HYBRID WATER FLOW-LIKE ALGORITHM FOR CAPACITATED VEHICLE ROUTING PROBLEM

MOKHTAR MASSOUD MOHAMMED KERWAD

THESIS SUBMITTED IN FULFILMENT FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF INFORMATION SCIENCE AND TECHNOLOGY
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI

2018

ALGORITMA UMPAMA ALIRAN AIR DIPERTINGKAT UNTUK MASALAH PENGHALAAN KENDERAAN TERUPAYA

MOKHTAR MASSOUD MOHAMMED KERWAD

TESIS YANG DIKEMUKAKAN UNTUK MEMPEROLEHI
IJAZAH DOKTOR FALSAFAH

FAKULTI TEKNOLOGI DAN SAINS MAKLUMAT
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI

2018

DECLARATION

I hereby declare that the work in this thesis is my own except for quotations and summaries which have been duly acknowledged.

24 January 2018

MOKHTAR MASSOUD
MOHAMMED KERWAD
P68219

ACKNOWLEDGEMENT

First and foremost, praise and thanks are due to our Almighty Allah, the most gracious and merciful for giving me patience, strength and guidance throughout my journey to complete my study.

I have been truly blessed and honored to have the assistance and guidance of two stellar supervisors who have defined my PhD experience. I take immense pleasure to express my sincere and deep senses of my supervisor Dr Suhaila Zainudin, for her valuable assistance, guidance and creative suggestion throughout the duration of this PhD research. Also, I would like to express my high appreciation to my co-supervisor Associate Professor Dr Zulaiha Ali Othman whose kind contribution and valuable assistance were essential to the completion of this work. She was willing to provide me with whatever I needed until the last moment of my work.

I would like to thank all post graduate students of FTSM, DMO group for their help, friendship, and creating a pleasant working environment throughout my years in UKM.

Most importantly, none of this could have happened without my parents and siblings, who offered their encouragement and love. They taught me that nothing comes easy, and that luck smiles only for those who trust in Allah and work hard. I am forever indebted to them for their constant support and best wishes, which they used to send everyday despite the millions of miles that separate us. I wish to express my unqualified thanks to my wife. I could never have accomplished this dissertation without her love, support, and understanding. I also wish to thank my little heroes Qussay and Mawada for doing their best to understand a father who had to be confined to his study for such a long time, making the FTSM graduate room and libraries his home.

I would, as well, express my appreciation to Dr Ayman I. H. Srour, Dr Hamzah Ali Alkhazaleh Dr Mansour Ali Alssager, and other friends for their insightful remarks, fruitful discussions, and the friendly and optimistic atmosphere that I had with them.

Also, I gratefully acknowledge to people and the government of my country Libya. I could not have gone through the doctoral program overseas without their financial support.

Last but not least, I dedicate this work, to, my family, my father, and my mother soul and ask Almighty Allah to forgive her and be merciful to her.

ABSTRACT

Capacitated Vehicle Routing Problem (CVRP) is widely studied with many applications in different domains. Nevertheless, CVRP still faced ongoing operational challenges. Population-based metaheuristics such as Genetic Algorithm (GA) and Ant Colony Optimization (ACO) have been proven as the most effective solution for CVRP due to its ability in solution diversification but still lacking on solution intensification. GA and ACO rely on a fixed number of solutions which restricts the solution search for a large data set. Therefore, these algorithms have been improved using appropriate searching strategies. Water Flow-like Algorithm (WFA) is a dynamic population-based metaheuristic that was successfully used to solve complex problems such as Bin-Packing Problem, Nurse Scheduling Problems, and Travelling Salesman Problem. WFA demonstrates self-adaptive and dynamic behavior in determining its population size and parameter settings during problem solving. WFA is able to balance between diversification and intensification capabilities for non-heavily constrained problems. It is made up of four components; flow splitting and moving, flow merging, water evaporation, and water precipitation. This research aims to propose WFA for solving CVRP by following three objectives. First, to propose a basic WFA for CVRP (WFA-CVRP), which consider suitable design strategy for solution representation, objective function calculation and iteration number parameter tuning. Second, to enhance WFA for CVRP by three constructive heuristics (random method, nearest neighbor and greedy randomized adaptive search procedure) embedded within precipitation operation to improve WFA exploration capabilities (IWFA-CVRP). Third, to enhance IWFA by hybridizing it with four single based methods (best improvement, first improvement, great deluge, and simulated annealing) to improve IWFA exploitation capability (HIWFA-CVRP). Hybridization between WFA and S-metaheuristics is determined to be effective and efficient in balancing solution diversification and intensification to utilize the superiority of both categories and improve their weaknesses. The experiments were conducted using 55 CVRP benchmark datasets. Basic WFA achieved two best results out of 55 datasets with improved up to 74.5% compared with the state-of-the-art. IWFA obtained two best results out of 55 datasets, with improved up to 76.36% compared with the state-of-the-art. IWFA also scored 34 best results from 55 datasets compared to WFA. Furthermore, HIWFA with great deluge metaheuristic outperformed basic WFA and IWFA in 33 out of 55 datasets, and better than the state of the art in 15 out of 55 datasets, with improvement of 27.27% in term of solutions quality. This indicates that the modifications and the hybridization capabilities of WFA can achieve a balance between intensification and diversification. The result is an effective HIWFA that can be used as a good method for CVRP solution. The results indicate that the proposed algorithm provides competitive results compared with state of the art.

ABSTRAK

Masalah Penghalaan Kenderaan Terupaya (MPKT) telah dikaji dengan meluas dengan aplikasi dalam berbilang domain. Namun, MKPT masih berhadapan dengancabaran operasi berterusan. Metaheuristik berasaskan populasi (Algoritma Genetik (AG) dan Pengoptimuman Koloni Semut (PKS) terbukti sebagai kaedah penyelesaian paling berkesan untuk MPKT kerana kemampuannya dalam kepelbagaian penyelesaian tetapi masih lemah dalam intensifikasi penyelesaian. AG dan PKS bergantung kepada bilangan penyelesaian tetap yang mengekang carian penyelesaian untuk set data besar. Oleh itu, algoritma ini telah dipertingkat menggunakan strategi carian bersesuian. Algoritma umpama-aliran air (UAA) dikenali sebagai metaheuristik berasaskan populasi dinamik yang berjaya digunakan untuk menyelesaikan masalah kompleks seperti Masalah Pembungkusan-Bekas, Penskedulan Jururawat dan Masalah Perjalanan Jurujual. UAA mempamerkan sifat penyesuaian kendiri dan tingkah laku dinamik dalam saiz populasi dan penetapan parameter semasa penyelesaian masalah. UAA berkebolehan untuk mengimbang antara keupayaan kepelbagaian dan intensifikasi untuk masalah tanpa kekangan berat. Ia mempunyai empat komponen; pemisahan aliran dan bergerak, penggabungan aliran, penyejatan air, dan presipitasi air. Kajian ini mencadangkan UAA untuk penyelesaian MPKT menerusi tiga objektif. Pertama, mencadangkan UAA asas untuk MPKT (UAA-MPKT), yang mengambil kira strategi reka bentuk sesuai untuk perwakilan penyelesaian, pengiraan fungsi objektif dan bilangan lelaran penalaan parameter. Kedua, untuk meningkatkan UAA untuk MPKT menerusi tiga heuristik konstruktif (kaedah rawak, jiran terdekat dan prosedur carian penyesuaian rawak tamak) terbenam dalam operasi presipitasi untuk meningkat keupayaan penerokaan UAA (UAAD-MPKT). Ketiga, untuk meningkatkan UAAD dengan hibrid bersama empat kaedah berasaskan tunggal (peningkatan terbaik, peningkatan pertama, banjir besar, dan penyepuhlindapan simulasi) untuk meningkatkan keupayaan eksploitasi UAAD (HUAAD-MPKT). Hibridisasi antara UAA dan S-metaheuristik bertekad untuk menjadi berkesan dan efisien dalam mengimbangi kepelbagaian penyelesaian dan intensifikasi untuk menggunakan keunggulan kedua-dua kategori dan memperbaiki kelemahan mereka. Eksperimen dijalankan menggunakan 55 set data penanda aras MPKT. UAA asas mencapai dua keputusan terbaik daripada 55 set data dengan peningkatan sehingga 74.5% berbanding tahap pencapaian terkini keputusan. UAAD mencapai dua keputusan terbaik daripada 55 set data dengan peningkatan sehingga 76.36% berbanding pencapaian terkini keputusan. Ia turut mencapai skor 34 keputusan terbaik daripada 55 set data berbanding UAA. Tambahan, HUAAD dengan metaheuristik banjir besar mengatasi UAA dan UAAD dalam 33 daripada 55 set data, dan lebih baik dalam tahap pencapaian terkini keputusan bagi 15 daripada 55 set data, dengan peningkatan 27.27% dari segi kualiti penyelesaian. Ini menunjukkan bahawa pengubahsuaian dan penghibridan dalam UAA boleh mengimbangi antara kepelbagaian dan intensifikasi. Hasilnya adalah dengan HUAAD yang berkesan sebagai kaedah yang baik untuk penyelesaian MPKT. Keputusan menunjukkan bahawa algoritma yang dicadangkan memberikan hasil yang kompetitif berbanding tahap pencapaian terkini.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACO	Ant Colony Optimization
ALNS	Adaptive Large Neighborhood Search
BI	Best Improvement
BKS	Best Known Solution
BPP	Bin-Packing Problem
CENTPSO	Combinatorial Expanding Neighborhood Topology Particle Swarm Optimization
COP	Combinatorial Optimization Problem
CS	Cuckoo Search
CVRP	Capacitated Vehicle Routing Problem
FI	First Improvement
GA	Genetic Algorithm
GD	Great Deluge
GRASP	Greedy Randomized Adaptive Search Procedure
HIWFA	Hybrid Improvement Water Flow-like Algorithm
HVRP	Heterogeneous Vehicle Routing Problem
IGA	Improved Genetic Algorithm
IWD	Intelligent Water Drops
IWFA	Improvement Water Flow-like Algorithm
LNS	Large Neighborhood Search
LNS-ACO	Large Neighbourhood Search with Ant Colony Optimization
MTVRP	Multiple Trips Vehicle Routing Problem
NN	Nearest Neighbor
PSO	Particle Swarm Optimization
PSO-VNS	Particle Swarm Optimization within Variable Neighborhood Search
PVRP	Period Vehicle Routing Problem
RABC	Routing Artificial Bee Colony

RM	Random Method
SA	Simulated Annealing
SDVRP	Split Delivery Vehicle Routing Problem
TS	Tabu Search
TSP	Traveling Salesman Problem
VNS	Variable Neighborhood Search
VRP	Vehicle Routing Problem
VRPPD	Vehicle Routing Problem with Pickup and Delivery
VRPTW	Vehicle Routing Problem With Time Windows
WFA	Water Flow-like Algorithm
WFA-CVRP	Water Flow-like Algorithm for CVRP

CHAPTER I

INTRODUCTION

The purpose of this chapter is to discuss the background of approaches mainly involved in solving the Capacitated Vehicle Routing Problem (CVRP) using a Water Flow-like Algorithm (WFA). Section 1.1 presents the background and motivation. Sections 1.2 and 1.3 address the problem statement and research questions, respectively. Section 1.4 discusses the research objectives. Section 1.5 covers the scope of the research. Section 1.6 illustrates the overview of the thesis.

