[1] http://en.wikipedia.org/wiki/Particle_swarm_optimization Retrieved 30th October, 2011.

[2] http://en.wikipedia.org/wiki/Particle_swarm_optimization Retrieved 7th November, 2011.

[3] Xin Chen and Yangmin Li. *Smooth Path Planning of a Mobile Robot Using Stochastic Particle Swarm Optimization.* IEEE International Conference on Mechatronics and Automation, Proceedings of the 2006, Luoyang, Henan 25-28 June 2006, pp. 1722-1727.

[4] Saska M., Macas M, Preucil L and Lhotska L. *Robot Path Planning using Particle Swarm Optimization of Ferguson Splines.* IEEE Conference on Emerging Technologies and Factory Automation, 2006. ETFA'06, Prague 20-22 Sept. 2006, pp. 833-839.

[5] Zhao Shaogang and Li Ming. *Path Planning of Inspection Robot Based on Ant Colony Optimization Algorithm*. In the Proceeding of International Conference on Electrical and Control Engineering (ICECE), 2010, Wuhan 25-27 June 2010, pp. 1474-1477.

[6] R. Rahimi-Vahed and S. M. Mirghorbani. *A multi-objective particle Swarm for flow shop scheduling problem.* Journal of Combinatorial Optimization, Volume 13, Number 1, pp. 79-102,2006.

[7] Ajith Abraham, Hongbo Liu, Weishi Zhang and Tae-Gyu Chang. *Scheduling Jobs on Computational Grids Using Fuzzy Particle Swarm Algorithm*. Proceedings of the Tenth International Conference on Knowledge-Based & Intelligent Information & Engineering Systems, KES 2006, Part II, pp. 500-507, 2006.

[8] Zeng Lingli, Zou Fengxing and Xu Xiaohong. *Mathematical Model and Hybrid Particle Swarm Optimization for Flexible Job-Shop Scheduling Problem*. Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation, 2009. New York, USA.

[9] Nurul Izzati binti Abdul Rahim. *Application of Particle Swarm Optimization for timetable scheduling problem.* Undergraduate Thesis Universiti Teknologi Malaysia,Malaysia.

[10] Elizabeth F. G. Goldbarg, Marco C. Goldbarg and Givanaldo R. *Particle Swarm Optimization Algorithm for the Traveling Salesman Problem.*2008.

[11] Jerome Le Ny. *On The Curvature-Constrained Traveling Salesman Problem.*

[12] M. Fatih Tasgetiren, P. N. Suganthan and Quan-Ke Pan. *A Discrete Particle Swarm Optimization Algorithm for the Generalized Traveling Salesman Problem.*Proceedings of the 9th annual conference on Genetic and evolutionary computation,New York, USA 2007.

[13] Fozia Hanif Khan, Nasiruddin Khan, Syed Inayatullah and Shaikh Tajuddin Nizami. *Solving TSP Problem by Using Genetic Algorithm*. International Journal of Basic & Applied Science IJBAS, Volume 9, Number 10.

[14] Kang-Ping Wang, Lan Huang, Chun-Guang Zhou and Wei Pang. *Particle Swarm Optimization for Traveling Salesman Problem*. Proceedings of Second International Conference on Machine Learning and Cybernetics, 2003, 2-5 Nov 2003, Volume 3, pp. 1583-1585.

[15] http://en.wikipedia.org/wiki/Travelling_salesman_problem Retrieved 9th November, 2011.

[16] Shigehiro Y., Katsura T. and Masuda T. *An Application of Particle Swarm Optimization to Travelling Salesman Problem*. Proceedings of SICE Annual Conference 2010, Taipei 18-21 Aug. 2010, pp. 1629-1632.

