DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

THE VEHICLE ROUTING PROBLEM WITH TRAFFIC CONDITIONS

by
Cansu KÖROĞLU

October, 2018
İZMİR

THE VEHICLE ROUTING PROBLEM WITH TRAFFIC CONDITIONS

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science
in Industrial Engineering, Industrial Engineering Program

by
Cansu KÖROĞLU

October, 2018
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"THE VEHICLE ROUTING PROBLEM WITH TRAFFIC CONDITIONS"** completed by **CANSU KÖROĞLU** under supervision of **ASSOC. PROF. DR. A. SERDAR TAŞAN** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. A. Serdar TAŞAN

Supervisor

Assoc. Prof. Dr. Yiğit KAZANÇOĞLU

(Jury Member)

Asst. Prof. Dr. Burcu FELEKOĞLU

(Jury Member)

Prof.Dr. Kadriye ERTEKİN

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

During the progress of this thesis, some special people encouraged me to finish and I would like to thank all those people who have supported me immeasurably.

Firstly, I wish to express my gratitude to my supervisor Assoc. Prof. Dr. A. Serdar TAŞAN. He gave me guidance and support for this thesis. I am also grateful to the authors who helped me to get the data sets that I need for the computations.

Besides, I must particularly thank my friend Kadir Tandoğan TİLKİ who have experienced this process with me. His friendship and sincerity eased the process.

Finally, I am very thankful to my whole family for their endless love, support and tolerance. They encourage me to complete all this process. They have opened up my horizon with their valuable and creative ideas through the development of this thesis, they are always by my side whenever I need.

Cansu KÖROĞLU

THE VEHICLE ROUTING PROBLEM WITH TRAFFIC CONDITIONS

ABSTRACT

The Vehicle Routing Problem (VRP) is a combinatorial optimization and integer programming problem seeking to service a number of customers with a fleet of vehicles. The classical VRP aims to find a set of routes at a minimal cost for beginning and ending points of the route at the depot, so that the known demands of all customers are fulfilled. Each customer is visited only once, by only one vehicle, and each vehicle has a limited capacity. It is important that the right amount of product should be transported at the right time to ensure customer satisfaction. So, choosing the right route is very critical for transportation systems. The vehicle routing problem is an NP-Hard problem. When number of customers and vehicles, constraints of capacity, and time increases, the solution of the problem becomes more difficult.

In this study, a vehicle routing problem, which is suitable for real-life situation is discussed. In order to establish the best route apart from the traveled total distance, time must be taken into account for customer satisfaction. Traffic volume on the selected route will affect the time which is spent on the route. So, traffic plays an important role in determining the route. In this study, traffic conditions on alternative ways are taken into account to solve VRP. To achieve this aim, literature review is conducted, then integer linear programing formulation and metaheuristic method which is a combination of Genetic Algorithm, Lin-Kernighan Algorithm and 2-opt Algorithm, is developed.

Keywords: Vehicle routing problem, traffic condition, genetic algorithm, Lin-Kernighan algorithm

TRAFİK KOŞULLARI İLE ARAÇ ROTALAMA PROBLEMİ

ÖZ

Araç rotalama problem (ARP), belirli bir talebi minimum maliyet ile başlangıç ve bitiş noktası depo olan rota boyunca müşterilere ulaştırmayı amaçlar. Temel bir araç rotalama problemi, kapasiteleri belirli araçlar ile tüm müşterilerin memnuniyetini sağlayan, minimum taşıma maliyetli rotaların oluşturulması için kullanılır. Müşteri memnuniyetini sağlayabilmek için taşımanın doğru miktarda ve istenilen zamanda yapılması önemlidir. Bu durum malzeme taşınmasında ve ürün dağıtımında doğru rotanın seçilmesinin önemini arttırmaktadır. Araç rotalama problemi NP-zor bir problemdir. Bu nedenle, matematiksel çözüm yöntemleri ile kabul edilebilir sürelerde sadece küçük boyutlu problemlerin çözümü yapılabilmektedir. Müşteri ve araç sayısı, kapasite ve zaman kısıtları arttıkça problemin çözümü zorlaşmaktadır.

Araştırmada gerçek hayat koşullarına uygun bir araç rotalama problemi ele alınmıştır. Rotanın en iyi şekilde oluşturulabilmesi için de kat edilen toplam mesafe dışında, müşteri memnuniyeti bağlamında zamanın da dikkate alınması gerekmektedir. Seçilen rota üzerindeki araç trafiği, rota üzerinde harcanan zamanı etkileyeceğinden, rota belirlenmesinde önemli bir rol oynamaktadır. Dolayısıyla, bu çalışmada araç rotalama probleminin çözümünde alternatif yollar üzerindeki trafik koşulları da dikkate alınmış olup, bu çalışma ile pratik hayata ve literatüre katkı yapılması hedeflenmiştir. Problemin formülasyonu ve çözümü için öncelikle literatür araştırması yapılmıştır. Literatür araştırmasını takiben, belirlenen araç rotalama probleminin tamsayılı doğrusal programlama ile matematiksel formülasyonu oluşturulmuştur ve Genetik Algoritma, Lin-Kernighan Algoritması ve 2-opt algoritmasının birleşiminden oluşan metasezgisel bir metot geliştirilmiştir.

Anahtar kelimeler: Araç rotalama problemi, trafik etkisi, genetik algoritma, Lin-Kernighan algoritması

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

CHAPTER ONE
INTRODUCTION

Vehicle Routing Problem (VRP) is one of the current challenges in transportation systems. Using limited resources in the most efficient way is important in all fields. Aim of the VRP is the same for minimizing transportation costs. It tries to find optimum routes with less resources and with higher satisfaction. Because of that, VRP is the frequently studied topic in literature and it is also encountered in daily life.

VRP has many variants because of its interest in daily life. The constraints encountered in daily life are also studied in the literature. In this study, while a solution method for VRP is developed for the literature, the probing effect of traffic condition to VRP which is faced in daily life has been tried to be reflected. For this purpose, an algorithm is developed for solving Capacitated Vehicle Routing Problem (CVRP), Pick-up Delivery Vehicle Routing Problem (PDVRP) and their variant that considers traffic condition.

Firstly, in Chapter 2, description of VRP and literature review is given for understanding of the problem. Then, mostly studied variants of VRP are described briefly.

In Chapter 3, solution methods for VRP are explained. Integer linear programing, dynamic programing and branch and bound method are described as exact methods. 2-opt Algorithm, Lin-Kernighan Algorithm and Genetic Algorithm are elucidated as heuristic and metaheuristic methods.

Integer linear programing formulation and a small computational study, which considers traffic condition, is performed in Chapter 4. This formulation is based for proposed metaheuristic that is described in the following chapters.

A case study for CVRP, PDVRP and their variants with effect of traffic condition are examined in Chapter 5. Different data sets for CVRP and PDVRP are tested. For traffic condition, real life data for Istanbul is obtained and proposed method is tested with this data. Lastly in Chapter 6, the study is concluded.

CHAPTER TWO
VEHICLE ROUTING PROBLEM

2.1 Description of Vehicle Routing Problem

The vehicle routing problem (VRP) is one of the main components of distribution. Organizations should deliver goods and conditions that vary each day (Cordeau, Laporte, Savelsberg, & Vigo, 2007). VRP is a combinatorial optimization subject that is widely studied in literature. It has exact and heuristic solution techniques and it generalizes Traveling Salesman Problem (TSP). Therefore, VRP is NP-hard like TSP. VRP was first studied by Dantzig & Ramser (1959) and they mentioned in their article that optimum routing of delivery vehicles between warehouse and a set of customers with minimum transportation cost. Kumar & Panneerselvam (2012) mentioned that VRP is about finding an optimal route for a fleet of vehicles to service a set of customers, given a set of constraints.

VRP literature growth is exponential with a %6 annual growth rate according to Eksioglu, Vural, & Reisman (2009). Because of this growth at the VRP literature, it is difficult to follow developments.

Transportation costs are considerable cost components for organizations. According to Bell & Griffis (2010), VRP is economically important because it reduces the cost of transportation. When an organization optimize its routes or number of its vehicle it can satisfy its costumer better and, in this way, it can achieve competitive position.

VRP has different types and these types vary according to the constraints which are problems have. Basically, VRP has three main constraints; routes should start and finish at the warehouse, all customers should visit once and demands of all customers should be satisfied. Although these three constraints are common for most types of VRP, they could be changed for some specific types. Kumar & Panneerselvam (2012) define variants of VRP as “formulated based on the nature of the transported goods, the quality of service required and the characteristics of the customers and the vehicles".

2.2 Types of Vehicle Routing Problem

Transportation is a matter of everyday life. Companies need to transport their services one point to another. As well as for the individuals transportation can be a problem. Knowing the optimum route between the points to be visited during the day, can prevent the waste of time and energy. So, VRP is a major problem in daily life and it encounters different constraints depends on situations. Although there are three main constraints for VRP, different constraints can be added, or main constraints can be extended depending on real life situations. VRP classified according to these constraints. The types of VRP which is frequently studied in the literature are as follows:

- Capacitated VRP (CVRP)
- VRP with Pick-up and Delivering (PDVRP)
- VRP with Time Windows (VRPTW)
- Multiple Depots VRP (MDVRP)
- Split Delivery VRP (SDVRP)

Braekers, Ramaekers, & Nieuwenhuyse (2016) classified 277 VRP articles which published between 2009 and 2015. Results about the variants of VRP and the usage in the literature are shown at Table 2.1.

Table 2.1 Variants of the VRP (Braekers, et al., 2016)

Variant	Relative presence (2009-2015)
Capacitated vehicles	90.52%
Heterogeneous vehicles	16.51%
Time windows	37.92%
Backhauls	18.65%
Multiple depots	11.01%
Recourse allowed	9.48%
Multi-period time horizon	8.87%
Precedence and coupling constraints	8.56%
Subset covering constraints	8.56%
Split deliveries allowed	6.12%
Stochastic demands	6.12%

These types also are divided into sub varieties with additional constraints. For example; CVRP is studied to fleet with homogeneous or heterogeneous capacities.

This study focuses on CVRP, PDVRP and these variants of VRP considering the traffic effect.

2.2.1 Capacitated VRP

Capacitated Vehicle Routing Problem (CVRP) is the variant of VRP which vehicles have certain capacities. Companies must service a set of customers with known demands on the routes that have minimum transportation cost, which are starting and ending points at a depot and the vehicles are homogeneous and having a certain capacity in CVRP (Pisinger & Ropke, 2007). Vehicles cannot transport more goods than their capacity at once. Customers should have satisfied under this additional capacity constraint.

CVRP is the most studied variant of the VRP in literature. According to Braekers, et al. (2016), CVRP is mostly studied variant of VRP at between years 2009 and 2015 with a %90 rate.

CVRP is also has its own variants. If the problem studied with vehicles which have different capacities, it is called Heterogenous VRP. In another situation, customers may have to visit in a certain time and this type is CVRP with Time Windows. These variants of CVRP can be expanded with specific additional constraints.

Minimizing the total travel distance is the general aim of all VRPs. Time spent in the transport and transportation costs such as fuel can be reduced when the travel distance is minimized. But in real life, minimizing travel distance is not enough for reducing spent time and transportation cost. External factors like traffic can affect the time spent on transportation. Spent time between two nodes is changed depending on traffic volume. One of the focuses on this research is CVRP with the traffic conditions.

2.2.2 VRP with Pick-Up and Delivering

Customers may have different needs in transportation network. Some may demand pick-up and the others may demand delivery or both simultaneously. Taşan & Gen (2011) explain VRP with Pick-Up and Delivering as "customers need simultaneous pick-up of goods from their location and delivery of goods to their location by vehicles start at a warehouse companies serve customers with pick-ups and deliveries from/to their locations".

Capacity of the vehicle is needed to consider in PDVRP. Because pick-ups and deliveries are affecting the capacity in different ways. This make the capacity planning more difficult than the other VRPs. The total amount of goods to deliver in a route must not exceed the capacity of vehicle, vehicle should have enough capacity to pick-up the goods in a route and capacity of the vehicle must not exceed at any customer. The fullness of the vehicle depends on the sequence of customers. So, in PDVRP, visiting order of customers should be considered besides total distance.

2.2.3 VRP with Time Windows

Timing is a situation that companies need to consider. Any delay in supply chain can cause trouble at other operations of companies and customer satisfaction is affected from this. The vehicle routing problem with time windows (VRPTW) considers these timing.

VRPTW is explained by Desrochers, Desrosier & Solomon (1992) as service, which is pick-up, or delivery can begin within the time window defined by customer and should include the earliest and the latest start of service.

2.2.4 Multiple Depots VRP

Companies should decrease their transportation cost to compete with the others. In large distribution systems having multiple depots can be more efficient than having one depot. Multiple Depots VRP handles with this situation. Surekha & Sumathi (2011) have mentioned that it is a difficult for companies to determine which customers are served by which depots without exceeding the capacity constraints.

2.2.5 Split Delivery VRP

Customers required to visited only once in classical VRP. This constraint removed for Split Delivery VRP. Each customer can be visited more than once and depending on this, demand of customers can be greater than the vehicle capacity in Split Delivery VRP (Archetti & Speranza, 2008)**.**

CHAPTER THREE
SOLUTION METHODS FOR VEHICLE ROUTING PROBLEM

3.1 Overview of Solution Methods for Vehicle Routing Problem

VRP has a major place in distribution management and many companies are dealing with this problem. The problem emerges in different types because of the diverse constraints which are companies are faced with in daily life. Due to these differences, VRP has been studied extensively in operational research literature. Also, it is difficult to solve as the size of the problem grows and this is another reason for researches in recent years. Therefore, there are many solution methods for VRP which are exact, heuristics and metaheuristic methods.

