ABSTRACT

Optimization is defined as a process of finding the best solution in the most effective way of a given problem. Generally, this means solving problems by choosing the values of real or integer variables from a given set of values in order to minimize or maximize a real function. Traveling Salesman Problem (TSP) is a one of the famous optimization problem in finding the shortest distance that passes through a given number of cities (each exactly once) and return to the starting point. This study proposed an enhanced Genetic Algorithm (GA) with Simulated Annealing (SA) which can be implemented and solve TSP. GA may have the advantages in finding a solution in a very short of time when dealing with small dataset. But, GA still needs to overcome its long computation time when handling large number datasets. Besides that, GA lacks in local search ability and sometimes it may have premature convergence. On the other hand, SA often used to find global solution but required a large computation time. The proposed algorithm can speed up the computation time while giving an optimum solution, which is the shortest distance compared to the conventional GA. The proposed method gives a better result in terms of shortest distance and smaller computation time when dealing five datasets from TSPLIB. The results have proved that the proposed method is convincing compared to other related method.

ABSTRAK

Pengoptimumam ditakrifkan sabagai satu proses untuk mencari penyelesaian yang terbaik dengan cara paling berkesan dalam masalah yang diberikan. Secara umum, ini bermakna menyelesaikan masalah dengan memilih nilai-nilai pembolehubah dari satu set nilai untuk meminimumkan atau memaksimumkan sesuatu fungsi nyata. *Traveling Salesman Problem* (TSP) merupakan salah satu masalah pengoptimumam yang terkenal dalam mencari jarak terpendek yang melalui semua bandar-bandar yang diberikan (hanya sekali) dan kembali ke titik awalan. Kajian ini mencadangkan satu method yang menggabungkan Algoritma Genetik (GA) dengan algoritma *Simulated Annealing* (SA) supaya dapat menyelesaikan masalah TSP. Walaupun GA mempunyai kelebihan dalam mencari penyelesaian dalam masa singkat ketika berhadapan dengan dataset yang kecil, GA masih perlu mengatasi masa pengkomputasinya semasa menangani data yang besar. Selain itu, GA tidak memiliki kemampuan untuk pencarian tempatan dan kadang-kadang mempunyai penumpuan prematur. SA sering digunakan untuk mencari penyelesaian global walaupun ia memerlukan masa pengkomputeran yang lama. Algoritma yang dicadangkan dapat mempercepat masa komputasi di samping memberikan penyelesaian yang optimum, iaitu jarak terpendek jika dibandingkan dengan GA konvensional. Kaedah yang dicadangkan memberikan hasil yang lebih baik dalam mendapatkan jarak terpendek dan masa pengkomputeran yang singkat ketika menangani lima dataset dari TSPLIB. Keputusan yang didapati telah membuktikan bahawa kaedah yang dicadangkan adalah lebih baik berbanding dengan kaedah yang lain.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.0 Introduction

During the past decades, the role of optimisation has steadily increased in several diverse areas, such as operations research, electrical engineering, computer science, communication and many more. Linear and non-linear optimisation, the search for an optimum solution, has been seen as major breakthroughs in soft computing. However, many of the large-scale combinatorial optimisation problems are yet to obtain the best solution.

Optimisation is a process that can find the most effective element in all available solutions. It can be either to seek the maximum or minimum of a mathematical problem which involve calculation of real or integer values. For example, we need to find the maximum of a certain function or trying to minimise a cost function. Travelling salesman problem (TSP) is one of the classical optimization problems. It finds the shortest path for visiting N numbers of cities and returning to the starting point. There are lots of techniques that are used to solve TSP.

Thus, many algorithms arise in order to give a better solution. For example, genetic algorithm, particle swarm algorithm, ant colony optimisation, harmony search, intelligent water drops, Tabu search and so on. Two famous methods that used to solve optimisation problems are Genetic Algorithm (GA) and Simulated Annealing (SA).

GA is a heuristic search method that simulates processes using the mechanisms of evolution and natural genetics, and applies it to solve problem. GA retains a population of some feasible solutions for a given problem. The population will goes through evolutionary process in the form of natural selection and natural genetics. The 'good' solution will be maintained in the population whiles the 'bad' solution will be eliminated and replaced with the good offspring in each generation.

Kirkpatrick, Gelett and Vecchi (1983) stated that SA is a probabilistic method which is used to find the global minimum of a cost function that may possess several local minima. It often used when the search space is discrete and be considering as a good algorithm since it is relatively general and not easily stuck in local maximum or minimum.

The basic idea of SA comes from annealing of metals. Physical annealing consist three stages in shaping metals, which is heating to the desired temperature, holding at that particular temperature and cooling. When a metal has been slowly cool down, the metals will forms into a smooth piece as the metal's molecules have come into a crystal structure. However, if the metal has been cool dramatically, rough jagged pieces will be formed and the metal will cover with bumps. The crystal structure represents the minimum energy state (optimal solution) while the jagged and bump pieces represent the local minimums and maximums. The detailed explanation of SA will be discussed in Chapter Two.

However, researchers still cannot decide which one of the techniques used is the best. Therefore, in this dissertation we proposed to enhance Genetic Algorithm with

Simulated Annealing to solve Travelling Salesman Problems and achieve better performance in term of shortest distance and computation time.