1.1 BACKGROUND AND MOTIVATION

In real life, transportation plays a central role in distributing goods and services (Wang 2013; Amous et al. 2017). Most companies spend a considerable amount of their revenues in distributing their goods when using inefficient transportation systems (Bell & Mcmullen 2004). Studies in North America and Europe shows that using computerized methods in distribution processes saves companies from 5% to 20% in transportation costs (Toth & Vigo 2002). However, even minimal savings in transportation costs may cause a relevant global impact (Ropke 2005; Roberti 2012). Undoubtedly, transportation provides a considerable share of environmental problems (Sathaye et al. 2006; Tahzib & Zvijáková 2012; Adiba et al. 2013). Tahzib and Zvijáková (2012) reported that road transportation is responsible for increased carbon dioxide emissions worldwide by approximately 23% between 1990 and 2010. Currently, companies acknowledge the importance of improving the design of their transportation process, providing high-quality services, and being environmentally friendly at the lowest effort (Qi & Li 2014).

Transportation and distribution problems are generally modeled as Vehicle Routing Problems (VRPs) (Zhang & Tang 2009; Tlili et al. 2014; Hosseinabadi et al. 2017). The VRP introduced by Dantzig and Ramser (1959) is one of the classical Combinatorial Optimization Problems (COPs). Moreover, VRP plays an important role in reducing transportation cost while satisfying routing constraints and orders of customers (Yousefikhoshbakht & Khorram 2012; Booyavi et al. 2014), VRP is categorized as nondeterministic polynomial time (NP) by combinatorial theory (Talbi 2009; Tavakkoli-Moghaddam et al. 2012). It has different classes of additional practical constraints introduced in the literature, such as capacity restricted vehicles known as Capacitated VRP (CVRP) (Huang & Ding 2013; Teymourian et al. 2016). VRP with Time Windows (VRPTW) (Yassen et al. 2015), a VRP with Pickup and Delivery (VRPPD) (Li & Lim 2003), and a VRP with Split Delivery (SDVRP) (Archetti et al. 2006). However, one of the extensively investigated VRPs is CVRP (Zhang et al. 2015; Teymourian et al. 2016) (Zhang et al. 2015; Teymourian et al. 2016) because it has many real-world applications, such as household waste collection, gasoline delivery, goods distribution trucks, and mail delivery (Yeun et al. 2008; Adiba et al. 2013). Therefore, this research focuses on the CVRP.

Various types of methods, such as exact and heuristic have been used to solve NP-hard problems (Ropke 2005; Balaprakash 2010). However, exact algorithms fail to obtain an optimal solution because of the computational time required (Talbi 2009; Yousefikhoshbakht et al. 2014). Heuristic algorithms such as nearest neighbor cannot find high-quality solutions. Moreover, these algorithms are usually embedded into the upper-level heuristic approach, called metaheuristics, to derive a quick initial solution, and the generated solution obtained is typically worse in heuristic algorithms than in metaheuristics (Dorigo & Stützle 2010). Nevertheless, metaheuristics tackle several COPs successfully within an acceptable amount of time (Talbi 2009). The use of metaheuristics to solve NP-hard problems has recently attracted increasing attention because of the success achieved when tackling many COPs (Vidal et al. 2013). Metaheuristics can be obtained from combining various concepts derived from heuristic, nature-inspired, artificial intelligence, and biological phenomenon methods (Chen & Ting 2006). These methods are used to find good quality solutions within a practical time but does not necessarily obtain the optimal solution (Tan et al. 2012).

Metaheuristics are mainly classified into two categories; single solution-based (S-metaheuristics) and population-based metaheuristics (P-metaheuristics) (Blum & Roli 2003; Talbi 2009). S-metaheuristics include simulated annealing (SA) (Kirkpatrick et al. 1983; Van Breedam 1995; Harmanani et al. 2011), and greedy randomized adaptive search procedure (GRASP) (Feo & Resende 1989; Layeb et al. 2013). On the other hand, P-metaheuristics include genetic algorithm (GA) (Baker & Ayechew 2003; Jie-Sheng et al. 2011), ant colony optimization (ACO) (Mazzeo & Loiseau 2004; Huang & Ding 2013), and a cuckoo search (CS) (Xiao et al. 2017). Each category has its own characteristics in solving problems. S-metaheuristics are easy to implement and systematically exploit the solution search space to obtain a good solution in a short time (Rabadi 2016; Talbi et al. 2016). However, the possibility of becoming trapped in the local optima is high and is considered as the weakness of S-metaheuristics because it focuses on exploiting rather than exploring (Talbi 2009; Rabadi 2016). P-metaheuristics focus on exploring by searching inside the space of the solution set to improve the efficiency and strength of problem space exploration. However, P-metaheuristics have weaknesses; for example, rapid convergence and redundant searches consume extra time when P-metaheuristics focus on exploration rather than exploitation (Yang & Wang 2007; Talbi 2009; Rabadi 2016).

The combination of different metaheuristics is widespread (i.e., hybrid metaheuristic) (Blum et al. 2011; Baghel et al. 2012). In addition, metaheuristic hybridization concept has been extensively studied and used by the operational research community due to its improved capabilities. These improvements come from utilizing the strength of other metaheuristic components and by exploiting the complementary character of different optimization strategies. Therefore, having an adequate combination of complementary algorithmic concepts can be the key for achieving top performance in solving many hard combinatorial optimization problems (Raidl 2006; Blum et al. 2011). However, one of the major key usage of such combination is the fact that it can provide a good balance between exploration and exploitation (Birattari et al. 2001; Lozano & García-Martínez 2010). Hybridization between S- and P-metaheuristics is determined to be effective and efficient in balancing solution exploration and exploitation to utilize the superiority of both categories and improve their weaknesses (Blum & Roli 2008; Zapfel et al. 2010). S-metaheuristics supports search exploitation on the solution search space in P-metaheuristics when these two categories are combined, whereas P-metaheuristics support search exploration in S-metaheuristics (Blum & Roli 2003, 2008). This hybridization can create the right balance between exploration and exploitation and improve the performance of a good solution for a given problem (Blum et al. 2011b; Talbi 2009). However, these metaheuristics are designed based on a fixed-sized population number and are not adequately intelligent to perform an efficient solution search (Yang & Wang 2007; Wu et al. 2010; Shahnazari-Shahrezaei et al. 2011). These metaheuristics suffer from redundant solution search, which causes unnecessary extra computational costs to the algorithm during the optimization process. Moreover, the conventional metaheuristics also lack self-adaptive or dynamic parameter tuning in terms of population size. The use of a unique or manually assigned number of the population size of metaheuristics for variant COPs is infeasible (Stützle et al. 2011).

A relatively new metaheuristic named as a water flow like algorithm from the nature-inspired family, which is self-adaptive and dynamic in its population sizes and parameter settings, has recently emerged in the literature (Yang & Wang 2007). Nonetheless, it has yet to be applied to the VRPs so far. Therefore, this work proposes WFA for solving the CVRP. The characteristic of using the WFA for solving CVRP are:

- WFA is dynamic in addressing population size by splitting and merging operation, whereby the number of solutions can increase or decrease during the optimization process.

- WFA can use the important features of metaheuristic algorithms by being self-adaptive in addressing the other parameters during the algorithm iterative searching process.

- WFA has the ability to escape from local optima by using evaporation operation.

- To avoid redundant searches, WFA reduces the number of solutions when multiple solutions move to the same location (objective value); resource is wasted in unavoidable redundant searches.

- WFA is characterized as being simple and flexible, so it motivates scholars to conduct modifications to improve its performance.

- Moreover, WFA has been successfully used to solve several COPs, such as bin-packing problem (Yang & Wang 2007), manufacturing cell-fraction problem (Wu et al. 2010), nurse scheduling (Shahnazari-Shahrezaei et al. 2011), hybrid flow-shop scheduling problem (Pargar & Zandieh 2012), fuzzy inference system (Kuo & Lee 2015) and traveling salesman problem (TSP) (Srour et al. 2014; Bostamam & Othman 2016).

Despite WFA's stronger ability to find quality solutions compared to other metaheuristic such as GA and ACO, it has limitations due to its search strategy in exploring the search space (Othman et al. 2013). Thus, researcher tried to speed up the convergence rate by hybridizing , which is commonly arises in many metaheuristic (Talbi 2009), and balancing between the exploration and exploitation which is reported by Bostamam and Othman (2016) and Othman et al. (2017).

In the CVRP, a set of vehicles of the same capacity located at a central depot require will be routed to serve a set of customers with known demands. Each customer is visited exactly once and by only one vehicle, and each route starts and ends at the depot. The total demand of any route should not exceed the capacity of the vehicle. The CVRP has attracted considerable attention in the operational research and artificial intelligence community. Apart from the difficulty in solving the problem, the CVRP has been selected for two reasons, that is, the CVRP represents various real-world applications and its results can still be improved. Moreover, the WFA is used to solve the CVRP for the first time.

1.2 PROBLEM STATEMENT

The CVRP is considered one of the most important optimization problems in transportation and distribution systems (Lin et al. 2009; Sze et al. 2017). It is described as a set of vehicles used to serve a number of customers with identified demands. Each vehicle has the uniform capacity that starts and ends at the same depot. Each customer is serviced exactly by one vehicle, and all of the customers must be assigned to vehicles while satisfying the requirements of all customers without violating the capacity constraint of the vehicles; aims to minimize total raveling distances (Toth & Vigo 2002).

CVRP is addressed by many current metaheuristics such as GA, ACO, and CS that can provide satisfactory results for different types of optimization problems (Wink et al. 2012), but finding an optimal solution is uncertain (Olafsson 2006; Srour et al. 2014; Mohammed, Ghani, et al. 2017). These metaheuristics suffer from several deficiencies; have a complex structure and are not easily enhanced. Others lack balance between intensification and diversification. Furthermore, these metaheuristics suffer from a fixed-sized population number, which is insufficient to perform an efficient solution search. They also suffer from their inability to consider the search status in tuning their parameters during the search because they fix them in advance regardless of any changes that may occur during the search (Wu et al. 2010; Shahnazari-Shahrezaei et al. 2011; Srour 2014). The optimal values for the parameters depend mainly on the problem and even the instance to be handled (Talbi 2009).