Braekers, et al., (2016) survey 277 articles which are published between 2009 and 2015, and determine the trend in VRP litarature. According to their study, metaheuristic methods are mostly used in recent years. The usage percentages of the methods are at Table 3.1.

Table 3.1 Overview of applied methods (Braekers, et al., 2016)

Applied Method	**Articles**	**Overall (%)**	**2009 (%)**	**2010 (%)**	**2011 (%)**	**2012 (%)**	**2013 (%)**	**2014 (%)**	**2015 (%)**
Metaheuristic	233	71.25	65	63	65	77	80	76	65
Exact Method	56	17.13	17	20	26	10	13	17	19
Classical Heuristic	32	9.79	11	15	15	17	4	5	10
Real-time solution methods	11	3.36	6	0	6	4	5	2	0
Simulation	7	2.14	4	4	6	0	2	0	0

3.2 Exact Methods

Exact methods can be useful for small problems because of it guarantee to give an optimum solution. VRP is a NP-hard problem and if problem has large instances, it may not be possible to find results in acceptable time with exact methods. Integer linear programming formulations, branch and bound method and dynamic programming are among the exact methods.

3.2.1 Integer Linear Programming Formulations

Mathematical formulations are used in integer linear programming for ensure exact solution. These mathematical formulations based on decision variables, constraints and objective function. Different types of VRP can be solved integer linear programming formulation by reorganizing variables, constraints or even objective function. Although integer linear programming is easy to adapt different situations, it is not always the most useful method. The solution time increases when problem size grows. Therefore, integer linear programing formulations methods are more suitable for small sized problems.

3.2.2 Branch and Bound Method

Branch and bound method has been developed for TSP (Balas & Toth, 1983). Encouraging results for big problems were obtained for TSP by Carpaneto & Toth (1980). Laporte, Mercure, & Nobert (1986) adapted for CVRP in their article.

Branch and bound method divides the problem into sub problems and decides which sub problem to be concluded. The method compares every new solution with the previous solutions and tries to find optimal route. If sub tours occur, method tries to avoid them.

3.2.3 Dynamic Programming

Successive steps are used to solve the problem in the dynamic programing. Small sub problems are solved according to these steps. Then these solutions are used for solving large sub problems.

3.3 Heuristic and Metaheuristic Methods

Laporte, Toth, & Vigo (2013) mentioned in their article that VRP holds major place in transportation management and because of that carries worldwide faced VRP on a daily basis. VRP arises in several forms because of the variety of constraints encountered in practice. For over 50 years, the VRP is studied frequently in operational research community. This is due partly to the economic importance of the problem, but also to the methodological challenges it poses. TSP, which is a special case of VRP, can now be solved with very large instances and VRP is much more difficult solve to compare to TSP. For example, CVRP, which is simple case for VRP with constraints of capacity, is still difficult to solve with hundred customers by exact algorithms. Therefore, in recent years, researches are focused on to develop powerful metaheuristic.

Exact methods may not be sufficient for problems which have many instances or constraints. Solving these large problems with exact methods can take long time. Therefore, heuristic and metaheuristic methods are needed for solving problems in reasonable time even if exact methods guarantee to give an optimum solution.

In addition to reasonable solving time, other reasons for using heuristic methods are mentioned in book which is written by Martí & Reinelt (2011). These reasons are; unknown solving method for problem, unsuitable available hardware for exact methods, difficult to model constraints with exact methods and heuristic method can be used part of global procedure that guarantees optimum solution. Again, according to Martí & Reinelt (2011) a good heuristic method should provide reasonable computational effort, near optimal solution with high probability and probability of finding bad solution should be low.

Sörensen & Glover (2013) have explained metaheuristic as follows, ‘A metaheuristic is a high-level problem-independent algorithmic framework that provides a set of guidelines or strategies to develop heuristic optimization algorithms’. Metaheuristic is trying to find optimal solution from set of solutions with iterative operations which is a computational method. The most important feature of metaheuristic is that no need to any knowledge about the optimization problem. Genetic algorithm, ant colony algorithm, simulated annealing and tabu search are some of the most popular metaheuristics.

3.3.1 2-opt Heuristic

2-opt heuristic methods is a local search method that compare every possible pair of nodes by change them each other. This way the method aims to find optimal solution. 2-opt is first proposed by Croes (1958) for solving TSP and it is also widely preferred for VRP.

3.3.2 Lin-Kernighan Heuristic

Lin-Kernighan heuristic is one of the most effective algorithms which is first developed by Lin & Kernighan (1973). Lin-Kernighan algorithm is general version of 2-opt and 3-opt algorithm. It changes places all nodes and rearrange the sub-tour for finding better solution. Pseudo code for solving VRP with Lin-Kernighan Algorithm is as follows:

```
Generate route
while i < stop criteria
    Pick random node, k.
    for all nodes k+t, t=2
        if distance[k-1, k] - distance[k-1, k+t]>0
            Produce new route k→k+t→k+t-1→k+1-2→…→k+t+1→…
            if new route is an improvement
                Change old route with new route.
                Increase the node count t ← t + 1.
            else
                Increase the node count t ← t + 1.
        else
            Increase the node count t ← t + 1.
        Increase the stop count i ← i + 1.
```

3.3.3 Genetic Algorithm

Genetic algorithm is proposed by Holland (1975). Genetic algorithm is a population-based algorithm which is inspired by evolution. Genetic Algorithm starts with an initial set of solutions called a population. Individuals in the population is called a chromosome which represents a solution to the problem and are scored by its fitness function. Chromosomes are composed of genes representing of variables. The chromosomes evolve through iterations called generations by using a kind of "natural selection" together with the genetics−inspired operators of crossover, mutation, and inversion.

The large-scale NP-hard problems, like VRPs with great number of customers or constraints, are difficult to solve in acceptable time. Genetic Algorithm is evolutionary algorithm and proposed for searching near-optimal solution and has a few steps for achieve to this optimal solution. Steps of Genetic Algorithm are as flows:

- Representation of the problem
- Initial population generator
- Suitable fitness function
- Parent selection
- Crossover
- Mutation

These steps can be modified according to problem definition. In this way, different kind of problems can be solved effectively with Genetic Algorithm. General pseudo code for Genetic Algorithm is as follows:

```
Generate initial population
Evaluate fitness function
while i < stop criteria
    Select parents parent chromosomes
    Crossover parent chromosomes for children
        if Mutate probability is true
            Mutate child chromosomes
        else
            Don’t apply mutation
    Evaluate new population
    Increase the stop criteria i ← i + 1.
```

3.3.3.1 Representation of the Problem

Representation of the problem is one of the most important decision to implement Genetic Algorithm. An appropriate representation will ensure that the Genetic Algorithm works efficiently for the problem. Representation depends on the specifications of the optimization problem. There are some representations that are frequently used; binary representation, real valued representation, integer representation and permutation representation.

3.3.3.1.1 Binary Representation. In this representation method, genes are represented by string of binary variables. Each gene has a value of 0 or 1. The structure of the chromosome can be seen at Figure 3.1.

X_1	X_2	X_3	X_4	X_5	…	X_n
0	1	1	0	0	…	1

Figure 3.1 Binary representation of chromosome

When the solution space occurs of Boolean decision variables like yes or no, binary representation method can be used. If problem is assumed to be VRP in Figure 3.2, x_n is decision variables for customers and 0 means that nth customer is not visited, 1 means that customer is visited.

3.3.3.1.2 Real Valued Representation. Genes should be defined as continuous variables rather than discrete for some problems. Real-valued representation is the most effective method for these types of problems. Representation of the chromosome is in the Figure 3.2.

X_1	X_2	X_3	X_4	X_5	…	X_n
0.2	0.1	0.7	0.9	0.5	…	0.1

Figure 3.2 Real-valued representation of the chromosome

In this representation, the value of x_1 is 0.2 for this specific chromosome.

3.3.3.1.3 Integer Representation. In some cases, solution space cannot be limited to binary variables. For example; if status of the machine is to be encoded as occupied, available and broken, it can be encoded as 0, 1, 2. In this case integer representation can be appropriate. In Figure 3.3, the machine x_2 is occupied.

X_1	X_2	X_3	X_4	X_5	…	X_n
1	0	0	2	1	…	1

Figure 3.3 Integer representation of the chromosome

3.3.3.1.4 Permutation Representation. In many problems, the solution is represented by an order of elements. Permutation representation method is the most effective way for these types of problems.

Solution of the VRP is the order of the customers. In CVRP vehicle must visit each customer exactly once and all routes must start end finish at the depot. In this way, when vehicle visit all customers, there are a permutation of all customers and depot which to minimize the distance.

X_1	X_2	X_3	X_4	X_5			…				X_n
0	5	1	0	2	6	0	3	4	0	7	0

Figure 3.4 Permutation representation of the chromosome

In Figure 3.4, 0 represents to depot and there are 7 customers. 4 routes are determined for this solution. Solution of the VRP is naturally a permutation and therefore permutation representation can be used effectively.

3.3.3.2 Initial Population Generator

Population is the sum of the chromosomes in generation. Chromosomes in the population are directly related to solutions because of they are affected by each other. Therefore, generating initial population is important for efficiency of the Genetic Algorithm.

Size and diversity are the main concern at the initial population. If size of initial population is larger, Genetic Algorithm can slow down. On the other hand, when the size is too small, population is not enough for finding better solution by genetic operators. Optimum size of the population can be found by trial and error. Diversity of the population can increase different chromosomes. This increases the likelihood of finding better results.

There are two general method for determine to initial population: Random generator and heuristic methods. Random generator determines the chromosomes completely random. In heuristic method, chromosomes are determined by heuristic algorithm. That method can lead to all chromosomes in population is similar and diversity can be reduced. Thence combining these two methods is generally preferred.

3.3.3.3 Suitable Fitness Function

A fitness function is defined as a function that calculates to accordance of the solutions with the problem. In some cases, the fitness function and the objective function may be the same, while in others it might be different based on the problem. The chromosomes with the better fitness values stays in the population when the chromosomes with the worse fitness values are replaced with better ones.

3.3.3.4 Parent Selection

Parent selection is the cruel steps for maintaining diversity in the population. Selected parents are used for applying crossover. There are several methods for parent selection.

3.3.3.4.1 Roulette Wheel Selection. In this method, the wheel is divided according to fitness value of the chromosomes. Better fitness value has a greater pie on the wheel. Random number between 0 and 360 is used for selecting parents. Therefore, the probability of choosing an individual depends directly on its fitness.

3.3.3.4.2 Rank Selection. Every chromosome in the population are ranked according to their fitness function. Selection of the parents depend on this ranking. The higher ranked chromosomes are preferred more than the lower ranked ones.

3.3.3.4.3 Random Selection. Parents are selected random from the population. Because it is completely random, there is no pressure to improve the population.

3.3.3.5 Crossover

Genetic Algorithm mimics the process of natural evolution. Two parents are selected when new chromosomes are created. Then, these selected parent chromosomes are crossover. During this process, chromosomes have the possibility of being mutated. At the end of these steps better chromosomes survive and weak ones are eliminated from the population. These steps constitute genetic operators for Genetic Algorithm.

New chromosomes are created from selected parents by crossover operator. One or more child chromosomes are produced using the genetic material of the parent chromosomes. There are a lot of method in the literature for crossover. Some of these popular ones in literature as follows:

3.3.3.5.1 One-Point Crossover. A random point from the parent's chromosomes is selected and remaining parts of parents are swapped for creating new child chromosomes. There is a multi-point crossover and in this method two points are selected for swap. Evaluating child chromosomes by one-point crossover can be seen at Figure 3.5.

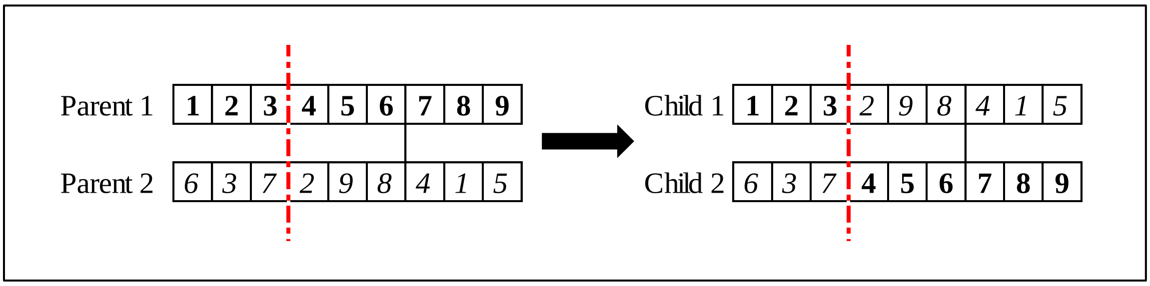

Figure 3.5 Child chromosomes by one-point crossover

3.3.3.5.2 Uniform Crossover. In uniform crossover method random genes of parent chromosomes are swapped between each other. Child chromosomes by uniform crossover is in Figure 3.6.