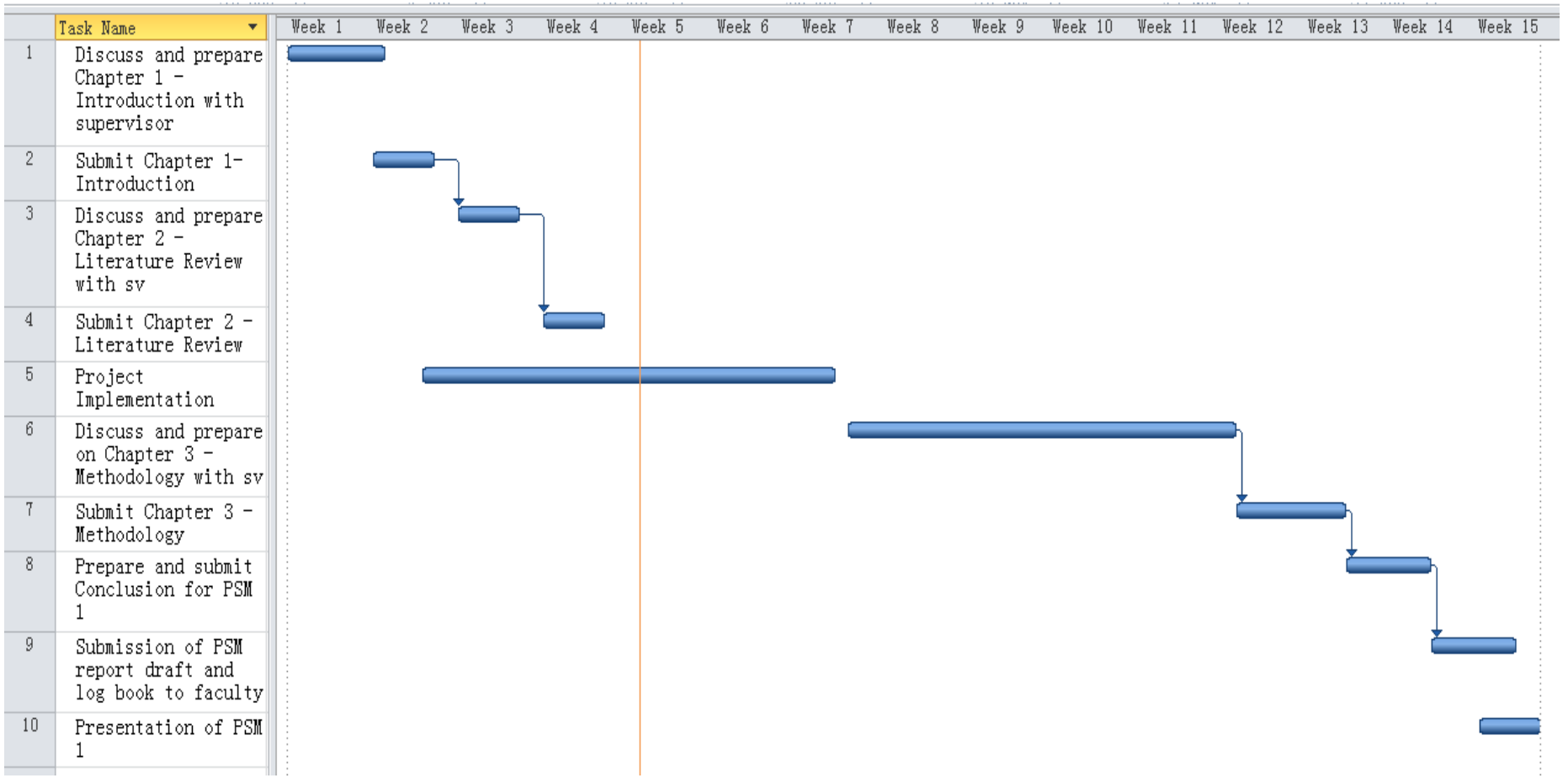
[17] Marco Dorigo, Gianni Di Caro and Luca M. Gambardella. *Ant Algorithms for Discrete Optimization*. Artificial Life, Volume 5, pp. 137-172, March 2006.