Owing to the importance of the CVRP and the inability of existing approaches to work well across available instances, the requirement to propose a new metaheuristic, which has the ability to work well with available instances, remains urgent. Therefore, this motivate us to move forward and investigate other algorithms that have not been utilized for solving the CVRP. Although the WFA has positive characteristic, referred in (Section 1.1), it has not been used for solving the CVRP. Hence, this research focus on investigating the capability of WFA in solving the CVRP. The research question that arises in this research is, "*Can the WFA tackle the CVRP? If so, what is the suitable design strategy that may include solution representation, objective function calculation, and parameter tuning?*"

WFA has several operations, namely, splitting, moving, merging, evaporation, and precipitation. Several scholars have addressed other drawbacks, particularly in the precipitation operation (Lee & Kuo 2012; Kuo & Lee 2015), whereby the same exact solution is being duplicated. However, this observation implies that the WFA has a significant chance of being trapped in the local optima because of the lack of solution diversification. The purpose of this operation is to explore solutions of unvisited region through the precipitation operation (Chang & Wu 2011). The enhancement of the precipitation operation can increase the solution diversity to make the process of finding the global minimum efficient (Kuo & Lee 2015). This research proposes three constructive heuristics, namely, random method (RM), nearest neighbor (NN), and GRASP, to construct different quality solutions from scratch and to enhance the WFA ability to explore new favorable areas in the solution search space and reduce the chance of becoming trapped in the local optima to overcome the abovementioned drawbacks. The research question is, "*What is the right constructive heuristic to construct an initial population instead of duplicating the same current solutions that can be enhanced in the WFA exploration to sufficiently cover the search space*?"

A metaheuristic can be successful for a given optimization problem if it can balance between diversification and intensification (Blum & Roli 2003). Hybridization between S- and P-metaheuristics is determined to be effective and efficient in balancing solution diversification and intensification to utilize the superiority of both categories and improve their weaknesses (Blum & Roli 2008; Zapfel et al. 2010). S-metaheuristics supports search exploitation on the solution search space in P-metaheuristics when these two categories are combined, whereas P-metaheuristics support search exploration in S-metaheuristics (Blum & Roli 2003, 2008). Bostamam and Othman (2016) and Othman et al. (2017) reported that the WFA lacks balance between diversification and intensification that can cause delayed convergence. Several scholars mentioned that the WFA exhibits slow convergence at the beginning of the search because the search is started using a single solution (Yang & Wang 2007; Lee & Kuo 2012). However, the WFA lacks intensification; thus, the balance between intensification and diversification could be improved by hybridizing WFA with local search (Bostamam & Othman 2016; Othman et al. 2017). Thus, in this research, a hybrid WFA with four local search, namely, best improvement (BI), first improvement (FI), great deluge (GD), and SA, is proposed to overcome aforementioned problem. The research question states, "*What is the right local search method to straighten the WFA intensification capability, which leads to an enhanced solution quality?*"

1.3 RESEARCH OBJECTIVES

The main aim of this thesis is to propose the WFA for solving the CVRP by utilizing its strengths and compensating for any weakness by hybridizing with other metaheuristic algorithms and to attain a suitable balance between exploration and exploitation. The following key objectives have been identified to achieve these objectives:

i. To design the WFA for solving the CVRP, which considers a suitable design strategy for solution representation, objective function calculation, and iteration number parameter tuning.

ii. To enhance WFA by embedded constructive heuristics into the precipitation mechanism to improve the WFA solution diversification to search in wide regions of the solution search space, thereby preventing the WFA from being trapped in the local optima.

iii. To hybridize the WFA with other S-metaheuristics to enhance the WFA solution intensification to attain a suitable balance between diversification and intensification of the solution search space.

1.4 RESEARCH SCOPE

This research focuses on proposing and developing the WFA for solving the CVRP to improve the quality of vehicle paths (i.e., minimize the total traveling distance). This research focuses on the CVRP because of the NP-hard nature of the problem. The CVRP represents the core problem of all VRPs and the significance of the problem to real-life applications. Furthermore, the CVRP models involve many real-life problems encountered in the physical distribution of goods and appear in many practical situations, such as collecting mails from mailboxes, picking up children by school buses, and delivering gasoline to gas stations (Roberti 2012). Therefore, solving this problem has a positive effect on these applications, especially in the transportation section and distribution systems.

The WFA has recently been categorized as a population-based algorithm and has been proven to be an influential method for solving many COPs. This research aims to enhance the WFA ability by conducting several modifications. Figure 1.1 illustrates the scope of this research. The proposed method is tested on 55 CVRP standard benchmark datasets, which consist of numerous vehicles to serve a number of customers ranging from 22 to 261. The benchmark dataset was downloaded from http://vrp.atd-lab.inf.puc-rio.br/. The WFA deals only with feasible solutions, which satisfy all of the problem constraints. The results are compared with each other and with the results of the state-of-the-art approaches reported in the literature.

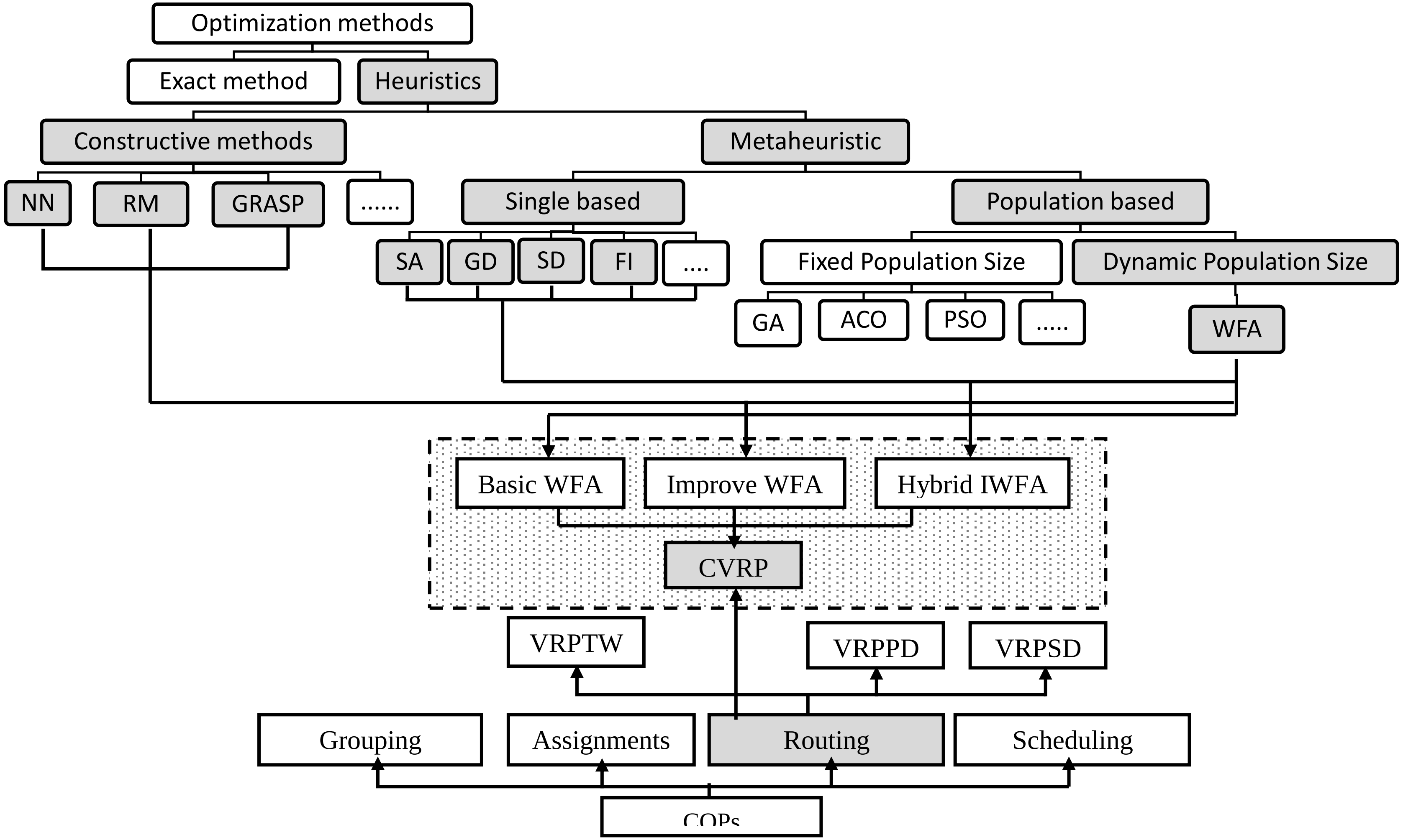

Figure 1.1 Scope of the research

1.5 THESIS OUTLINE

This thesis comprises eight chapters, including the introductory chapter (as depicted in Figure 1.2), organized as follows.

Chapter I provides an overview of the research, such as the background and motivation, problem statement, research questions, objectives and contributions, and research scope.

Chapter II presents a brief review of the VRP and its important extensions. Then, the CVRP, which is the focus of this thesis, and the algorithms used to tackle the problem are reviewed. These algorithms can be categorized into exact, heuristic, and metaheuristic algorithms (i.e., S- and P-metaheuristics). In addition, the explanation and description of the fundamentals, concepts, and operations and previously published studies on WFA are provided. This chapter also reviews the state-of-the-art methods that are concerned with WFA applications and improvements.

Chapter III illustrates the research methodology used in this thesis. It consists of five phases, namely, identification of the problem domain, preprocessing, constructive algorithm, improvement algorithm, and evaluation and comparison phases.

Chapter IV explains the development of the WFA for solving the CVRP. The basic principles and fundamentals of the proposed algorithm are discussed. The chapter starts with an illustration and description of the operations of the WFA for solving the CVRP, including the representation of the CVRP. This chapter also describes the initialization components, flow moving, and the water presentation mechanisms, which are developed for the problem. Then, the chapter shows the process of the experimental tests that have been performed to evaluate the proposed algorithm. The results and analysis of the experiments and the statistical studies are provided in this chapter. The analysis and statistical evaluations are conducted to test the performance, scalability, and efficiency of the tested metaheuristics using the CVRP benchmark datasets.

Chapter V clarifies the exploration capability of the WFA using three constructive heuristics, namely, RM, NN greedy heuristic, and GRASP, which are used to build an initial population of diverse solutions. These algorithms are separately combined with the basic WFA and called the improved WFA (IWFA). The result of the IWFA is compared with the result of the basic WFA. A diversity measurement mechanism of the CVRP solutions is also described in this chapter. The aim of the preliminary experiment is to analyze these methods in terms of solution quality and diversity and to propose a method to measure the diversity of the solutions in the population.

Chapter VI expresses the hybridization of the IWFA with other S-metaheuristics to further enhance the quality of the solution. Four algorithms, namely, BI, FI, GD, and SA, are considered in this chapter. These algorithms are separately combined with the IWFA and then evaluated. The S-metaheuristic is used to intensify the search process that balances between the search exploration and exploitation in the WFA. Thus, favorable solutions can be obtained.