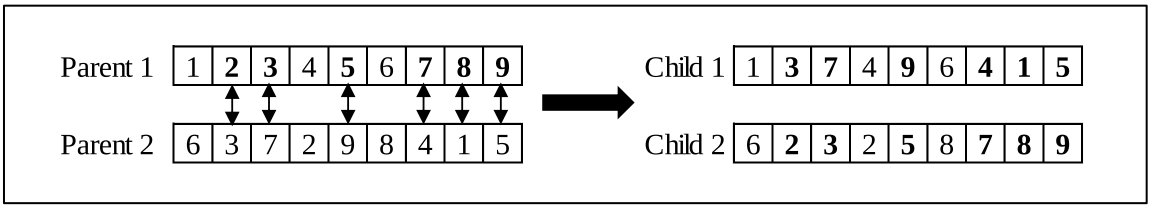

Figure 3.6 Child chromosomes by uniform crossover

3.3.3.5.3 Partially-Mapped Crossover (PMX). This method is proposed by Grefenstette, et al., (1985). A portion of parent chromosomes is exchanged. Then remaining genes of parent chromosomes are added to child chromosomes, taking care not to be the same as the first portion part. If one of the remaining genes are already in the portion, the corresponding gene added from the other parent chromosomes. In this way, two child chromosomes are obtained.

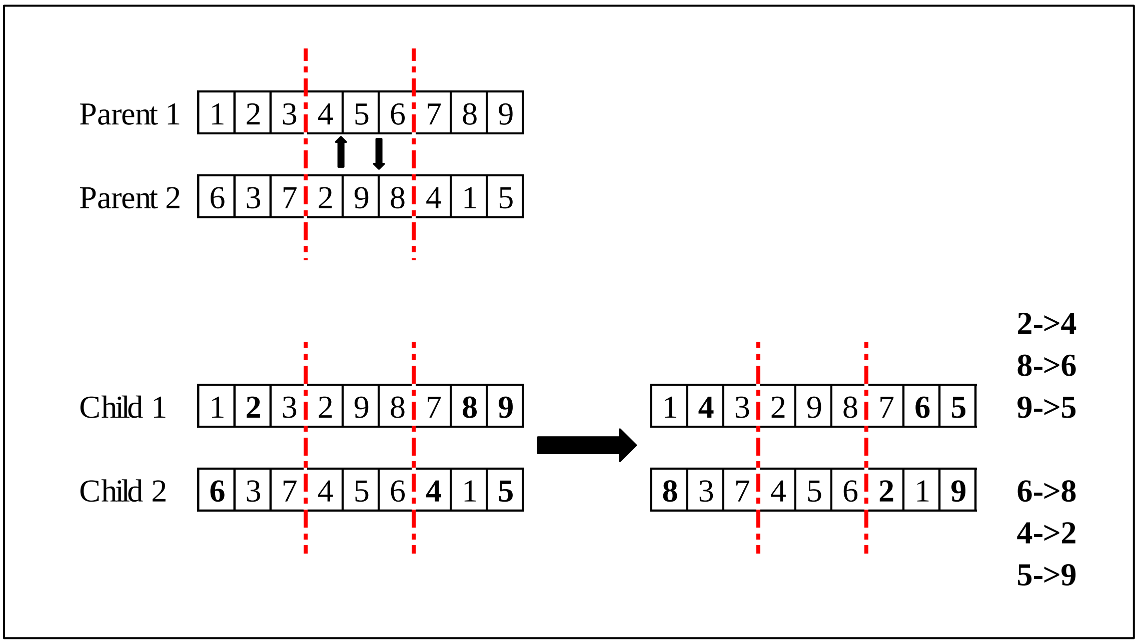

Figure 3.7 Parent chromosomes for PMX crossover

In Figure 3.7, PMX procedure is shown. For this example; 4, 5, 6 at parent 1 and 2, 9, 8 at parent 2 are selected for mapping. These portions of parents are swapped between each other. Then remaining parts are added child 1 and child 2. Second gene of child 1, which is represented by 2, is same as the first gene of the selected portion of parent 1. So, this gene is exchange with the first gene of the selected portion of parent 2 (2=>4).

PMX crossover method is suitable for VRP with permutation representation. Because with this method, all genes, in that case all customers, are used exactly once.

3.3.3.6 Mutation

Mutation can be defined simply as a small random modification in the chromosomes for obtain new solutions. Diversity can be increased with mutation. Like crossover, there are a lot of method for mutation and these methods can be alterable according to problem. Some of the commonly used methods are as follows:

3.3.3.6.1 Bit Flip Mutation. This method is used with binary representation method. One or more random genes are selected and modified. If gene is 1, it is transformed into 0. Representation of bit flip mutation is in the Figure 3.8.

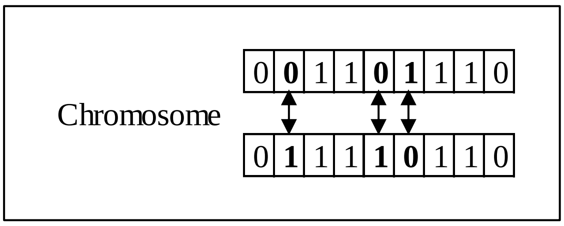

Figure 3.8 Bit flip mutation

3.3.3.6.2 Swap Mutation. Two random genes are selected and swapped between each other. This method is generally used with permutation representation. Figure 3.9 shows swap mutation.

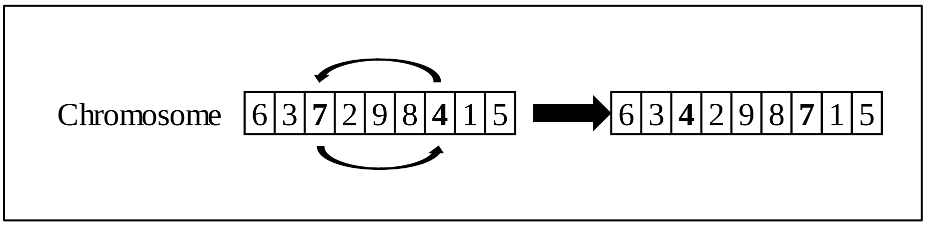

Figure 3.9 Swap mutation

3.3.3.6.3 Scramble Mutation. Scramble mutation is shown at Figure 3.10. Random portion of the chromosome is selected and genes in this portion are shuffled randomly. This method is also commonly used with permutation representation.

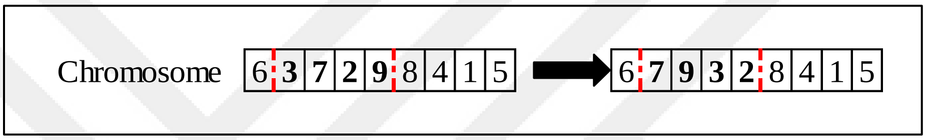

Figure 3.10 Scramble mutation

After crossover and mutation, chromosomes with better fitness values join the population and worse ones are eliminated.

These all steps of Genetic Algorithm repeat until termination condition is met. Determining the termination condition is important because of the running time of algorithm. If termination condition is set incorrectly, algorithm can still run after finding the optimal solution or it can be stop before find the optimal or best solution that algorithm can be achieved.

Termination condition is usually determined according to three condition. There has been no improvement in the population after certain numbers of iterations, algorithm can be terminated. Algorithm can be terminated after an absolute number of generations. Lastly, the objective function value has reached a pre-defined value, algorithm can be stopped.

Genetic Algorithm can be modified according to studied problem. All of the mentioned steps should be selected carefully for the efficiency of algorithm.

CHAPTER FOUR
INTEGER LINEAR PROGRAMMING FOR THE VEHICLE ROUTING PROBLEM WITH TRAFFIC CONDITIONS

4.1 Problem Definition

VRP, which is suitable for real-life situation is discussed in this study. In order to establish the best route except the traveled total distance, time must be taken into account for customer satisfaction. Vehicle traffic on the selected route will affect the time which is spent on the route, traffic plays an important role in determining the route. So, mathematical model is developed which is considering traffic conditions.

Proposed approach is based on VRPTW methodology and some new constraints are added for reflect the traffic effect.

4.1.1 Mathematical Model of the Problem

Mathematical model is developed by adding paramaters V_i and L_i, constraints 4.8 and 4.9 to the model of Kumar & Paneerselvam (2012) which is formulated for VRPTW.

Variables:

- X_{ij}, $\in \{0, 1\}$, 0 if there is no arc from node i to node j, and 1 otherwise
- T_i, arrival time at node i
- L_i, leaving time at node i
- V_i, time per km

Parameters:

- m_i, demand at node i
- *cap*, capacity of vehicle
- s_i, service time at node i
- *Tmax* maximum route time
- ER_i, earliest arrival time at node i
- LT_i, latest arrival time at node i
- $dist_{ij}$, distance between node i and node j
- *z*, time period
- *h*, time per km for time period
- *N*, total node
- *P*, total time period

Mathematical Model:

$$\text{Minimize} \sum_{i=0}^{N} \sum_{\substack{i=0 \\ i \neq j}}^{N} dist_{ij} V_i X_{ij} \tag{4.1}$$

Subject to

$$\sum_{j=1}^{N} X_{ij} = 1, \qquad for\ i = 0 \tag{4.2}$$

$$\sum_{j=1}^{N} X_{ji} = 1, \qquad for\ i = 0 \tag{4.3}$$

$$\sum_{\substack{i=0 \\ i \neq j}}^{N} X_{ij} = \sum_{\substack{j=0 \\ j \neq i}}^{N} X_{ij}, \qquad for\ i, j \in \{1, \dots, N\} \tag{4.4}$$

$$\sum_{i=0}^{N} X_{ij}\left(dist_{ij}V_i\right) + s_i + T_i \le T_j\,, \qquad for\; j = \{1, \dots, N\} \tag{4.5}$$

$$\sum_{i=1}^{N} m_i \sum_{\substack{j=0 \\ i\neq j}}^{N} X_{ij} \le cap \tag{4.6}$$

$$\sum_{i=0}^{N} \sum_{\substack{j=0 \\ i\neq j}}^{N} X_{ij}(dist_{ij}V_i) + s_i \le \; T_{max} \tag{4.7}$$

$$V_i = \begin{cases} L_i \ge z_1 & h_1 \\ L_i \ge z_2 & h_2 \\ \quad\;\; . & \\ \quad\;\; . & \\ L_i \ge z_P & h_p \end{cases}, \qquad i = \{1, \dots, N\} \tag{4.8}$$

$$(T_i + s_i) = L_i, \qquad i = \{1, \dots, N\} \tag{4.9}$$

$$ER_i \le T_i \le LT_i \tag{4.10}$$

$$X_{ij} \in \{0,1\} \tag{4.11}$$

The objective function, which is seen formulation 4.1, minimizes the total travelling time. The constraints 4.2 and 4.3 ensure that for each route, there is exactly one arc which is outgoing and incoming from the depot. These constraints ensure that a complete tour for each vehicle is ensured. The constraint 4.4 makes sure that each node is visited exactly once. The constraint 4.5 guarantees that the arrival time of each vehicle at the node j is less than the specified arrival time (T_j) at that node. The constraint 4.6 controls the vehicle capacity. Demand of the nodes in one route should equal or less than capacity. The constraint 4.7 ensures that the total time of travel of the route of each vehicle is less than or equal to the maximum route time (*Tmax*). Constraint sets 4.8 and 4.9 are added to reflect traffic conditions. Time per kilometer

(V_k) is determined according to time period z_p. Since the time period is determined hourly, leaving time at node i is calculated by dividing 60 in constraint 4.9. The constraint 4.10 ensures that arrival time of node i is between earliest and latest arrival times of node i. Last constraint 4.11 shows that X_{ij} is a binary variable. If X_{ij} is equal to 0 there is no arc from node i to j, 1 otherwise.

4.2 Example for Proposed Problem

A small example problem is performed according to developed mathematical model. This example problem includes 5 nodes which includes depot. Customer 1 represents the depot. All customers have demands, which have to be delivered, and time windows.

Proposed Problem is solved with Lingo optimization modeling software.

4.2.1 Data for Proposed Problem

Distance matrix is in Table 4.1, demands of the customers are in Table 4.2 and time windows are in Table 4.3 for the example problem. Capacity of the vehicle is 15 for this problem. Traffic volume changes throughout the day. Especially, morning and evening hours have high traffic volume due to working hours. In order to reflect this change to the problem, one day is divided into four according to working hours. Traffic has high volume at 6 am to 1 pm and 5 pm to midnight. Traffic has low volume at midnight to 6 am and 1 pm to 5 am. When the traffic volume is high, V_k variable is 1.2, in other case V_k variable is 1.

Table 4.1 Distance matrix for example problem

Distance Matrix	1	2	3	4	5
1	0	996	2162	1067	499
2	996	0	1167	1019	596
3	2162	1167	0	1747	1723
4	1067	1019	1747	0	710
5	499	596	1723	710	0

Distance matrix shows that distance between two customers. For example, the distance between customer 4 and customer 2 is 1019 kilometer.

Table 4.2 Demands of the customers

Customer	1	2	3	4	5
Demand	0	6	3	7	7

Customer 1 has a 0 demand because of it is depot. Customers have 6, 3, 7 and 7 demands respectively.

Table 4.3 Earliest, latest and service times for customers

Customer	1	2	3	4	5
Earliest Time	0	1000	2800	2000	1800
Latest Time	99999	2500	2900	3000	3900
Service Time	0	10	12	10	10

In Table 4.3, there are time windows for customers. Customer 2 don't accept vehicle up to 1000 seconds after vehicle starts its tour.