1.1 Problem Background

As mentioned earlier, TSP finds the shortest path for visiting *N* numbers of cities and returning to the starting point. It has been shown to be NP-hard while some researchers say it is NP-Complete (Hjertenes, 2002). NP also known to be hard to solve since there have been extensive efforts to find polynomial-time algorithms for problems in NP. It is also the class of problems that a nondeterministic Turing machine accepts in polynomial time. "It is implies that if a problem is NP-Complete there is no polynomial time algorithm for it unless P=NP, although this has not been officially proved yet" (Hjertenes, 2002).

Since the computer now are ever-expanding in speed and memory, the TSP problems less than 100 cities can be solved to optimality within reasonable time (Hjertenes, 2002). Nevertheless, if given a 16-city TSP problem, there are 653,837,184,000 distinct routes that can be generates. Thus, it is much harder if we use a big number of cities. But unless P=NP proves to hold, no polynomial time algorithm is going to be discovered for the TSP.

It is not easy to represent optimization problems. The simplest TSP is to find an optimal solution for visiting *n* cities and returning to the original point, where the inter-city distances are symmetric and known. TSP problems consist of finding a minimal length Hamiltonian circuit in the graph. But if given a cost for the travel destination between the cities, what is the tour with smallest cost?

In order to solve TSP, GA has the advantages in finding a feasible solution in a very short time if the data that used is small or moderate. It can quickly scan a vast solution set. But, if the data that used is large, the computation time is directly proportional to the size of the databases. This means the larger the size of the databases, the longer the time GA needs to produce a solution. Poor solutions do not affect the final solution negatively since they will be discarded, which is very useful for complex or loosely defined problems. However, GA has two other major drawbacks, which is lack of the local search ability and the other is the premature convergence. Premature convergence means the solution obtained is stuck in local optima when the population for an optimization problem is converged too early.

Although SA provide the ease of implementation and the ability to provide reasonably good solutions for many combinatorial problems but it needs a great deal of computer time for many runs and carefully chosen parameters. Besides that, SA has a major disadvantage which is slow convergence. This makes it not suitable for many complex optimisation problems. SA does not choose to climb an “interior” wall (towards better local optima and the global optimum) or an “exterior” wall (towards inferior local optima) when trying to escape from local minima. Although the proof of convergence for SA does not require this insight, the practical (time-based) performance of an SA implementation may be affected.

1.2 Problem Statement

There are some problems that represent the reality situation in GA and SA. GA can find a solution in a very short of time if the database is small. Conversely, the time needed to generate a solution is considerably long if the database is large. Since the optimisation problem is a NP-Hard problem, there is no guarantee any techniques can solve this kind of problem optimally. Thus, the computation time increase whenever the size of the database increases.

On the other hand, SA has the problem of slow convergence. The time taken for obtaining a solution grows exponentially. Somehow, due to the cooling characteristic in SA that eliminates bad solution while cooling, SA can easily jump out of being trapped in local maxima. Thus, enhancing Genetic Algorithm with Simulated Annealing will generate a shortest distance (optimum solution) in a shorten time if GA can adapt the cooling characteristic in SA.

Thus, the hypothesis of this study can be stated as:

Enhanced Genetic Algorithm with Simulated Annealing would reduce computation time in solving Travelling Salesman Problem and gives a better solution, which is the shortest distance compared to Genetic Algorithm.

1.3 Dissertation Aim

The aim of the dissertation is to explore the conventional Genetic Algorithm for Travelling Salesman Problem and enhance it so that it can achieve a better solution (shortest distance) and reduce computation time.

1.4 Objective

The objectives of this dissertation are:

i. To analyses the problem found in GA and the advantages and disadvantages of other techniques that used in finding feasible solution for TSP.
ii. To design and develop the proposed method by enhancing GA with embedded SA so that the performance of GA can be improved.
iii. To analyses the performance of GA, SA and the proposed method in solving TSP.

1.5 Dissertation Scope

The scopes of this dissertation are defined as follows:

i. Experiments will be done for Genetic Algorithm, Simulated Annealing and the proposed method only.
ii. Data that will be used is from benchmark data (TSPLIB, www.iwr.uni-heidelberg.de/sgroups/comopt/software/ TSPLIB95).
iii. A comparison will be done on the computation time and shortest distance that produced by each experiments.

1.6 Significance of the Dissertation

TSP represent of a larger class of problems known as combinatorial optimization problems. Thus, the efficient solution for TSP can also be used to solve others problems in NP-complete class, for example vehicle routing problem. The algorithm proposed can be used in many other important applications in VLSI design, job sequencing, data clustering, production scheduling, curve reconstruction and cutting wallpaper. Proposed Method reduces the disadvantages in GA and adds advantages of SA in this proposed algorithm.

1.7 Organization of Dissertation

In this dissertation, there are consists of five chapters. In the first chapter, it will present the introduction, problems background, problems statement, aim, objective, scope and significant of this study. Chapter 2 will review the literature of this study, which is about SA, GA and other methods that have been used for solve this problem. Chapter 3 will list out the methodology that will use in this study. Chapter 4 will present the experimental result of the study. Chapter 5 is about the conclusion and the finding of this study.