Chapter VII shows the analysis and evaluation of the proposed algorithm on the basis of the results obtained. The results are compared with those of the other available approaches in the literature used to solve the CVRPs.

Chapter VIII concludes this thesis by summarizing the findings and contributions of this research and highlights the recommendations for future research.

Chapter I	• Introduction
Chapter II	• Literature review
Chapter III	• Research methodology
Chapter IV	• Objective 1 • Research question 1
Chapter V	• Objective 2 • Research question 2
Chapter VI	• Objective 3 • Research question 3
Chapter VII	• Evaluation against the state-of-the-art methods
Chapter VIII	•Conclusion and future work

Figure 1.2 Structure of the thesis

CHAPTER II

LITERATURE REVIEW

2.1 INTRODUCTION

This chapter provides a review of available algorithms that have been discussed in literature and applied to solve CVRP. It also provides the definition of CVRP, with the standard benchmark dataset used in this study. An overview of the WFA algorithm is presented in this chapter.

The definition of VRP and some VRP variants are mainly presented in Sections 2.2. Section 2.3 describes CVRP with its specification and formulation. Section 2.4 discusses several well-known algorithms applied to CVRP. Then, an overview of the WFA algorithm is presented in Section 2.5. Finally, this chapter is summarized in Section 2.6.

2.2 VEHICLE ROUTING PROBLEM

The vehicle routing problem was introduced by Dantzig and Ramser (1959) as the truck dispatching problem. It is categorized as an NP-hard combinatorial optimization problem (Lenstra & Kan 1981; Yousefikhoshbakht et al. 2014). VRP was formulated as a complex extension of TSP with salesmen (Yeun et al. 2008), which have different routes and each vehicle with a specific route. The vehicle routing problem is one of the most significant and widely studied problems because of its application in distribution systems and the transportation industry (Nagata & Bräysy 2009; Yousefikhoshbakht & Khorram 2012). VRP searches for a number of vehicle routes that can serve a number of customers with the least cost (minimum traveling distances) (Dantzig & Ramser 1959). The capacitated vehicle routing problem represents the traditional and common extension of VRP (Teymourian et al. 2016), in which a vehicle capacity constraint exists. Consequently, the total demands of all customers in each route should not exceed the capacity of the vehicle. In CVRP, the fleet of vehicles is homogeneous, i.e., all vehicles have similar capacities (Vidal et al. 2013; Booyavi et al. 2014).

Due to the fast growth of real world applications with increased requirements, and to make VRP models more realistic and applicable, a wide variety of the VRP exists by adding different constraints to the basic model (Yousefikhoshbakht et al. 2014; Sze et al. 2017). The rest of this section presents an overview of some variants of VRPs:

a. Split-delivery Vehicle Routing Problem (SDVRP)

The split delivery VRP represents the extension of VRP in which the demand of each customer can be greater than the vehicle capacity. In this variant, to allow the split of customer demand, the constraint of visiting the customer exactly one time by vehicle should be canceled, thus each customer can be visited more than once until demand is fulfilled (Archetti et al. 2006).

b. Vehicle Routing Problem with Time Windows (VRPTW)

Vehicle routing problem with time windows can be described as a set of routes designed in such a way that each point is visited only once by exactly one vehicle within a given time interval (*service time*). All routes start and end at the depot, and the total demands of all points on one particular route must not exceed the capacity of the vehicle (Bräysy & Gendreau 2005).

c. Heterogeneous Vehicle Routing Problem (HVRP)

The heterogeneous VRP is a variant of CVRP with a depot that use different vehicle types, i.e., the vehicles of each type have a specific capacity. Thus, the HVRP solution consists of multiple routes and each one is associated with the type of vehicle (Li et al. 2007).

d. Period Vehicle Routing Problem (PVRP)

The period VRP deals with planning for vehicle routes over a multi-day period. In this extension, any customer may be visited multiple times and these visits are organized based on an allowable combinations set of distribution days (Francis et al. 2008).

e. Multiple-trips Vehicle Routing Problem (MTVRP)

In multiple trips VRP the vehicle can go to the depot many times to load and unload the goods called a schedule, with the total duration not exceeding a maximum driving time (Vidal et al. 2013).

2.3 CAPACITATED VEHICLE ROUTING PROBLEM

Capacitated vehicle routing problem is a COP that has received considerable attention (Zhou et al. 2013; Qi & Li 2014). CVRP structure is a mix of two NP hard problems namely TSP and bin packing problem (BPP) (Vidal et al. 2013). TSP is about a travelling salesman who wants to visit a number of cities and visit each city exactly once, starting and ending at the same city. BPP can be described as follows: Given *n* items with its weight and *n* bins with capacity of each bin, the target is to assign each item to one bin so that the total weight of the items in each bin should not exceed the capacity, at the same time use the minimum number of bins (Falkenauer 1996). This condition can be related to CVRP, and specific demand of customers can be assigned to vehicles by solving BPP, while TSP aims to find the best route for each vehicle, and the least costly sequence of visits for the customer assigned to it.

CVRP has received much attention and become a more interesting research area because of its real application in transportation and its economic importance in reducing operational costs in distribution systems (Alabas-Uslu & Dengiz 2011; Yousefikhoshbakht & Khorram 2012; Hosseinabadi et al. 2017), thereby making it an interesting subject for computer scientists (Abdulmajeed & Ayob 2014). Furthermore, achieving an optimal solution with traditional optimization methods is not easy because of the high computational complexity of large-scale problems (Kır et al. 2017).

Therefore, this study focuses on CVRP and can be defined as the process of designing a least cost set of routes to serve a set of customers in such a way that: each vehicle (route) starts and ends at the depot, the total demand of each route does not exceed the vehicle capacity, and each customer is visited exactly once by exactly one vehicle (Toth & Vigo 2002; Vidal et al. 2013). Figure 2.1 shows the graph of the CVRP solution, in which the number 0 represents the depot, the numbers 1 to 8 represent the customers, and each complete circle of route represents one vehicle.

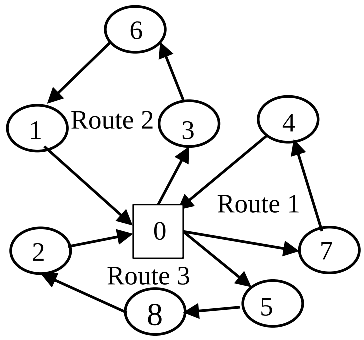

Figure 2.1 A graph representation of CVRP solution with 8 customers and 3 vehicles

2.3.1 CVRP Formulation

The CVRP is described as the graph theoretic problem: let $G = (V, E)$ be a complete and undirected graph where V is the vertex set and E is the edge set. Vertex set $V = (0, 1, 2, \ldots, n)$ corresponds to n customers, whereas vertex 0 corresponds to the depot. A fleet of k identical vehicles of capacity Q is based in the depot, and each customer i has a non-negative demand qi. Its objective is to find the optimal routes for distributing various items between customers and depot by a fleet of vehicles at minimal traveling distances with the following summarized constraints:

- Each vehicle starts and ends its route at the depot.
- Each customer is served exactly once by one vehicle.
- The customer's total demand must not exceed the vehicle capacity

Many different formulations and model for CVRP can be found in the literature (Toth & Vigo 2002; Kara et al. 2004). In the present thesis, the mathematical model for CVRP is follow (Yousefikhoshbakht & Khorram 2012).

The variables are defined follows:

Dis = total distance travelled by all vehicles.

$$x_{ijs} = \begin{Bmatrix} 1, & vehicle\ s\ departs\ from\ i\ to\ j \\ 0, & otherwise \end{Bmatrix}$$

$$y_{is} = \begin{Bmatrix} 1, & customer\ i\ is\ served\ by\ vehicle\ s \\ 0, & otherwise \end{Bmatrix}$$

Coefficients:

c_{ij} : cost from customer i to customer j
q_i : demand of customers i (i = 1, 2, 3..., n)
n : total number of customers
k : total number of vehicles
Q_k : capacity of k^{th} vehicle
s : vehicle number (1, 2, 3 ..., k)
Y_{is}: binary variable: its value is 1 if the customer i is delivered by the vehicle s; otherwise it is 0.
X_{ijs}: binary variable: its value is 1 if the vehicle s travels directly from customer i to customer j, otherwise it is 0.

The objective function is:

$$Minimize \quad dis = \sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{s=1}^{k} c_{ij}x_{ijs} \tag{2.1}$$

Subject to

$$\sum_{i=1}^{n} q_i\, y_{is} \le Q_k, \quad s = 1, \dots, k \tag{2.2}$$

$$\sum_{i=0}^{n} x_{ijs} = y_{js}, \qquad j = 1,2,\dots,n; \;\; s = 1,2,\dots,k \tag{2.3}$$

$$\sum_{j=0}^{n} x_{ijs} = y_{is}, \qquad i = 1,2,\dots,n; \;\; s = 1,2,\dots,k \tag{2.4}$$

$$\sum_{s=1}^{k} y_{is} = \begin{cases} 1, & i = 1,2,\dots n \\ k, & i = 0 \end{cases} \tag{2.5}$$

Equation (2.1) is the objective function of the problem to minimize the traveling distance. Equation (2.2) avoids exceeding capacity of each vehicle. Equation (2.3) and Equation (2.4) ensure that does not exceed the maximum number of vehicles. Equation (2.5) guarantees that each customer is served exactly by one vehicle.

2.3.2 Solution Representation

The solution is presented as a one-dimensional vector, where the numbers 1 to 8 represent the customers, and 0 indicates the depot. The length of the solution represents the total number of customers. Each solution has a number of routes that can be counted based on the total number of 0 minus 1. The example of the solution representation, as shown in Figure 2.2, has three routes. The first route serves two customers (7 and 4), the second serves three customers (5, 8 and 2), and the third route serves three customers (3, 6 and 1).

Route 1	0	7	4	0	
Route 2	0	5	8	2	0
Route 3	0	3	6	1	0

0	7	4	0	5	8	2	0	3	6	1	0

Figure 2.2 Solution representation

The initial solution is generated by randomly selecting customers for the current route without violating the capacity of the vehicle. If a violation occurs, a new route is created and this procedure is repeat until all customers are routed.

2.4 RELATED WORKS TO CAPACITATED VEHICLE ROUTING PROBLEM

Researchers from various fields have been motivated to solve the CVRP and proposed many approaches (Kumar & Panneerselvam 2012; Zhou et al. 2013). These solutions can be perceived as one of two types: exact methods, and heuristic methods (Szeto et al. 2011).