4.2.2 Results of the Problem

Objective function of the problem is minimizing the total distance and it is 6601 km. Selected routes are in Figure 4.1. There are two routes which don't exceed the vehicle capacity.

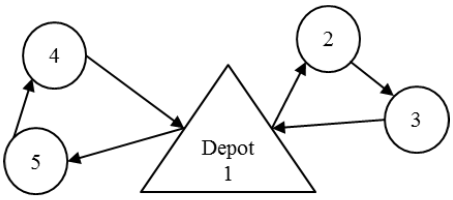

Figure 4.1 Selected routes

Example problem has 5 nodes and one day is divided into four sections. These instances are the maximum for solving the problem in acceptable time by exact methods which is applied with Lingo optimization modeling software. Example problem with 6 nodes is tried to solve but there is no feasible solution. Also, discussed problem has four section for one day. When these sections are increased, feasible solution cannot be found. If it is desired to increase instances, heuristic or metaheuristic methods may be considered. In Chapter Five of this study, metaheuristic method is developed for problems with larger instances and more constraints.

CHAPTER FIVE
COMPUTATIONAL STUDY FOR PROPOSED METHOD

5.1 Proposed Metaheuristic Method

CVRP and PDVRP is NP-hard problem like all VRPs, for this reason, only problem instances with smaller size can be solved with exact methods in acceptable time. Heuristic and metaheuristic methods allow to solve problem with the larger size of instances. Real life transportation problems can have many customers and companies must be able to find optimum routes fastest in order to use time efficiently. Therefore, various heuristic and metaheuristic algorithms are used in this research. Due to efficiency of Genetic Algorithm in solving combinatorial problems like VRPs it is used as based approach. Lin-Kernighan and 2-opt as local search algorithms are developed in this research to solve CVRP, PDVRP and VRP with Traffic Condition. The flow chart of the entire method can be seen at Figure 5.1.

All the steps of Genetic Algorithm are mentioned in Chapter Two. The method used for each step is shown in the Table 5.1.

Table 5.1 Genetic Algorithm steps for proposed method

Steps of Genetic Algorithm	**Method Used**
Representation of the chromosomes	Permutation representation
Initial population	Random Initialization + Heuristic Initialization
Fitness Function	Objective function
Parent Selection	Roulette wheel
Crossover	PMX crossover
Mutation	Swap mutation

As can be seen from the flow chart, Lin-Kernighan and 2-opt algorithms are used at different part of proposed method. Especially, Lin-Kernighan algorithm is used after random initialization of population. The aim of using Lin-Kernighan algorithm at this stage is to improve initial population. Also, Lin-Kernighan and 2-opt algorithm are used after mutation steps to provide diversity. One of these two algorithms is chosen according to random probability or algorithms may not be selected again this random probability. Iterations of the algorithms are finished according to their termination conditions.

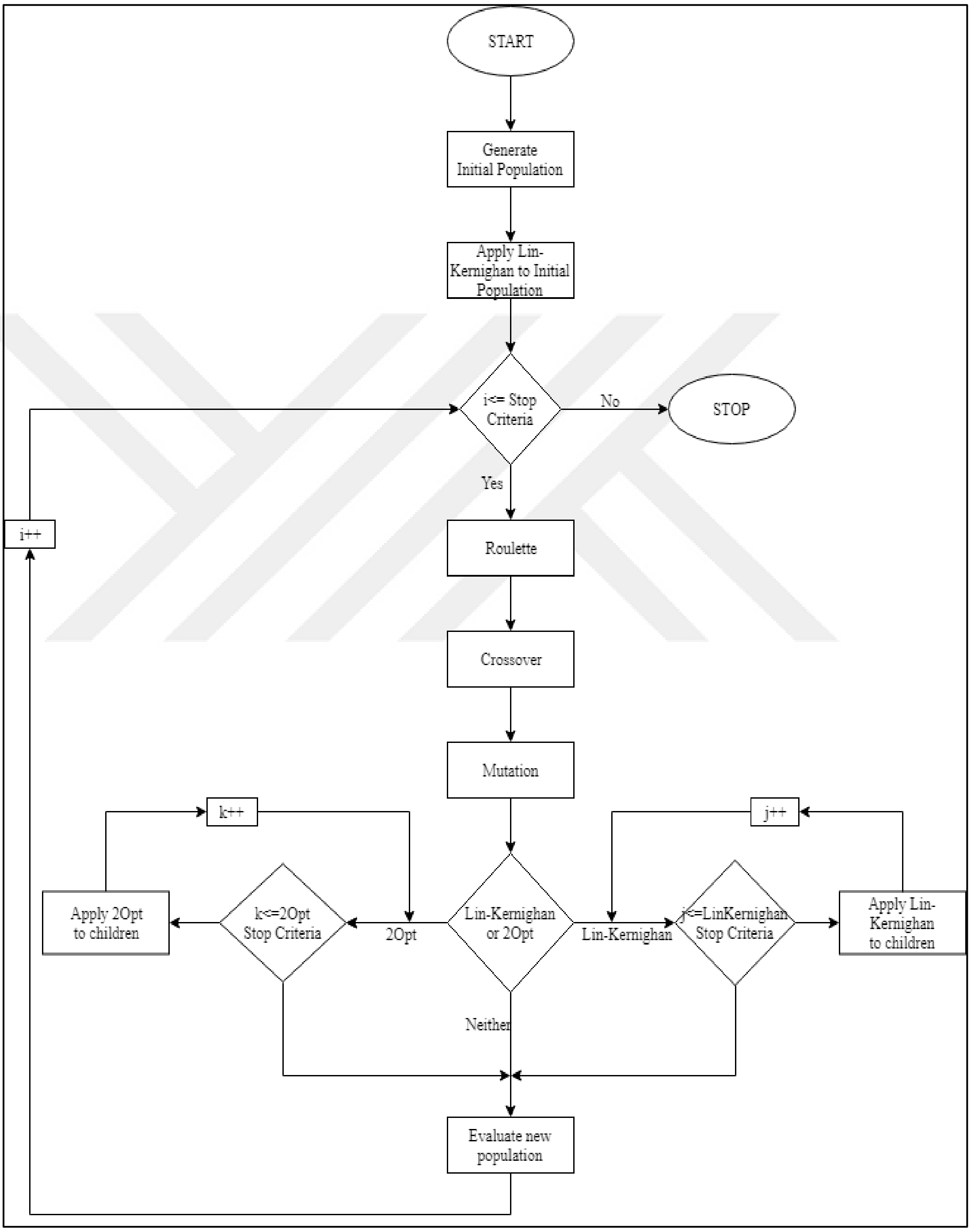

Figure 5.1 Flow chart for entire method

5.2 Application of the Proposed Method

The application is formulated in C# programing language at Visual Studio 2013 integrated development environment. Using these tools Windows Form application is developed. The whole view of the application can be seen Figure 5.2.

Parallel programming has been utilized to make the application work faster and more efficiently. Threads have been used to apply parallel programing to application. In this way, separate parts of proposed method are running simultaneously. These parts are distributed to the core by using threads in C#. As the number of cores in the used computer increases, the speed and efficiency of the application increases. As mentioned before, when the number of the instances increased, more time is needed to solve the problem. Through parallel programming, this effect of larger problems is reduced.

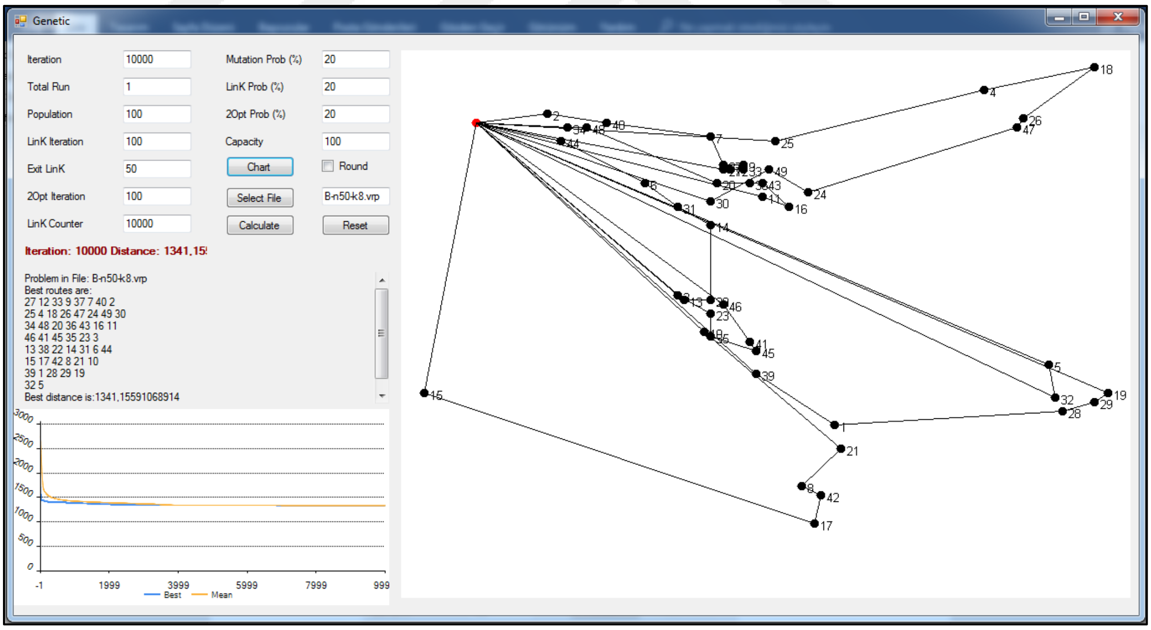

Figure 5.2 Application screen

Parameters of proposed method are set via the application. This part of application screen is in the Figure 5.3. Iteration box is used for termination condition. In Figure 5.3, after 10000 iterations algorithm is terminated. Total run box shows how many times the algorithm will work in a row. In that case, the algorithm will work 10000 x 20 times. Population size is determined by population box. LinK and 2-opt iteration boxes are used determine the termination conditions for these algorithms. Exit LinK box is used to prevent the Lin-Kernighan algorithm from working in vain. If algorithm cannot find better solution after the iterations until the value in the box, Lin-Kernighan algorithm will be terminated. In proposed solution method, Lin-Kernighan algorithm is applied initial population. Iteration for this stage is determined in LinK Counter box. Mutation Prob, LinK Prob, 2-opt Prob boxes are for probability of using these steps. LinK Prob and 2-opt Prob are related to each other. For the example in Figure 5.3, with a 20 percent probability Lin-Kernighan algorithm is applied, with a 20 percent probability 2-opt algorithm is applied and with a 60 percent probability none of these algorithms are applied. Capacity box represents the vehicle capacity for studied problem set.

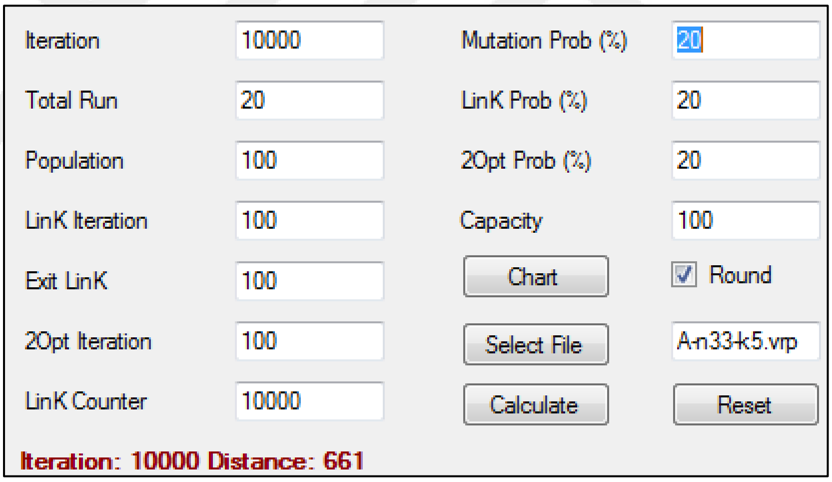

Figure 5.3 Parameter settings

Best and mean values according to iterations can be seen via chart button. Chart is shown in Figure 5.4. This button helps to determine number of iterations which is suitable for the problem. Round check-box is used for rounding distance value to closest integer.

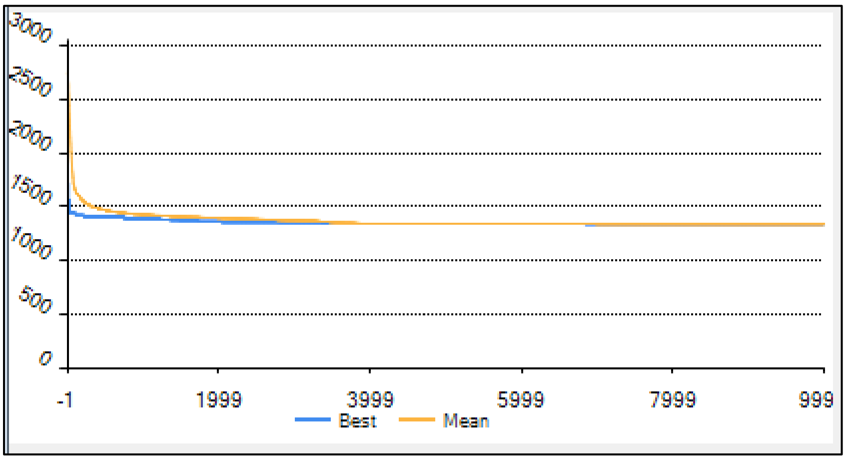

Figure 5.4 Iteration chart

Different data sets can be run via select file button. One file or the list of files can select for the application. When the list of files is selected, all data sets in the list are run one by one. Application starts with calculate button and reset button sets the default value of all parameter.