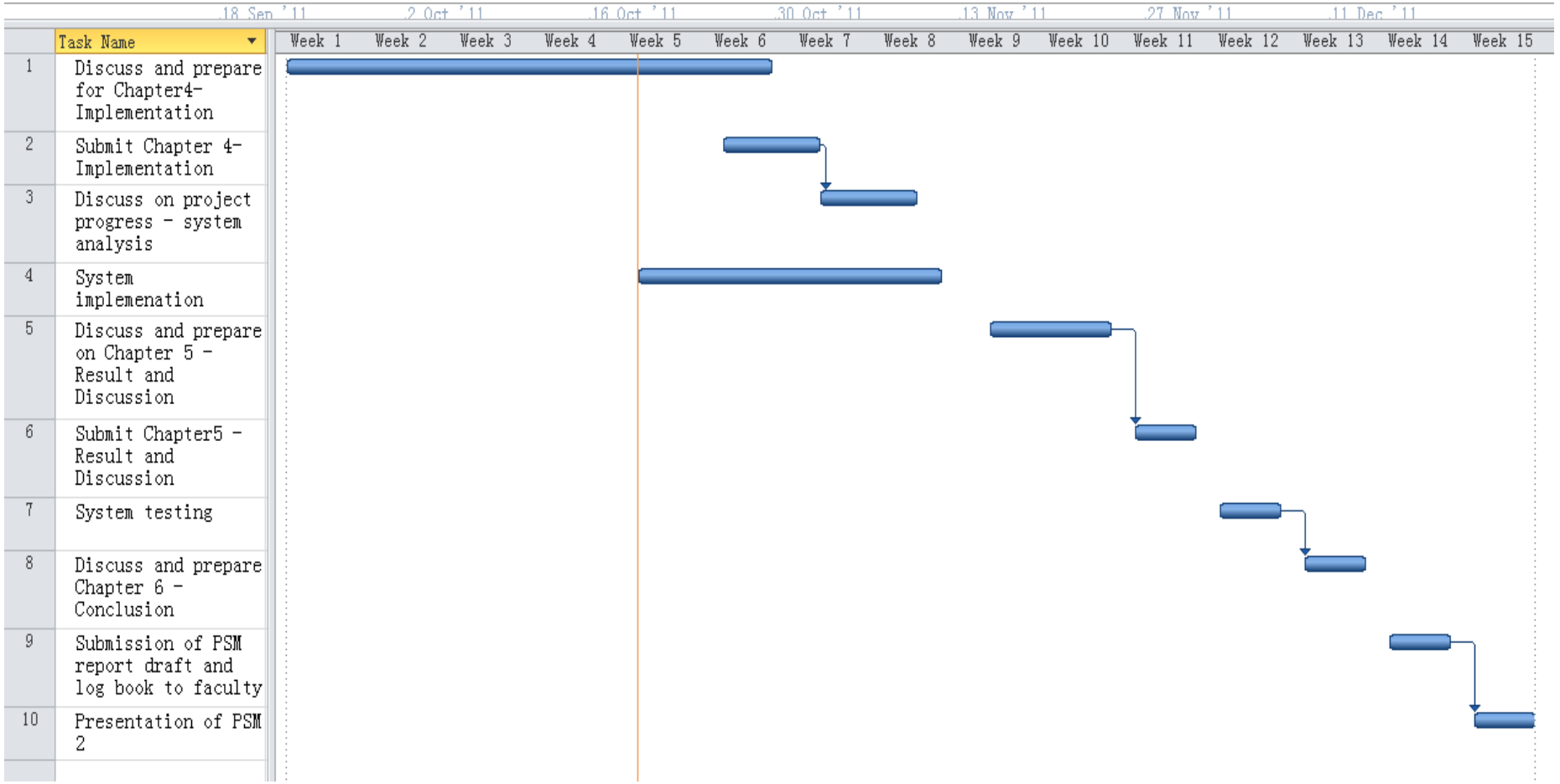
[18] Magnus Erik Hvass Pedersen and Andrew John Chipperfield. *Simplifying Particle Swarm Optimization*. Applied Soft Computing, Volume 10, Issue 2, pp. 618-628, March 2010.

[19] http://www.swarmintelligence.org/ Retrieved 30 October, 2011.

[20] Sigl B., Golub M. and Mornar V. *Solving Timetable Scheduling Problem by using Genetic Algorithms*. Proceedings of the 25th International Conference on Information Technology Interfaces, 2003. ITI 2003, 16-19 June 2003, pp. 519-524.

APPENDIX A - Gantt chart




18 Sep '11
2 Oct '11
16 Oct '11
30 Oct '11
13 Nov '11
27 Nov '11
11 Dec '11
Task Name
Week 1
Week 2
Week 3
Week 4
Week 5
Week 6
Week 7
Week 8
Week 9
Week 10
Week 11
Week 12
Week 13
Week 14
Week 15
1 Discuss and prepare for Chapter4- Implementation
2 Submit Chapter 4- Implementation
3 Discuss on project progress - system analysis
4 System implemenation
5 Discuss and prepare on Chapter 5 - Result and Discussion
6 Submit Chapter5 - Result and Discussion
7 System testing
8 Discuss and prepare Chapter 6 - Conclusion
9 Submission of PSM report draft and log book to faculty
10 Presentation of PSM 2