Even though exact methods are able to achieve optimal solutions for CVRP they are recommended only to deal with problems of small size (Zhang & Tang 2009). Owing to the growth in problem size and the fact that CVRP is an NP-hard problem, exact methods are not applicable because they are time consuming and only suitable for small-size problems (Talbi 2009; Teymourian et al. 2016). Therefore, researchers have proposed heuristic methods that are proven to generate good-quality solutions within a reasonable amount of time but do not guarantee optimal solutions (Talbi 2009; Tan et al. 2012). The proposed heuristic methods can be divided into two traditional approaches: constructive and metaheuristic (Bräysy & Gendreau 2005; Tan et al. 2012). Constructive heuristics aims to construct an initial feasible solution from scratch (Blum & Roli 2003) whereas metaheuristic approaches can be divided into two classes: single-solution based (S-metaheuristics) and population-based (P- metaheuristics). S-metaheuristics represent a general class of improving heuristics that aim to improve the quality of a solution by exploring its neighborhoods. P-metaheuristics aim to obtain good solutions by dealing with a population of solutions instead of a single solution (Laporte & Semet 2002; Blum & Roli 2003; Dorigo & Stützle 2010).

2.4.1 Exact Algorithm

Exact methods have the ability to obtain the best solutions and guarantee their optimality (Talbi 2009). Numerous exact methods have been proposed to handle CVRP. These methods include dynamic programming (Christofides et al. 1981), branch and bound (Toth & Vigo 2001) and branch and cut (Mitchell 2002; Naddef & Rinaldi 2002). although exact algorithms can obtain optimal solutions with guaranteed optimality for small problem size, they are unsuitable for large VRP problems because of the required computational time which rises exponentially with problem size (Kytöjoki et al. 2007;

Puljić 2012; Yousefikhoshbakht & Khorram 2012; Akpinar 2016). Therefore, many real-world practitioners would usually prefer satisfactory solutions in competitive time rather than the optimal solution that takes years to achieve.(Weise et al. 2009; Gendreau & Potvin 2010). Additional details on various exact algorithms have been offered by Larsen (1999).

2.4.2 Constructive Heuristics

Constructive heuristics are applied extensively to combinatorial optimization problems. These algorithms gradually initialize the feasible solution from scratch, and then insert the solution components iteratively without violating the route constraints and without backtracking until the solution is completely initialized while considering the feasibility of the solution (Laporte & Semet 2002; Dorigo & Stützle 2010). The advantages of these heuristics are their simple implementation and efficiency in finding good diverse solutions (Layeb et al. 2013). However, compared with metaheuristic algorithms, constructive heuristics generate low-quality solutions (Talbi 2009). Numerous types of constructive heuristics have been reported in the literature, such as NN (Wink et al. 2012; Xiao & Jiang-Qing 2012), saving heuristics by Clarke and Wright (Juan et al. 2010), and RM (Brajevic 2011; Akpinar 2016)

Several types of constructive heuristics have been used to construct the initial solution for CVRP. The following sections concentrate on reviewing some constructive heuristics in the literature.

RM is the most common heuristic to construct the initial solution in many COPs (Alkhazaleh et al. 2013). Several researchers (Szeto et al. 2011; Hsu et al. 2014; Akpinar 2016; Mazidi et al. 2016) have used RM to generate the initial solution for CVRP. Every route is randomly initialized, i.e., customers are randomly inserted one by one into the current route. Then, the current route is added into the solution after deleting any customer that is inconsistent with the problem restrictions. To serve the remaining un-routed customers, the processes of creating and adding new routes are repeated until a feasible initial solution is generated.

NN is a popular tour construction heuristic that has been used in many fields since its conception in the early 1950s (Gutin et al. 2002; Alkhazaleh et al. 2013). This algorithm starts at an arbitrary point, and then selects, from among the unsequenced nodes, the node that is closest to the last point that was inserted into the current route (Bentley 1992). The selected point is inserted at the end of that route, while this point is not a contrary problem constraint. The process of selecting and inserting unrouted points is repeated until no further points can be inserted into the current route. Finally, this heuristic continues by constructing new routes until all customers are routed. By using different starting nodes, various final solutions can be generated (Talbi 2009).

Zheng et al. (2013) and Zhou et al. (2013) utilized GRASP to construct the solutions for CVRP. This procedure is a multi-start method in which each iteration consists of a construction and local search phase. In the first phase, a random greedy solution is constructed. Then, the constructed solution is improved by a local search until the local optimal solution is found. In the construction phase in general, consists of creating a list that includes the candidate element that can be inserted into the partial solution while maintaining feasibility. In this phase, one element at a time is iteratively constructed for a feasible solution. At each iteration, the next element that can be inserted into the partial solution is determined by ordering all the elements in a candidate list. Through a greedy function, the incremental cost is evaluated for each candidate element. The candidates are ordered based on their greedy value in a restricted candidate list. At each iteration, one candidate element is randomly selected from the restricted candidate list and added to the partial solution (this step denotes the probabilistic aspect of the heuristic). The restricted candidate list is updated once the element has been added to the partial solution. This update is performed by reevaluating the greedy value of the unvisited candidate elements; this is the adaptive aspect of the heuristic (Feo & Resende 1989; Resende & Ribeiro 2008; Chen et al. 2012).

To summarize, several constructive heuristics have been used to construct the initial solution for CVRP. The type of method used in the construction phase in the metaheuristic algorithm plays an important role in ensuring the effectiveness and efficiency of the algorithm. Based on the preceding discussion, using improved solutions as initial solutions will not often improve the local optima. In this study, we use three constructive heuristics that are most frequently used to construct solutions for CVRP; these heuristics are RM (Brajevic 2011; Qin & Yi 2011; Szeto et al. 2011; Zhang & Lee 2015; Akpinar 2016; Mazidi et al. 2016; Mohammed, Gani, et al. 2017), the random strategy may generate a high deviation in terms of the obtained solutions by choose the customer randomly and add it to the route. NN (Chen & Ting 2006; Du & He 2012; Xiao & Jiang-Qing 2012; Chen, Chang, et al. 2015; Vincent et al. 2017), this heuristic choose starting point "customer" randomly, and then selects, the customer that is closest to the last customer that was inserted into the current route aims to reduce the solution distance. Hence, *n* different solutions may be obtained with NN. GRASP (Resende & Ribeiro 2008; Suárez & Anticona 2010; Chen et al. 2012; Zheng et al. 2013; Zhou et al. 2013), this heuristic create a list that includes the candidate customers that can be inserted in the partial solution with keeping the feasibility. Each candidate customer in the list has been evaluated by a greedy function, which make the diversity to the solution. Thus, the following research question is asked: *What is the suitable constructive heuristic to use in constructing the solution that covers as much of the search space as possible?*

2.4.3 Metaheuristics

Metaheuristics are general algorithmic frameworks that have been exhaustively employed over the past decades to address hard combinatorial optimization problems. These algorithmic frameworks can solve hard and large-scale problems efficiently and effectively by gaining near-optimal solutions within a short amount of time. These frameworks can also effectively handle a wide range of constraints (Blum & Roli 2003, 2008). Metaheuristics can be built by merging various concepts extracted from heuristic methods, artificial intelligence, and nature-inspired and biological evaluations (Osman & Laporte 1996).

The term "metaheuristic" was first introduced by Glover (1986). Osman and Kelly (1996) formally define metaheuristics as *"an iterative generation process which guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space[;] learning strategies are used to structure* *information in order to find efficiently near-optimal solutions"*. Similar definitions of metaheuristics are provided by Osman and Laporte (1996) and Vob et al. (1999).

Metaheuristics have been distinguished as strategies in search exploration (diversification) and search exploitation (intensification) to guide the searching process and explore the search space efficiently to find near-optimal solutions. In search exploration, the algorithm explores all unvisited regions in the search space to escape from the local optima. The algorithm exploits the accumulated search experience (Blum & Roli 2003; Talbi 2009). Each heuristic search algorithm aims to explore the search space of a given problem iteratively and efficiently. However, most heuristic search algorithms are trapped in the local optima during the search process. The major feature of a metaheuristic is its ability to escape from the local optima, thereby creating opportunities to obtain an improved solution (Burke & Silva 2005).

Although metaheuristics are effective and efficient in solving problems, they still suffer from parameter optimization. Setting the best parameter usually redetermines the value of these parameters while solving the problem instances because in-depth knowledge of the problem structure is needed to carefully select the parameter set (Alabas-Uslu & Dengiz 2011). Nonetheless, finding good parameter values requires human expertise and time, which are both expensive and rare (Neumüller et al. 2011). According to Adenso-Diaz and Laguna (2006), approximately 10% of the total time dedicated to designing and testing a new heuristic or metaheuristic is spent on development, and the remaining 90% is consumed by fine tuning parameters. An alternative way to tune parameters is by controlling them throughout the run, and heuristics used in this manner to tune their parameters are generally called adaptive, reactive, or self-tuning heuristics (Alabas-Uslu & Dengiz 2011). Furthermore, these metaheuristic algorithms with a fixed number of solutions are slow in conducting an efficient search solution; a small number of solutions may increase the convergence of the algorithm and reduce the solution exploitation, and a large number may cause unnecessary computation and useless searching because of redundant search (Wu et al. 2010; Lee & Kuo 2012). Based on the problem size, determining a suitable population size is difficult because various problems require different parameter settings during the optimization process (De Jong 2007).

To design metaheuristic techniques, one should consider two standards, namely, (i) search space exploration or diversification, and (ii) its exploitation or intensification. Overall, population-based metaheuristics focus on exploration while single-solution based metaheuristics focus on exploitation (Talbi 2009; Boussaïd et al. 2013; Črepinšek et al. 2013). The following sections report some of the important studies that have employed metaheuristic algorithms to solve CVRP.

a. Single-Solution Based Metaheuristics

The procedure of any S-metaheuristic (local search) starts with one solution to solve an optimization problem and, at each iteration, uses a neighborhood operator to generate a neighboring solution. Based on the accepting criterion, the selected solution is adopted as the current solution. This procedure is repeated for a certain number of iterations until the termination criterion is satisfied (Di Gaspero et al. 2003). During the optimization process, the solution iteratively moves toward a good solution in a certain trajectory in the search space. The final solution may or may not belong to the current solution (Crainic & Toulouse 2003; Talbi 2009). The main merit of using the S-metaheuristic is its strength in finding a good solution faster than the population-based heuristic for combinatorial optimization problem with a smooth solution space. However, the weakness of these algorithms is that they focus more on search exploitation than on search exploration, which means that they do not perform a wider scan of the entire search space, thereby increasing the possibility of being stuck in local optima (Blum & Roli 2003, 2008).

The neighborhood structures have a prominent role in the performance of any S-metaheuristic (Talbi 2009). The existence of an adequate neighborhood leads to enhanced ability of a S-metaheuristic to generate good solutions (Zapfel et al. 2010). Neighbors of a solution are obtained by making minor changes to the solution using neighborhood structures or move operators (Talbi 2009). Some of these operators that have been applied to solve CVRP are the following:

- 2-opt operator: This operator aims to cut two edges between sequential costumers, thereby leading to division of the route into many routes, and then reconnecting these route in various ways to reduce the traveling distance (Bräysy & Gendreau 2005).