Solution of the problem appears both in writing and as a map. Routes and best distance value for this example can be seen Figure 5.5. Each lines of this part of application represent one route for best solution.

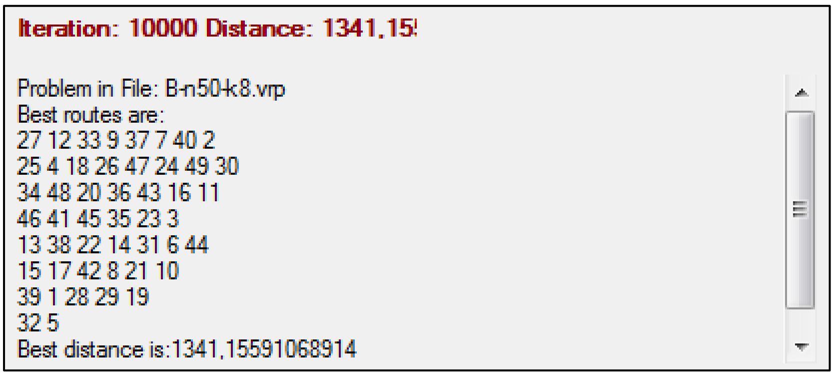

Figure 5.5 Routes and solution

Figure 5.6 represents to map version of the routes. The dot which is connected to all routes is depot and other ones are the customers. Customers are connected to each other with lines and it creates routes.

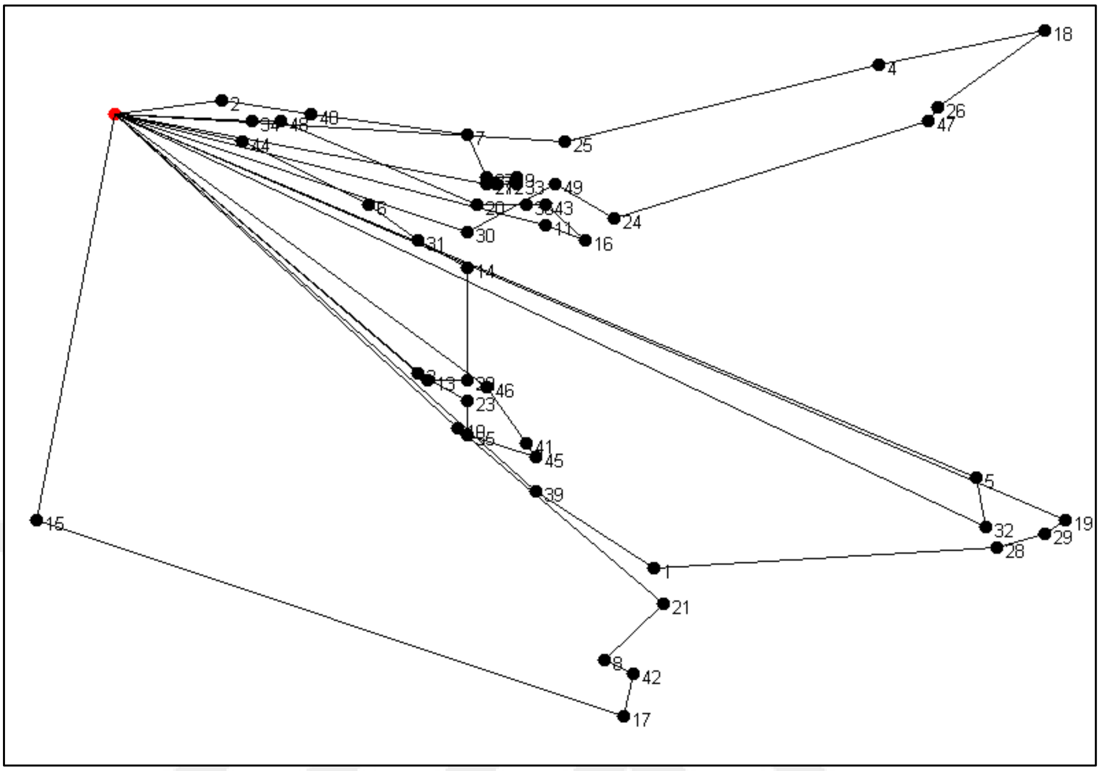

Figure 5.6 Map representation of routes

While the user interface is being prepared, it has been taken care that the user can change parameters and see the results through this interface. In this way, parameter settings for different problems can be made easily.

5.2.1 Computational Results for CVRP Data Sets

Proposed method and developed application are tested with different CVRP data sets. These test problems, which are A set, B set and P set, are selected set of CVRP problems from Augerat et al., (1995). Problems are tested with a computer which has 2.00 GB ram, dual core and 8.00GB ram, 8 cores. Best 20 results of 50 computations are taken into account. It takes 10 seconds to 1 minute to reach the solution according to problem size.

During the computational experiments, same parameter values are used. Used parameters are; population size is 100, number of iterations are 10000, mutation rate

20% and probability of applying Lin-Kernighan and 2-opt is 20%. Problems have been solved 20 times and the values in the tables are the averages of these solutions.

24 problems in A set have been solved. Results for these problems can be seen at Table 5.2 and Figure 5.7. Optimum result is achieved for A-n33-k5.vrp problem. The optimum route found is shown at Figure 5.8.

Table 5.2 Computational results for A data set

Problem Instance	**Vehicle Capacity**	**Optimum**	**Best**	**Mean**	**Worst**
A-n32-k5.vrp	100	784	797.87	808.15	816.60
A-n33-k5.vrp	**100**	**661**	661	670.92	689.84
A-n33-k6.vrp	100	742	742	745.03	752.28
A-n34-k5.vrp	100	778	780.94	789.94	799.75
A-n36-k5.vrp	100	799	802.13	813.95	837.90
A-n37-k5.vrp	100	669	678.00	691.18	725.09
A-n37-k6.vrp	100	949	952.47	966.62	993.59
A-n38-k5.vrp	100	730	734.18	740.13	746.10
A-n39-k5.vrp	100	822	829.52	840.70	848.00
A-n39-k6.vrp	100	831	835.28	846.74	873.96
A-n44-k6.vrp	100	937	939.26	959.55	1002.36
A-n45-k6.vrp	100	944	961.00	977.01	1040.51
A-n45-k7.vrp	100	1146	1174.66	1190.31	1232.60
A-n46-k7.vrp	100	914	919.38	927.94	939.44
A-n48-k7.vrp	100	1073	1107.44	1127.21	1175.76
A-n53-k7.vrp	100	1010	1039.66	1070.03	1160.23
A-n54-k7.vrp	100	1167	1195.51	1213.51	1234.70
A-n55-k9.vrp	100	1073	1084.45	1103.47	1117.42
A-n60-k9.vrp	100	1354	1387.93	1422.33	1439.43
A-n61-k9.vrp	100	1034	1061.47	1107.39	1184.41
A-n64-k9.vrp	100	1401	1337.98	1384.44	1432.47
A-n65-k9.vrp	100	1174	1668.26	1702.47	1726.86
A-n69-k9.vrp	100	1159	1365.03	1383.38	1398.77
A-n80-k10.vrp	100	1763	1453.07	1480.61	1497.01

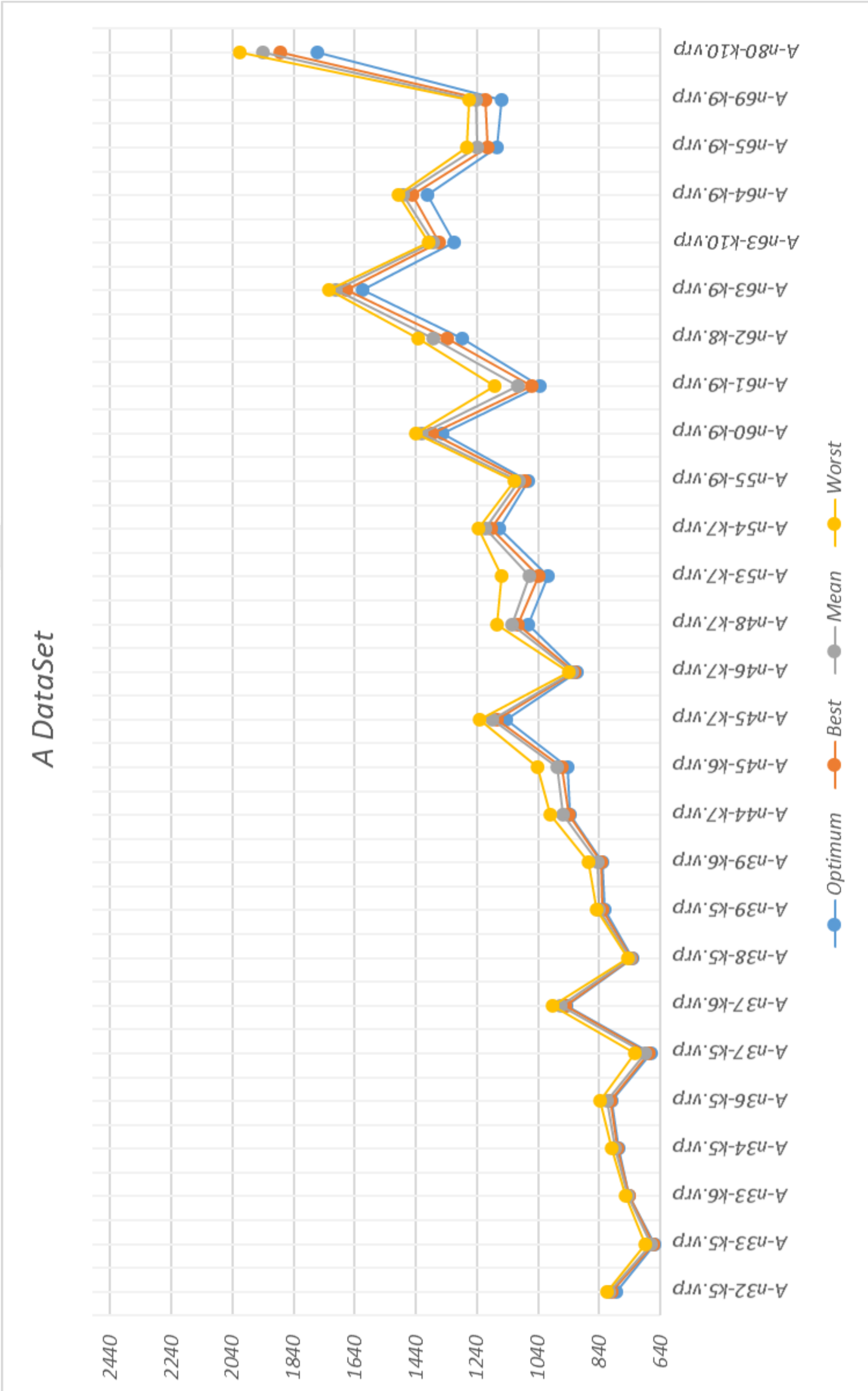

Figure 5.7 Graph of A Data Set Results

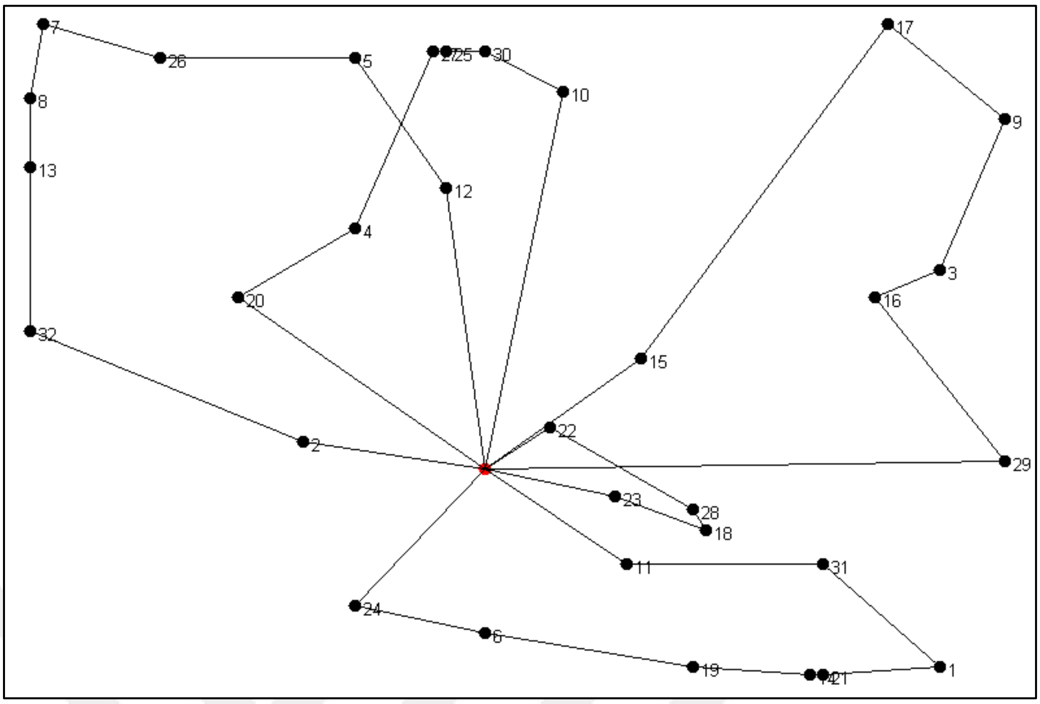

Figure 5.8 Optimum routes for A-n33-k5

The difference of the results from the optimum results is 3% on average. First 15 problems have customer nodes less than 50 and the difference of the results from the optimum results for these problems is 1.2% on average. According to this, proposed method is better for smaller instances.

22 problems in B set have been solved. Results for these problems can be seen at Table 5.3 and Figure 5.9. Optimum results are achieved for B-n31-k5.vrp and B-n41-k6.vrp problems. The optimum routes for B-n31-k5.vrp can be seen at Figure 5.10. Optimum routes are:

- depot, 29, 4, 25, 5, 18, 16, 21, depot
- depot, 7, 23, 8, 12, 28, 26, depot
- depot, 30, 13, 17, 9, 6, 22, depot
- depot, 3, 1, 19, 24, 11, 15, 14, depot
- depot, 20, 27, 10, 2, depot

Table 5.3 Computational results for B data set

Problem Instance	Vehicle Capacity	Optimum	Best	Mean	Worst
B-n31-k5.vrp	**100**	**672**	**672.00**	**673.61**	**676.09**
B-n34-k5.vrp	100	788	789.00	791.92	795.98
B-n35-k5.vrp	100	955	956.29	956.29	956.29
B-n38-k6.vrp	100	805	806.00	806.00	806.00
B-n39-k5.vrp	100	549	551.00	558.11	570.58
B-n41-k6.vrp	**100**	**829**	**829.00**	**841.24**	**868.22**
B-n43-k6.vrp	100	742	743.00	749.62	761.29
B-n44-k7.vrp	100	909	913.00	923.41	942.11
B-n45-k5.vrp	100	751	754.00	761.03	808.10
B-n45-k6.vrp	100	678	681.00	700.01	738.95
B-n50-k7.vrp	100	741	747.65	753.93	759.00
B-n50-k8.vrp	100	1312	1332.24	1344.72	1361.45
B-n52-k7.vrp	100	747	1019.33	1031.00	1057.73
B-n56-k7.vrp	100	707	749.00	759.13	776.98
B-n57-k7.vrp	100	1153	721.00	731.73	743.03
B-n57-k9.vrp	100	1598	1154.28	1167.32	1184.08
B-n63-k10.vrp	100	1496	1628.30	1651.31	1677.53
B-n64-k9.vrp	100	861	1555.41	1588.08	1625.11
B-n66-k9.vrp	100	1316	888.00	918.41	930.69
B-n67-k10.vrp	100	1032	1337.50	1359.63	1379.00
B-n68-k9.vrp	100	1272	1059.11	1103.06	1116.99
B-n78-k10.vrp	100	1221	1289.00	1323.69	1344.40

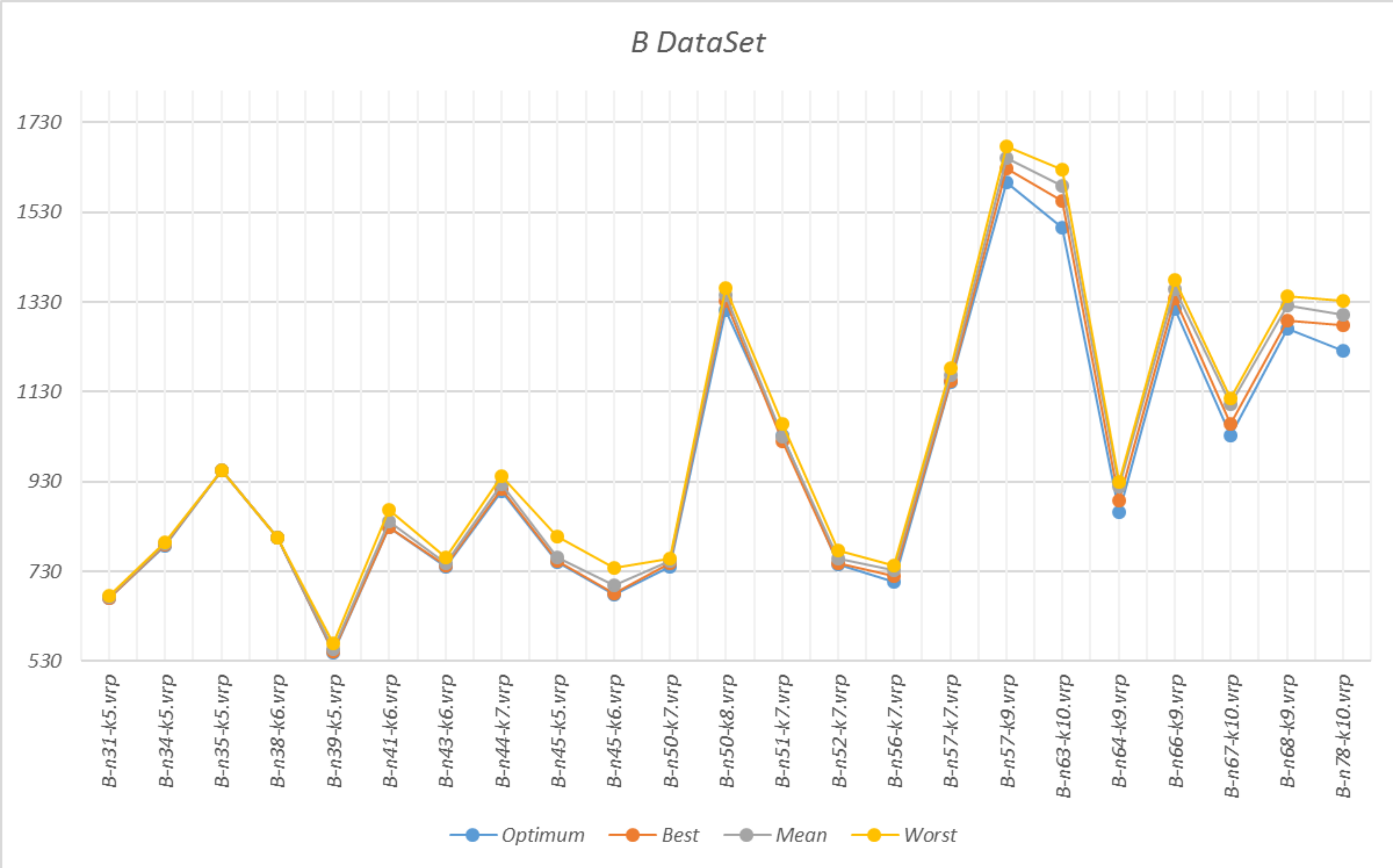

Figure 5.9 Graph of B Data Set Results

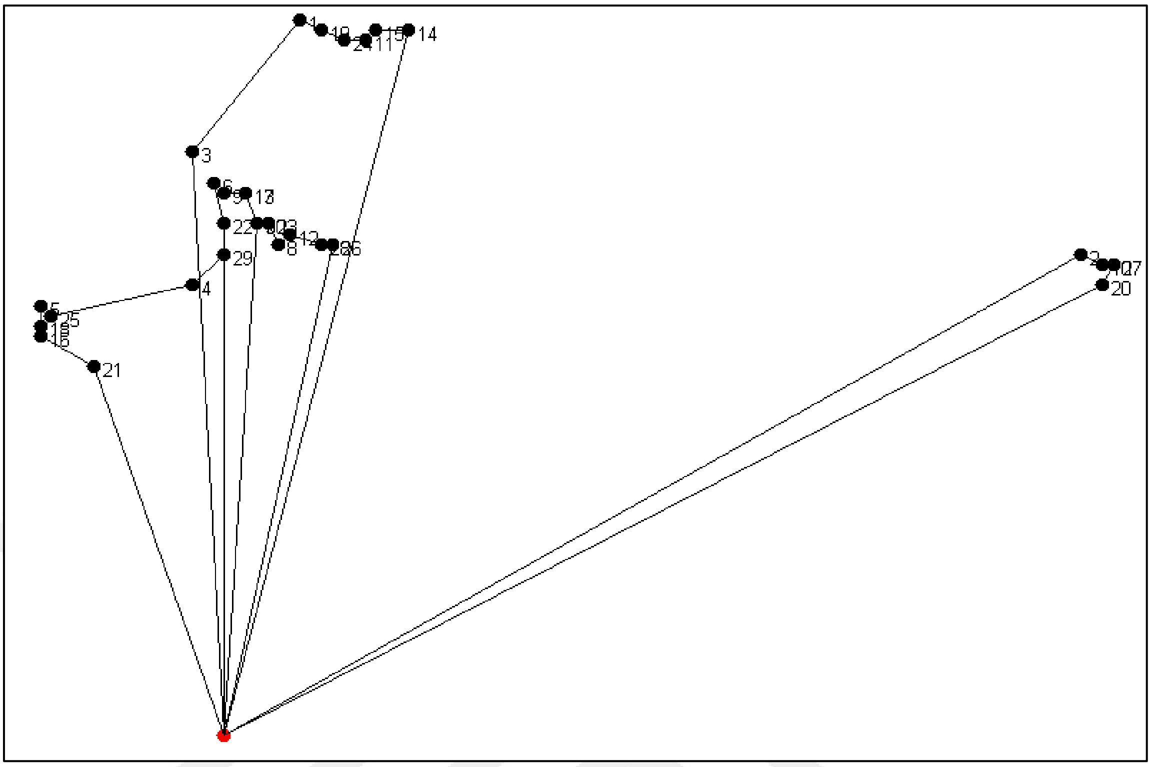

Figure 5.10 Optimum routes for B-n31-k5.vrp problem

Optimum distance for problem B-n41-k6.vrp is 829. In Figure 5.11, another solution for this problem can be seen. Distance for these routes are 831 and this result is different 0,2%. from optimum result. Routes for best distance is 831 as follows:

- depot, 13, 30, 36, 32, 17, 14, 29, 1, depot
- depot, 5, 15, 31, 22, 16, 8, depot
- depot, 12, 23, 10, 40, 20, 7, depot
- depot, 28, 11, 34, 33, 6, 25, depot
- depot, 24, 35, 2, 9, 37, 3, 18, 21, depot
- depot, 4, 39, 38, 27, 19, 26, depot

Routes for optimum distance are:

- depot, 5, 15, 31, 22, 13, 8, depot
- depot, 30, 36, 12, 32, 17, 14, 29, 1, depot
- depot, 23, 10, 40, 20, 16, 7, depot
- depot, 25, 6, 33, 34, 11, 28, depot
- depot, 24, 35, 2, 3, 18, 21, 37, 9, depot
- depot, 4, 39, 38, 27, 26, 19, depot

These two solution is slightly different from each other. The difference of the results from the optimum results is 2.5% on average. Results for smaller problem is better like A set.

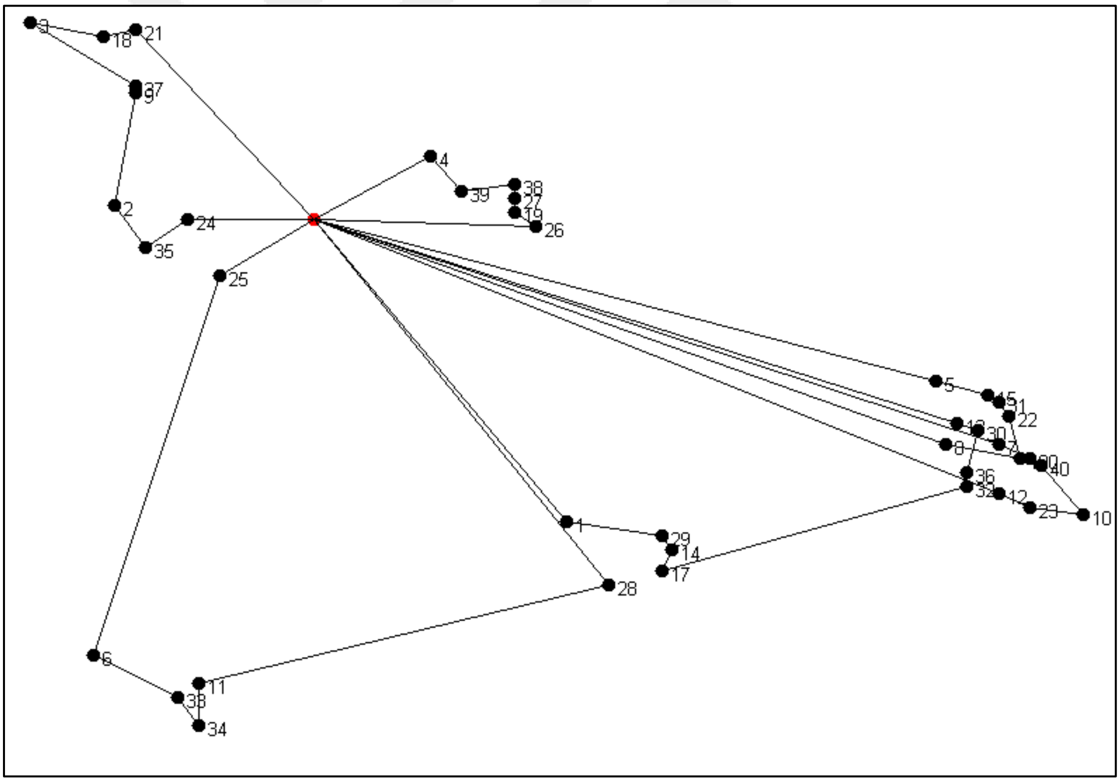

Figure 5.11 Near optimum routes for B-n41-k6.vrp

20 problems in P set have been solved. Results for these problems can be seen at Table 5.4 and Figure 5.12. Optimum results are achieved for first 6 problem. Optimum routes can be seen for P-n16-k8.vrp and P-n23-k8.vrp in Figure 5.13 and Figure 5.14.