- Random swap: Two positions are selected randomly in this operation (in the solution vector) i and j, where $i \neq j$, and then the customers located in positions i and j are swapped where such a move is feasible.

- Random move: In this operation, one costumer at position i is selected randomly and then moved to another position j, which is selected randomly, where $i \neq j$, and then the customer is relocated from position i to position j, and only the feasible move is accepted.

The acceptance strategy is an important component of the S-metaheuristics, which determines the acceptance of the neighbor solution. The acceptance move strategy has a significant role in escaping from the local optima (Burke et al. 2010). A local optimum is defined as "a point in the search space where all neighboring solutions are worse than the current solution" (Burke & Kendall 2005). Acceptance move strategies can be divided into the following two groups: deterministic and non-deterministic (Burke et al. 2010).

a. Deterministic move acceptance where the same decision is always returned regardless of the initial and current solution. In terms of deterministic acceptance criteria, the following concepts are adopted (Talbi 2009; Burke et al. 2013).

- **Best improvement (steepest descent):** The entire neighborhood of the current solution is generated and the best neighboring solution is accepted.

- **First improvement:** The neighboring solutions are generated one by one until the first improving neighbor that is better than the current solution is reached. Then, an improving neighbor is selected to replace the current solution.

- **Random selection:** In this strategy, a random selection is applied to the neighbors that are improving the current solution.

- **Accept all solutions**: Any generated solution is accepted regardless of its quality. This type is adopted to diversify the search.

b. Non-deterministic acceptance criteria where different decisions are usually returned even if the initial solution is the same. Usually, in this type of criteria, few parameters are used such as current temperature, water level, and time (or current iteration) that have a significant impact on the acceptance or rejection of the neighbor according to the acceptance probability (Burke et al. 2010; Burke et al. 2013). Non-deterministic acceptance methods are great deluge and simulated annealing (Burke et al. 2010).

To solve CVRP, many S-metaheuristic approaches have been introduced, such as SA (Harmanani et al. 2011; Vincent et al. 2017), tabu search (TS) (Wisniewski et al. 2012), and GRASP (Layeb et al. 2013).

The following subsections discuss some local search algorithms employed to solve CVRP.

i. Simulated Annealing

SA is a probabilistic optimization algorithm proposed by Kirkpatrick et al. (1983) to solve combinatorial optimization problems. The idea of SA is based on a physical process where the materials are heated at high temperature, and then the temperature is slowly reduced to increase the size of the crystals and decrease their defects. The SA has characteristics that allow it to escape from local optima. It utilizes a random search that accepts good and bad solutions within a specific probability (Wang et al. 2015). Compared with P-metaheuristics, SA has shown good performance in improving local optimization in terms of computation time because it requires less memory space. Therefore, SA has potential to improve several metaheuristics (Othman et al. 2017). As reported in the literature, SA is used as a local search algorithm to improve the P-metaheuristics (Hung et al. 2009). The main advantage of using SA is that it can improve the solution search exploitation without being trapped in local optima (Othman et al. 2017).

In detail, the SA begins with an initial feasible solution and proceeds to generate a neighboring solution by perturbing the current solution. The neighbor with a better objective function than the current one is always accepted. Otherwise, the worse neighbor is accepted or rejected with the probability *P*, which is calculated by formula $P=e^{\Delta f/t}$, where Δf is the difference of the objective function of the current solution *s* and the new neighbour $S`$, i.e., $\Delta f = f(S`) - f(S)$ (for minimization problem); and *t* is a control parameter that denotes temperature. The temperature is cooled off gradually according to a cooling schedule, $g(t)=\beta.t$, where β is the cooling rate ($\beta<1$). The temperature is decreased during the optimization process and thus, the probability of accepting a worse solution also decreases (Harmanani et al. 2011). The pseudocode of the basic SA algorithm is shown in Figure 2.3.

Simulated Annealing Algorithm

```
Input: cooling schedule.
s= s0 ; \\Generation of the initial solution
Tmin = final temperature;
T = Tmax ; \\ Starting temperature
Repeat
    Repeat
        Generate a random neighbor s′;
        ΔE = f(s′) − f(s);\\ f(s)the objective function value of solution s
        If ΔE ≤ 0 then s = s′ \\ Accept the neighbor solution
        Else Accept s′ with a probability e^(−ΔE/Temp) > rand (0 -1) ;
    Until Equilibrium condition
         \\ e.g. a given number of iteration executed at each temperature T
        T = g(T); \\Temperature update
Until Stopping criteria satisfied \\ e.g. T < Tmin
Output: Best solution found.
```

Figure 2.3 Pseudo-code for simulating annealing algorithm (Talbi 2009)

Harmanani et al. (2011) proposed SA with combination of random and greedy algorithm for solving CVRP. The algorithm used a greedy algorithm to generate an initial solution based on a first-fit approach. The result shows that the proposed algorithm achieved a good result for the given problem.

Afifi et al. (2013) proposed SA to solve VRP with time windows and synchronization constraints. The algorithm uses 2-opt and or-opt heuristics to deal with this problem. Experiments on the instances from the literature show that their SA is fast and outperforms the existing approaches and it can improve the result up to 35%.

Wang et al. (2015) proposed SA with residual capacity and radial surcharge, which improves the cheapest insertion heuristic to generate an initial solution, whereby the solution is exchanged between synchronous parallel SA to solve the VRP with simultaneous pickup and delivery during specific individual time windows. Computational results were compared with results from a GA that minimizes the number of vehicles up to 34.48% as the primary objective.

ii. Tabu Search

TS algorithm was first introduced by Glover (1986). It consists of exploring the search space by saving the last *n* visited solutions in the tabu list to avoid repeating the search on those solutions while producing new solutions. The algorithm aims to escape from local optima by leading the algorithm to consider new regions of spaces. However, the parameters (i.e., tabu list and stopping criteria) of the TS algorithm should be tuned, a step that is closely related to the problems (Gendreau et al. 1998). The pseudocode of the basic tabu search algorithm is shown in Figure 2.4.

```
Tabu Search Algorithm
Input: Select initial solution Sol;
Set Sol as a current solution;
Set Sol' as a best solution;
Initialize the tabu list
while stop condition not met
      Given neighborhood function N, tabu list T, and aspiration criterion
      Find the best possible solution Sol' of N(Sol);
      Sol = Sol'    // Replace the current solution by the new one
      Insert the solution Sol (or its attributes into the tabu list T)
      If penalty (Sol) < penalty (Sol')
           Sol' = Sol;    //Save the best so far solution
      endif
      Update the tabu list T;
End while
```

Figure 2.4 Pseudo-code for basic tabu search algorithm (Talbi 2009)

Taillard et al. (1997) and Jin et al. (2012) proposed TS for CVRP. TS is a searching method based on storage structure. By introducing a flexible storage structure (tabu graph) and corresponding taboo rules, this algorithm tries to avoid circuitous search. The result has improved up to 12.5% compare with state-of-the-art algorithm.

Brandão (2011) presented a TS for the heterogeneous fixed-fleet VRP. The results demonstrate that the proposed TS produces high-quality solutions with acceptable time. Four new best solutions are reported for a set of test problems used in the literature. TS was improve the result up to 28%.

iii. Great Deluge

GD algorithm is a local search procedure that was introduced by Dueck (1993). It is an alternative to simulated annealing and is less dependent on parameters. The great deluge algorithm requires only two parameters i.e., computational time and the estimation/desired of a solution quality. The algorithm always accepts a better solution, and a worse solution is accepted if it is less than or equal to a boundary value (for minimization problems), which is referred to as *level*.

At the start, the algorithm sets the *level* as the cost of the initial solution. Then, at each iteration, the level is decreased by the rain speed (UP) using Equation (2.6), this process is repeated until the stopping criterion is reached (Talbi 2009). The pseudocode for the great deluge algorithm is presented in Figure 2.5.

$$level = \text{level} - \text{UP} \tag{2.6}$$

Great Deluge Algorithm

Input Level = 1;
Set initial water level: LEVEL;
Set the rain speed UP
Set s = s_0 ; \\ initial solution generation
Repeat
    Generate a random neighborhood of s' of s;
    If $f(s') \leq$ *level* **then**
    s = s'; \\ *Accept the neighbor solution*
    level = level -UP ; \\ update the water level

Until stopping criteria is meet;
Output: Best solution found.

Figure 2.5 Pseudo-code for great deluge algorithm (Talbi 2009)

Bagayoko et al. (2015) proposed a new insertion GD to solve the practical forest routing problem. They also proposed a mathematical model to minimize the total cost of transport and include hard capacity and hard time-windows constraints. Compared with TS in terms of time and solution quality, the GD method obtains better quality solutions. Since limited solution methods based on GD are available for application to VRP, the following provides a summary of GD for other COPs:

Mcmullan (2007) introduced an extended version of GD for the course timetabling problem which, while avoiding the problem of entrapment in local optima, uses simple neighborhood search heuristics to obtain solutions in a relatively short amount of time. Based on a standard set of benchmark datasets, the results beat over half of the currently published best results with an improvement of up to 60% in some cases.

Kahar and Kendall (2015) introduced a modified and extended GD for the real-world examination timetabling problem. They investigated different initial solutions as well as altered the number of iterations. The result of their proposed GD can produce good-quality solutions compared with the original GD with improvement up to 55%.

Kifah and Abdullah (2015) employed adaptive non-linear GD to tackle the patient admission problems. In their study, the adaptive non-linear decay rate (level) yields better performance than a linear decay GD. Moreover, the proposed GD improved the result up to 83.33% when compared to other state-of-the-art methods.

Acan and Ünveren (2015) proposed a two-stage memory GD for real-valued global optimization functions and used level-based acceptance criterion, which is applied for each best solution extracted in a particular iteration. The performance of the proposed GD is tested on three sets of benchmark global optimization functions with varying sizes. The results demonstrated that the proposed GD performs at least as good as the state-of-the-art methods.