Table 5.4 Computational results for P data set

Problem Instance	Vehicle Capacity	Optimum	Best	Mean	Worst
P-n16-k8.vrp	**35**	**450**	**450.00**	**450.07**	**451.34**
P-n19-k2.vrp	**160**	**212**	**212.00**	**212.03**	**212.66**
P-n20-k2.vrp	**160**	**216**	**216.00**	**216.25**	**219.94**
P-n21-k2.vrp	**160**	**211**	**211.00**	**211.09**	**212.71**
P-n22-k2.vrp	**160**	**216**	**216.00**	**216.61**	**217.87**
P-n23-k8.vrp	**40**	**529**	**529.00**	**529.11**	**531.17**
P-n40-k5.vrp	140	458	459.00	476.33	509.54
P-n45-k5.vrp	150	510	514.43	532.08	557.14
P-n50-k7.vrp	150	554	567.00	581.16	594.00
P-n50-k8.vrp	120	631	654.24	664.07	673.00
P-n50-k10.vrp	100	696	715.74	728.93	747.74
P-n51-k10.vrp	80	741	745.42	779.29	792.00
P-n55-k7.vrp	170	568	587.78	606.63	620.18
P-n55-k10.vrp	115	694	711.00	727.91	749.49
P-n60-k10.vrp	120	744	963.00	984.93	1022.26
P-n60-k15.vrp	80	968	756.03	787.39	806.25
P-n65-k10.vrp	130	792	1002.65	1022.91	1054.64
P-n70-k10.vrp	135	827	819.50	843.30	858.00
P-n76-k4.vrp	350	593	863.68	895.77	917.00
P-n76-k5.vrp	280	627	625.19	642.67	654.67

Best, mean and the worst values are the same for first problem in the Table 5.4. The reason of this is that the size of the problem is very small. All the chromosomes in the population have been improved in 10000 iterations. It means that less iteration is enough for this problem.

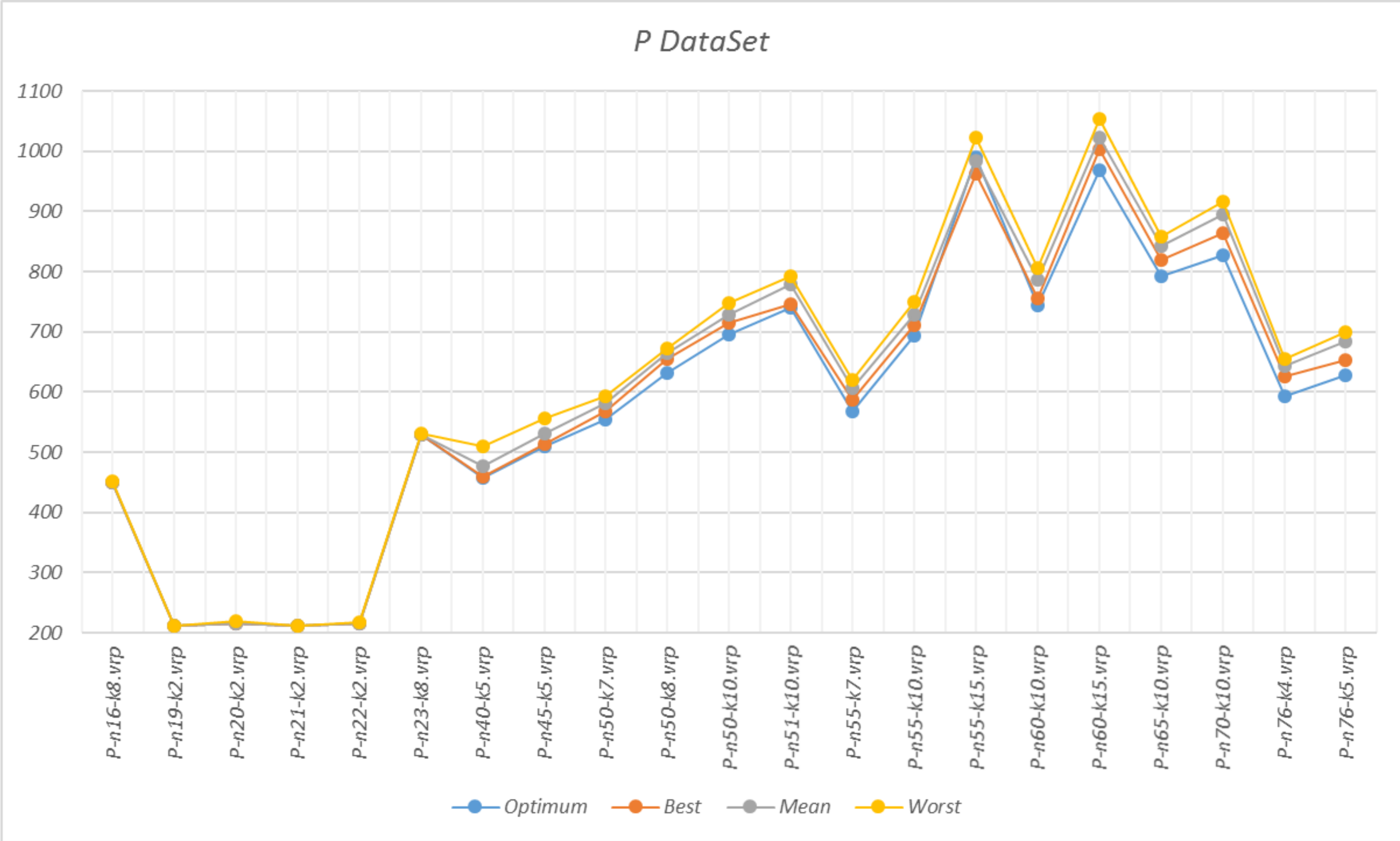

Figure 5.12 Graph of P Data Set Results

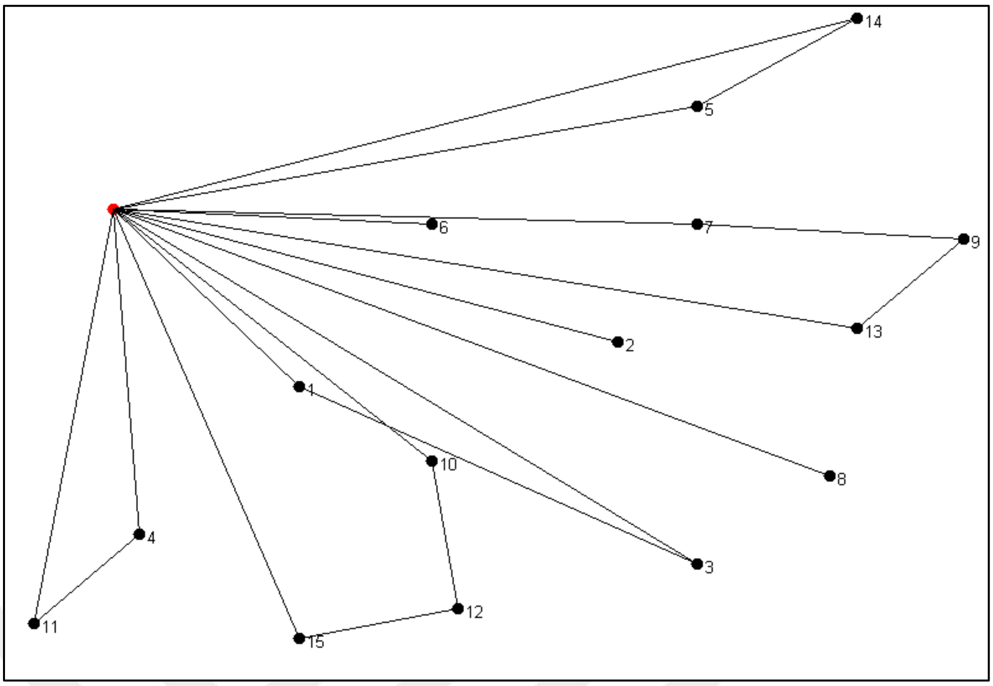

Figure 5.13 Optimum routes for P-n16-k8.vrp

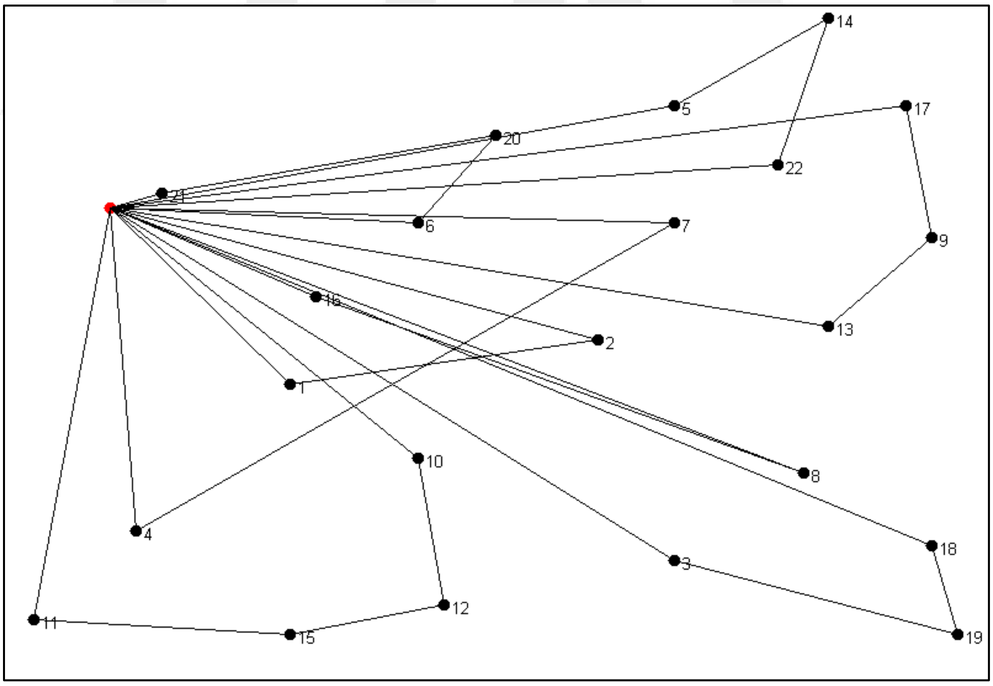

Figure 5.14 Optimum routes for P-n23-k8.vrp

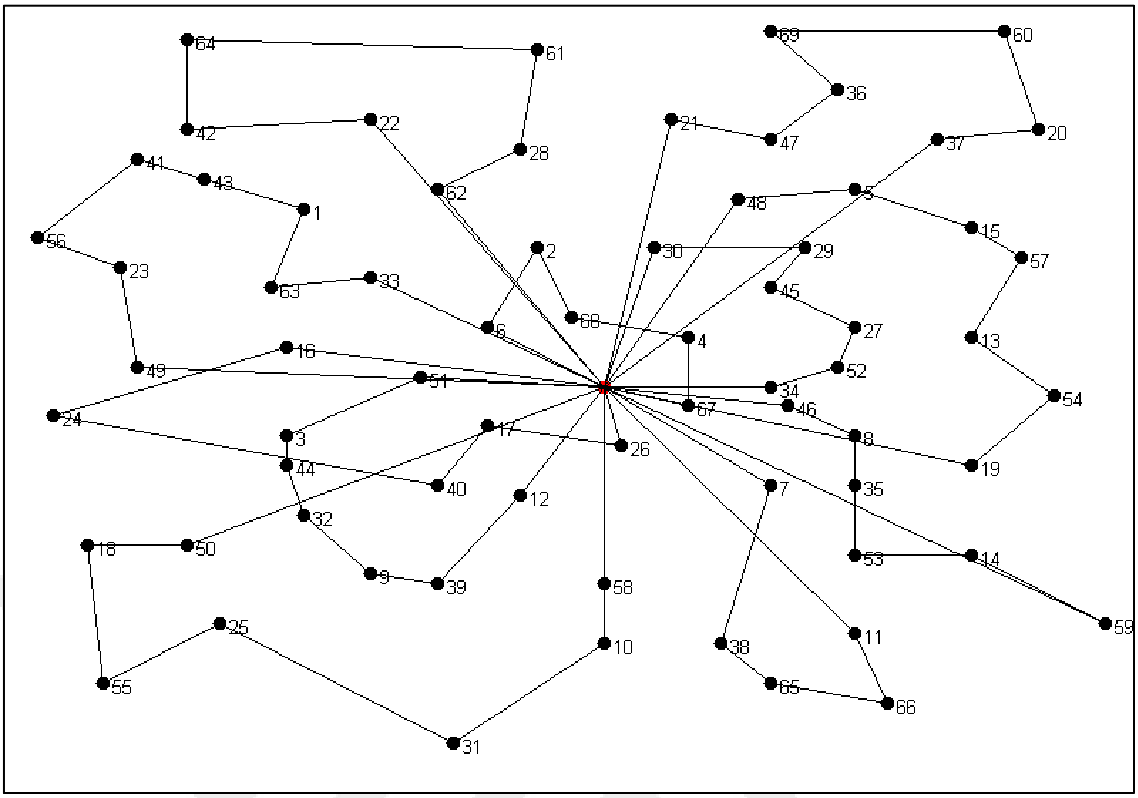

Figure 5.15 Near optimum routes for P-n70-k10.vrp

The difference of the results from the optimum results is 5% on average of all problems. P data set has larger size problems both of customer and capacity. In Figure 15, it can be seen that how complicated the problem P-n70-k10.vrp is.