Mandal and Kahar (2015) presented partial graph heuristic with GD to solve the examination timetabling problem. The proposed method is generally able to produce competitive results when compared with the state-of-the-art with improvement up to 33.3%.

iv. Variable Neighborhood Search

Variable Neighborhood Search (VNS) was first proposed by Mladenović and Hansen (1997). The basic idea of VNS is to explores either at random or systematically a set of neighborhoods to change in the neighborhood of a local search to escape from local optima (Talbi 2009; Amous et al. 2017). The pseudocode for the variable neighborhood search is presented in Figure 2.6.

```
Variable Neighborhood Search
Input: a set of neighborhood structures N_k for k=1, …, k_max for shaking.
x = x_0; /*Generate the initial solution*/
repeat
        k = 1;
        repeat
          Shaking: pick a random solution x′ from the k^th neighborhood N_k(x) of x;
          x″ =local search(x′) ;
          If f(x″) < f(x) Then
              x = x″ ;
              Continue to search with N_1 ; k = 1 ;
          Otherwise k=k+1 ;
        Until k = k_max
Until Stopping criteria
Output: Best found solution
```

Figure 2.6 Pseudo-code for variable neighborhood search algorithm (Talbi 2009)

Xiao et al. (2014) proposed VNS with SA to solve CVRP by combining the strengths of both algorithms. They utilized predefined neighborhood structures to improve search efficiency while bringing in the uphill ability of SA to endow the algorithm with the feature of global optimization. The result of the proposed algorithms were competitive when compared to other state-of-the-art methods.

Amous et al. (2017) proposed VNS for CVRP to minimize the total traveled distance. The algorithm includes a variable neighborhood descent based on several different neighborhood structures to intensify the search effort. This algorithm introduced the variable neighborhood descent algorithm at the phase of the local search to increase the in-depth search of the VNS algorithm. The experimental results show that VNS improved the result up to 29.1% compared to those reported in the literature.

v. Large Neighborhood Search

Large Neighborhood Search (LNS) was introduced by Shaw (1997). Designing LNS may improve the quality of the obtained solutions because more neighbors are considered at each iteration. However, additional computation time is needed to generate and evaluate a large neighborhood. The pseudocode for the large neighborhood search is presented in Figure 2.7

```
Large Neighborhood Search
Function LNS(s ∈ (solutions ) , q ∈ N )
    solution s_best=s;
    repeat
        s′ = s;
        remove q requests from s′
        reinsert removed requests into s′;
        if (f (s′) < f (s_best)) then
        s_best = s′;
        if accept(s′, s) then
        s = s′;
    until stop-criterion met
  return s_best;
```

Figure 2.7 Pseudocode for large neighborhood search algorithm (Ropke & Pisinger 2006)

Akpinar (2016) proposed LNS with ACO to solve CVRP. The proposed hybrid algorithm aims to enhancing the performance of LNS by providing a satisfactory level of diversification through the solution construction mechanism of the ACO. The experimental results show that the proposed algorithm has satisfactory performance in solving CVRP with improvement up to 56.67%.

Kır et al. (2017) proposed an algorithm based on adaptive large neighborhood search (ALNS) and TS with several specifically designed operators and features to solve the CVRP. Some successful features of tabu search and ALNS are allowing infeasible solutions, flexible parameters, destroy/repair operators, diversification strategy, intensification strategy, and adaptive memory. The proposed algorithm improved the result up 46.6%.

b. Population-Based Metaheuristics

Population-based metaheuristics start from an initial set of solutions called population. Then, they iteratively use a generation and replacement procedure to generate a population of solutions. The generation procedure is implemented to generate a new population of solutions using evolutionary operators, such as crossover and mutation operators or other techniques. The replacement procedure is implemented to replace the current population with the newly generated population by using an updated technique. This process employs iteration until a given stopping criterion is reached (Talbi 2009). The main merit of using population-based metaheuristics is that their strength allows improved diversification of the entire search space of a combinatorial optimization problem. They perform a wider scan of the entire search space, which leads the search process to move from a region to another that contains different solutions in terms of quality and structure. This capability increases the possibility of obtaining a near-optimal solution. However, population-based metaheuristics are considered weak in exploiting or intensifying the solution search space (Blum & Roli 2008). The following subsections discuss some population-based algorithms that have been employed to solve CVRP.

i. Ant Colony Optimization

ACO algorithm was proposed by Dorigo et al. (1991) and can be defined as the imitation of the behavior of ant colonies when looking for food. During their journey in search of food, ants leave actual spots in the routes they trod to send a sign to the others to keep following a good route to obtain food. These actual spots are called pheromones. By doing so, other ants follow the pheromone path to search for neighboring resources. The level of pheromones on the path is associated with the distance of the path to the food source. A strong pheromone level means low distance trails over a period. In the algorithm, each solution is represented by ants, and the information on food, as represented by the pheromone, is used to generate solutions in the next iteration (Dorigo & Stützle 2010). The pseudocode of the basic colony optimization algorithm is shown in Figure 2.8.

Ant Colony Optimization Algorithm
Initialize the pheromone trails
Repeat
For each ant **Do**
Solution construction using the pheromone trail ;
Update the pheromone trails:
Evaporation ;
Reinforcement ;
Until Stopping criteria
Output: Best solution found

Figure 2.8 Pseudocode for basic colony optimization algorithm (Talbi 2009)

Mazzeo and Loiseau (2004) proposed the ant colony algorithm for CVRP. Ant colonies are regarded as procedures that build solutions. Several solutions are built at the same time, exchange information during the procedure, and use the information of previous iterations. The role of the tabu list is obtained by a set of already visited neighbors, and is forbidden for the ants at the current iteration. The neighbor client is randomly chosen according to probability. Proposed algorithm improved the result up to 14.29% compared to state-of-the-art algorithm

Xiao and Jiang-Qing (2012), proposed ant colony optimization to tackle CVRP. The algorithm constructs the candidate solutions by using NN, and then these solutions are subjected to mutation operation and 2-opt heuristics. The numerical results demonstrate the competitiveness of the proposed algorithm.

ii. Genetic Algorithm

GA was proposed by Holland (1975). Since then, GA has been popular because it can contribute to formulating good solutions for complex mathematical problems in a reasonable amount of time (Talbi 2009). The major procedures within GA include the selection, crossover, mutation, and updating processes. GA begins with a group of solutions created either arbitrarily or through a specific version of heuristic algorithms.

Then, the fitness of these solutions is measured. Thereafter, the process of selection, according to the fitness value, is employed to choose two solutions known as parents. These parents are selected randomly or by using a selection mechanisms such as the Roulette wheel (Baker 1987). The selected parents pass through the crossover and/or mutation operator. The goal of the crossover operator is to merge the chosen parents to utilize or interchange worthy data between these parents to create the offspring. The mutation is a mono operator that operates on the produced offspring to obtain the necessary diversity and protect the search against being stuck in local optima. When the offspring is more efficient than the worst individual in the current population, it replaces the worst solution. This procedure is reiterated several times in a process called generation in GA. The pseudocode of the genetic algorithm is presented in Figure 2.9.

```
Genetic Algorithm
Start
   P = initial population;
   evaluate (P);
While termination criterion not satisfied Do
      P′ = recombines (selected (P));
      mutate (P′);
      evaluate (P′ );
      P = replace (P′ ∪ P);
End while
End
```

Figure 2.9 Pseudocode for genetic algorithm (Zapfel et al. 2010)

Baker and Ayechew (2003) proposed a genetic algorithm to solve CVRP. The algorithm starts by constructing a set of solutions randomly. To improve the quality of the solution, two types of neighborhood search are performed on each individual of the initial population (2-opt and λ-interchange). Computational results are given for the pure GA, which is compared using a GA with multiple neighborhood search. The result shows that the hybrid GA is competitive with TS and SA in terms of solution time and quality.

Nazif and Lee (2012) proposed GA to solve CVRP, which uses an optimized crossover operator designed by a complete undirected bipartite graph to find an optimal set of delivery routes that satisfy the requirements and provide minimal total cost. Their findings show that the proposed GA is competitive when compared to the state-of-the-art methods with improvement 7.69% in terms of the solution quality.

Kumar et al. (2014) proposed GA to solve VRP using specialized crossover called fitness-aggregated GA and various fitness assignment approaches. The researchers investigated the performance of the proposed GA on popular VRP benchmark instances. The results show that the proposed GA is competitive with the best-known results in the literature and improved the result up to 5.26%.

iii. Particle Swarm Optimization

Particle swarm optimization (PSO) is a population-based optimization algorithm introduced by Kennedy and Eberhart (1995). It simulates the collective behavior of wild animals in nature, such as a flock of birds and a school of fish searching for a region with enough food. In PSO, each particle represents a solution, and the swarm of particles moves through the search space to find the global optima. To direct the search toward the best region in the search space, the swarm of particles is maintained throughout the search procedure, and the particles share their information among the others. The particles move through the multidimensional problem search space with particular velocity and follow the currently known best particles. Each particle adjusts its position according to its own experience and that of its neighbor particles. At each iteration, the particles move to the next position with specific velocity using their best solution and global best-solution values. The algorithm balances between exploration and exploitation by combining local search methods with global search techniques (Shi & Eberhart 1999). The pseudocode of the basic PSO algorithm is shown in Figure 2.10.

```
Particle Swarm Optimization
Random initialization of the whole swarm ;
Repeat
  Evaluate f (xi) ;
  For all particles i
        Update velocities of particles vi(t);
        Move to the new position: xi(t) = xi(t − 1) + vi(t) ;
        If f (xi) < f (pbesti) Then pbesti = xi ;
        If f (xi) < f (gbest) Then gbest = xi ;
        Update(xi, vi) ;
  EndFor
Until Stopping criteria
```

Figure 2.10 Pseudocode for basic PSO algorithm (Talbi 2009)

Kim and Son (2012) proposed PSO to solve CVRP, which uses particle encoding and decoding based on a probability matrix for assignment of customers to routes and used other algorithms to sequence customers within the routes. Compared to the existing research that uses the PSO solely, the proposed approach applies the PSO to both simultaneously. The results shows, PSO improve the result up to 42.86% compared with the previous methods.

Chen et al. (2015) proposed PSO to solve CVRP with pickups and deliveries, where the PSO uses adaptive multi-swarm strategy to enhance the simple search strategy. The proposed PSO employs multiple PSO with the so-called punishment mechanism to search for the optimal solution. The proposed PSO improved the result up to 57.14% compare with state-of-the-art methods.

Marinakis and Marinaki (2013) introduced a method called combinatorial expanding neighborhood topology PSO (CENTPSO). This method boosts the performance of the algorithm by using an expanding neighborhood topology. The researchers replaced the equation of positions, including a path-relinking strategy and a different role of the velocities. Scholars used a local neighborhood topology where the size of the neighborhood begins from a small one and expands during the iterations. Thus, the proposed algorithm combines the advantages of the exploration abilities of a global neighborhood structure with the exploitation abilities of a local neighborhood structure. The results show that the quality of the solutions improved up to 37.5% from state-of-the-art algorithms.

Tlili et al. (2014) proposed a hybrid metaheuristic called PSO-ANS, which integrated a variable neighborhood search within PSO. The result shows that the proposed method was introducing the solution cost up to 0.96%.

iv. Intelligent Water Drops

The intelligent water drop (IWD) algorithm is a P-metaheuristic developed by Hosseini (2007). This algorithm is inspired by the natural way in which water flows to a river. A water drop follows the shortest path when it moves from one point to another by depending on the force of gravity and terrain covered. When a drop of water moves, it carries an amount of soil on its path depending on its velocity. The more velocity, the greater the amount of soil carried. Thus, the path with the least amount of soil is used by other water drops (Hosseini 2007).

Wedyan and Narayanan (2014) proposed the IWD and SA algorithms to solve CVRP. The results of each algorithm were compared with each other and with known optimal results. The IWD obtained better results than the SA algorithm. However, certain limitations arise when using some static parameters that have been assumed to be equal to particular values. The result shows that IWD improved the result up to 57.14% compared with state-of-art methods.

To solve CVRP, Booyavi et al. (2014) proposed an improved IWD (IIWD) algorithm that yielded reasonable consequences in exploring and exploiting the solution space. The experimental results revealed that the proposed algorithm can effectively tackle these instances with improved up to 57%s. compared with the state-of-the-art methods. However, they highlighted the issues of hybridization of IIWD with other heuristics or metaheuristics.

v. Cuckoo Search Algorithm

The cuckoo search (CS) algorithm is a P-metaheuristic developed by Yang and Deb (2009). CS was inspired by the brood parasitic behavior of some cuckoo species. Cuckoos use nests of other birds as their host nests. Cuckoos move via Levy flights in the search space and lay their eggs in the chosen host nests. Levy flights can be considered as the key component of CS, which describes the foraging patterns of many animals and insects. In fact, efficient randomization provides the algorithm an improved balance between intensification and diversification by preserving the step length, whether small or large (Xiao et al. 2017)

Zheng et al. (2013) proposed a hybrid cuckoo search with GRASP called CS-GRASP algorithm. They used GRASP to initialize the population at first stage, and then used swap, inversion, and path relinking as intensification strategy to explore trajectories between elite solutions and CS to keep the best solution obtained during the iteration process. One of the main problems is how the cuckoo will move from the current solution to the global or local optimum. The result shows that the proposed algorithm improve the result up to 16.67% compared with state-of-the-art.

Alssager (2017) applied basic CS and hybrid CS with SA for CVRP. The author hybrid CS with SA to enhance solution intensification. The result shows that proposed algorithm was able to find near-optimal solution in reasonable time

Xiao et al. (2017) applied improved CS for CVRP. This study proposed a new way to adapt an extension of CS called CS-Ouaarab. Levy flights specialize the search areas for CS-Ouaarab; thus, CS-Ouaarab is able to seek good solutions using local search. Levy flights provide displacements by zones but not solutions so that the probability of being trapped in a local optimum is reduced. The results show that the proposed algorithm produced acceptable results in terms of solution quality.

Other population-based algorithms

Zhang and Lee (2015) proposed improved artificial bee colony algorithm with a specific design for the CVRP. Called RABC, the improved algorithm better balances the effect of diversification and intensification. The greedy selection mechanism is applied in the neighborhood search to ensure that the new solution is employed only if it is better than the current solution. In the RABC algorithm, the tradeoff between the exploration of the search space and the exploitation of the promising area is achieved. The results show that the proposed algorithm improved the results by 17.89% compared with basic artificial bee colony (Zhang & Lee 2015).

c. Hybrid Metaheuristic

The research on metaheuristics evolve over time from simple to more difficult form, the researcher on operations research have been working on an impressive number of algorithms that purely follows the paradigm of the traditional and basic framework of metaheuristics. This is due to of the fact that any metaheuristics on its basic form have limited successes for many COPs (Raidl 2006; Blum et al. 2011). This motivates many scholars to get the benefits from the valuable optimization expertise that was done over years. Clearly, each individual metaheuristic had its weaknesses and strengths depending on its own characteristics (Blum et al. 2011). Thus, the research community turns toward the hybridization of different metaheuristics such as GRASP, SA, and GA, or combination with various another algorithmic and heuristic components. This kind of combination has considerably risen among the operations research (El-Abd & Kamel 2005; Talbi 2009; Blum et al. 2010). However, the best results obtained for many COPs in industry or in academia are obtained from these combinations, which are well known as hybrid metaheuristic.

The motivation behind the hybridization of different heuristics and metaheuristics is to exploit the complementary character of different optimization strategies. Hybrids are believed to benefit from synergy and therefore largely exploit many areas in the search space with the benefit of extensively explore and focus of the local search space area (Blum et al. 2011). However, one of the major key usage of such combinations is the fact that it can provide a good balancing between diversification and intensification (Lozano & García-Martínez 2010; Krause et al. 2013). However, the balancing between diversification and intensification is considered as difficult task (Talbi 2009). Hybridization has shown to be successful for many applications, therefore it may serve as guidance for new algorithm developments (Preux & Talbi 1999; Blum et al. 2010; Lozano & García-Martínez 2010).

Wang and Lu (2009) applied hybrid genetic algorithm to solve CVRP. The proposed algorithm involves three stages. First, the nearest addition method was incorporated into sweep algorithm to generate an initial population rather than adopting either the nearest addition method or sweep algorithm alone. Second, using response surface methodology to considerably reduce the time required to identify GA parameter settings compared with the conventional trial-and-error method was employed to optimize crossover probability and mutation probability through systematic experiments. Finally, the improved sweep algorithm combined with the elitism policy improves the exploration capability of GA, and avoids premature and rapid convergence to a limited region. The proposed algorithm improved the result up to 20% compared with best solution.

Huang and Ding (2013) proposed ACO, which adopts new transfer rules, adds the path weight matrix, save matrix, angle-factor functions, and new visibility functions, and at the same time updates the pheromone model with a reward function, thereby overcoming the limitations of slow convergence and falling into local optimum. The 3-opt method was used to update the optimal solution to shorten the length of rescue route. The experimental results show that the proposed algorithm is competitive compared with other algorithms.

Qi (2013) proposed hybrid ACO with simulated annealing for CVRP. SA provides a good initial solution for ant colony optimization, and iterative local search method is used to search for a close-to-optimal solution in local scope. Experimental results show that the proposed algorithm is superior to original ant colony optimization.

Adiba et al. (2013) proposed hybrid ant colony system with large neighborhood search aims to improve the quality of solution found. The computational results show that the proposed algorithm has shown to be competitive with the best existing methods in terms of solution quality with improvement up to 20%.

Akpinar (2016) proposed hybrid large neighbourhood search with ant colony optimization (LNS–ACO) algorithm aims at enhancing the performance of search algorithm by providing a satisfactory level of solution diversification. Computational results indicate that the proposed algorithm has improved the result up to 56.67%.

Zhou et al. (2013) proposed hybrid bat algorithm with GRASP for CVRP. The aim of this hybrid to enhanced an intensification strategy of bat algorithm to explore local trajectories connecting elite solutions. The results show that the proposed algorithm obtain competitive result with improvement up to 16.67%.

Chen et al. (2006) proposed hybrid particle swarm optimization and simulated annealing. SA used to avoid being trapped in a local optimum. The computational study showed that the proposed algorithm is a feasible and effective approach for CVRP and improved the result up to 20% compared with state-of-the-art in term of solution quality.

Yassen et al. (2015) propose a hybrid metaheuristic algorithm that hybridizes a harmony search with SA for the purpose of improving the performance of harmony search algorithm. Harmony search algorithm is used to explore the search spaces. Whilst, SA is used inside the harmony search algorithm to exploit the search space and further improve the solutions that are generated by harmony search algorithm. The results showed that the hybrid harmony search algorithm obtained better results when compared to basic harmony search algorithm (without SA) and improved it up to 5.36% when compared with state-of-the art. These results demonstrated that the use of the local search algorithm within the harmony search does improve the search process.

The literature review shows that researchers have recently focused on P-metaheuristic rather than local search approaches for capacitated vehicle routing problem. The reason may be that the quality of the solutions produced by population-based approaches are better than those produced by local search approaches because the local search approaches are more concerned with exploitation than exploration. However, in the exploration process, the selected solutions may be trapped in local optima. This condition motivates us to focus on the population-based metaheuristics to solve CVRP.

2.4.4 Discussion of Previous Approaches

Over the decades of research on metaheuristics many algorithms have been proposed to address CVRP problem, but the solution is still ongoing. This condition is due to the NP hardness of the problem (Bräysy & Gendreau 2005). Furthermore, some of these algorithms cannot explore the entire search space, which leads to fewer chances of escaping local optima. Moreover, the complex structure of other algorithms complicates the task of enhancing their quality (Cordeau et al. 2002; Bräysy & Gendreau 2005). It is clear that each individual metaheuristic had its strengths and weaknesses depending on its optimization characteristics, which motivated researchers towards the combination of different metaheuristics to overcome this weakness.

Table 2.1 presents a review of the related studies to solve CVRP by focusing on the techniques that have been used on the CVRP benchmark dataset. The table is clustered based on the algorithm employed (i.e., single-solution-based and population-based algorithms). Table 2.1 also shows the strength and limitations of each algorithm. The single-solution-based algorithms provide a quick search in the iterative search process and can escape from the local optima. However, some limitations exist, such as weakness in exploring the search space and diversification (Blum & Roli 2003, 2008). On the other hand, the population-based algorithms can provide good exploration, but they need longer computational time (Blum et al. 2011). They are also weak in terms of intensification and a number of parameters must be tuned in advance.

Various single-solution-based and population-based algorithms were widely studied with the CVRP dataset. Although population-based metaheuristics have been applied on the CVRP, no investigation on the behavior of the WFA algorithm has been conducted. Thus, to choose a suitable population-based algorithm and/or hybrid algorithm, further investigation on CVRP should be performed. Our goal in this study is to explore a relatively new algorithm called WFA because its application in CVRP has not been sufficiently examined in the literature. Thus, the next section discusses the WFA algorithm.

Table 2.1 Summary of studies on related algorithms and their characteristics for CVRP datasets.

Techniques **Single-solution based**	**Author**	**Strengths**	**Limitations**
Simulated annealing	Harmanani et al (2011); Afifi et al. (2013); Wang et al. (2015)	• Able to escape from local optima. • Good in terms of exploitation. • Effective convergence.	• Many parameter need to be tuned. • Influenced by generated random number. • Need intensive computational requirements which require more time.
Variable neighborhood search	Xiao et al. (2014); Amous et al. (2017)	• Able to escape from local optima. • Good in terms of exploitation. • Fewer control parameter needed	• Deals with a large number of neighborhood structures; thus, the algorithm take more time.
Tabu search	Jin et al. (2012); Brandão (2011)	• Able to escape from local optima. • Good in terms of exploitation and exploration.	• Many parameter need to be tuned. • Tabu list size significantly affects the performance of search process; thus, it should be set properly.
Large neighborhood search	(Akpinar 2016); Kır et al. (2017)	• Able to escape from local optima. • The algorithm achieves the intensification and diversification by systematically changing the neighborhood structure.	• Many parameters need to be tuned. • It deals with a large number of neighborhood structures; thus, the algorithm take more time.

To be continued …