Computational results show that proposed method is better for smaller problem. Even so, there are a near optimal solution for larger problem. All problems are solved with same parameter. Parameters setting can improve results for larger problems.

5.2.2 Computational Results for PDVRP Data Sets

Proposed method is used for PDVRP with some changes due to problem structure. PDVRP has additional capacity constraint. There are two type of demand and these are affecting the capacity differently. Nagy & Salhi (2005) mentioned that weak and strong feasibility in their article. In weakly feasibility, capacity is never exceeding with the total load of pick-up and delivery. In this study, both of weak and strong feasibility is ensured when the fitness function is calculated. Load factor of the vehicle is checked in every new customer node and the route never exceeds the vehicle capacity. If adding new customer to route exceeds the capacity of the vehicle, this new customer will not be added.

Breedam (1996) data set is used for testing the PDVRP in Table 5.5. Problems are tested with a computer which has 8.00 GB ram and 8 cores. All problems are same size, and they have 100 customer nodes. It takes 3 minutes to reach the solution due to large problem size. Results for test data are in Table 5.5.

Table 5.5 Computational results for proposed method

Problem Instances	**Proposed Method**		
	Best	**Mean**	**Worst**
1pp.dat	1136	1228	1305
2pp.dat	1637	1672.95	1722
3pp.dat	1778	1850.9	1942
4pp.dat	1546	1592.4	1656
5pp.dat	1001	1046.95	1080
6pp.dat	1029	1070.2	1134
7pp.dat	1049	1112.75	1181
8pp.dat	1027	1070.65	1133
9pp.dat	1852	1933.2	2026
10pp.dat	1124	1170.8	1254
11pp.dat	1265	1334.95	1412
12pp.dat	1558	1607.75	1662
13pp.dat	1024	1060.6	1120
14pp.dat	1898	1974	2045
15pp.dat	1013	1049.05	1113

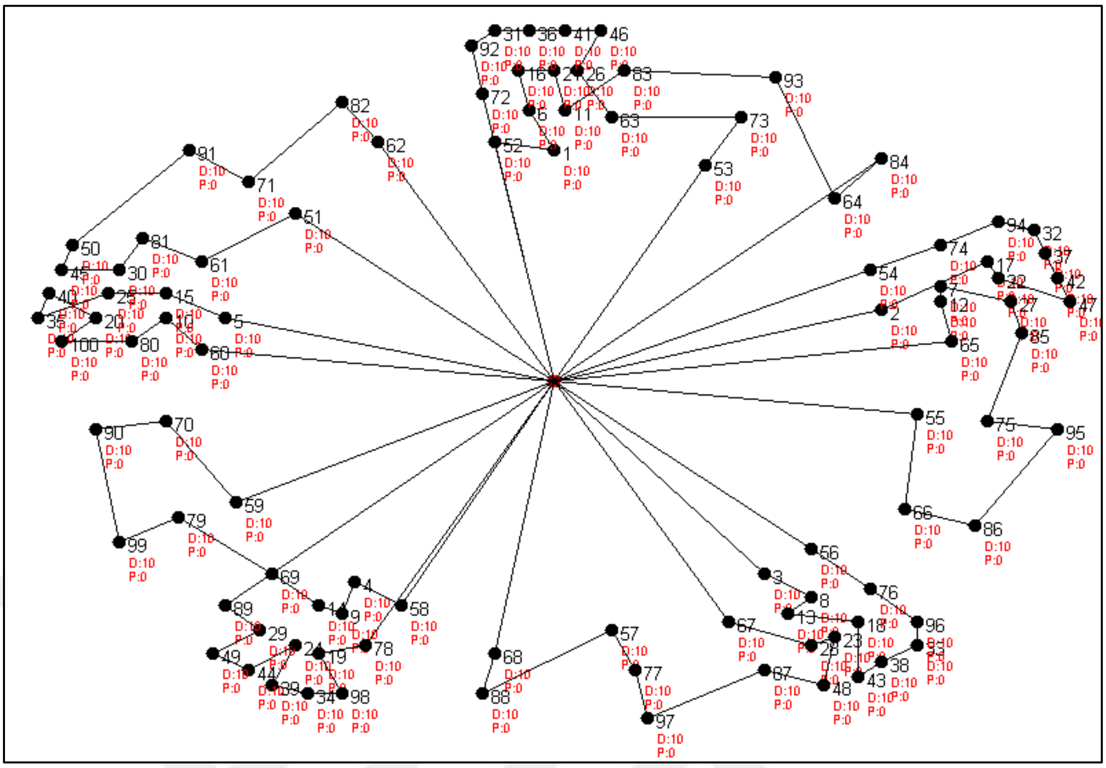

Figure 5.16 Routes for 1pp.dat problem

Figure 5.16 shows that routes for 1pp.dat problem. This solution is not the solution that mentioned in Table 5.5 and its distance is 1202. Delivery and pick-up values can be seen in the map.

Test data used for PDVRP has demand only pick-up or delivery for each customer. In this study, developed application allows to simulations pick-up and delivery operation. To test simulations pick-up and delivery random Pick-up demand which is between 0 and 26 are added to A, B and P data sets. Results for this data set can be shown at Table 5.6.

Table 5.6 Results of simultaneous Pick-up and Delivery

Problem Instance	Optimum of CVRP	Best	Mean	Worst
A-n32-k5.vrp	784	801.4	835.2	868.0
A-n33-k5.vrp	661	662.3	686.0	710.7
A-n33-k6.vrp	742	744.3	751.4	763.9
A-n34-k5.vrp	778	783.2	802.8	827.9
A-n39-k6.vrp	831	835.3	861.6	894.6
B-n31-k5.vrp	672	676.1	677.3	685.0
B-n34-k5.vrp	788	791.1	799.1	808.7
B-n35-k5.vrp	955	956.3	961.5	974.8
B-n38-k6.vrp	805	808.7	815.7	837.5
B-n39-k5.vrp	549	553.3	563.2	569.6
B-n41-k6.vrp	829	837.1	850.3	872.3
B-n43-k6.vrp	742	749.5	755.4	776.0
B-n44-k7.vrp	909	915.2	938.5	961.9
B-n50-k7.vrp	741	748.4	766.5	788.5
B-n57-k7.vrp	1153	1156.0	1181.9	1255.3
P-n16-k8.vrp	450	454.4	454.5	455.0
P-n19-k2.vrp	212	212.7	213.3	220.6
P-n20-k2.vrp	216	217.4	217.7	219.6
P-n23-k8.vrp	529	535.5	539.9	546.8

5.2.3 Computational Results for CVRP and PDVRP with Traffic Conditions

Constraints of traffic condition are added as described in the Chapter 4 to application. The distance is multiplied by a coefficient, which is represent traffic volume, while the objective function is being calculated. In that way, traffic condition is reflected to problem.

Developed application is used for solving real life problem with real distance values between 20 locations in Istanbul. These distance values are obtained from Google Maps. These locations are as follows:

- Hilton Hotel (Start Point - Depot)
- Modern Art Museum
- Topkapı Palace
- Mısır Bazaar

- Galata Tower
- Dolmabahçe Place
- Sultan Ahmet Mosque
- Gezi Park
- Kapalı Bazaar
- Ayasofya Mosque
- Beyoğlu
- Yedikule Zindanları
- Miniatürk
- Fatih Mosque
- Panorama 1453
- Eyüp Sultan Mosque
- Rahmi Koş Museum
- Yıldız Park
- Kariye Museum
- Exhibition and Trade Center Perpa

Coefficients of the traffic condition are determined according to graph in Figure 5.17 (Yandex, 2012).

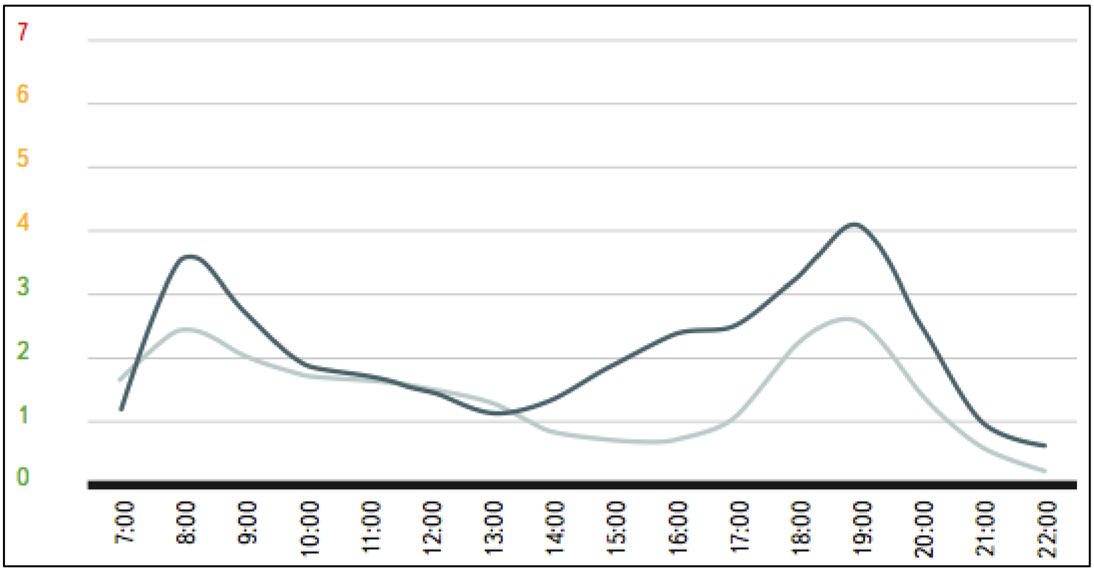

Figure 5.17 Graph for coefficient of the traffic condition in Istanbul both Europe and Asian sides

16 time periods are defined, and coefficients of these periods are in Table 5.7. Computational study has been made using this data. 8 real life problems have been studied and these problems are as follows:

- 10 nodes, CVRP, high traffic volume
- 10 nodes, CVRP, low traffic volume
- 10 nodes, PDVRP, high traffic volume
- 10 nodes, PDVRP, low traffic volume
- 20 nodes, CVRP, high traffic volume
- 20 nodes, CVRP, low traffic volume
- 20 nodes, PDVRP, high traffic volume
- 20 nodes, PDVRP, low traffic volume

Table 5.7 Coefficient for time periods

Time Period		**Coefficient**
07:00	07:30	1.75
07:30	08:30	3
08:30	09:30	2.5
09:30	10:30	2
10:30	11:30	1.75
11:30	12:30	1.5
12:30	13:30	1.25
13:30	14:30	1.37
14:30	15:30	1.8
15:30	16:30	2.3
16:30	17:30	2.75
17:30	18:30	3.25
18:30	19:30	4.75
19:30	20:30	2.5
20:30	21:30	1
21:30	07:00	0.5

Starting time is 7 a.m. for high traffic volume and 10 p.m. for low traffic volume. Capacities is 35 for CVRP and 50 for PDVRP with 20 nodes. When capacity is determined, P data set is considered because of the number of nodes. Demands are generated as uniformly distributed values between 0 and 30. In PDVRP version of the problem, pick-up and delivery are done simultaneously. There is only one vehicle and all routes are traveled with this vehicle. The elapsed time is calculated accordingly one vehicle.

As can be seen in Table 5.8 fitness values are higher when the traffic volume is high. The effect of traffic condition can be seen directly from the computational study.

Table 5.8 Computational results for VRP with traffic conditions

		High Traffic Volume		Low Traffic Volume	
		Best Fitness Value	**Routes**	**Best Fitness Value**	**Routes**
10	CVRP	77490	depot, 4, depot depot, 7, 1, 5, depot depot, 6, 3, depot depot, 2, 8, 9, depot	41164	depot, 4, depot depot, 3, 6, depot depot, 9, 8, 2, depot depot, 7, 1, 5, depot
	PDVRP	52198	depot, 6, 8, depot depot, 4, depot depot, 2, depot depot, 1, 7, 5, depot depot, 9, 3, depot	31937	depot, 2, 9, 8, depot depot, 7, 1, 5, depot depot, 4, depot depot, 3, 6, depot
20	CVRP	80962	depot, 9, 19, 18, 12, depot depot, 17, 13, depot depot, 10, depot depot, 15, depot depot, 3, 16, depot depot, 4, 8, depot depot, 2, 6, depot depot, 14, depot depot, 1, 5, 7, depot depot, 11, depot	38314	deot, 14, 3, 16, 11, 15, depot depot, 8, 4, 5, 2, 7, 10, depot depot, 1, 6, 9, 19, 18, 12, 17, 13, depot
	PDVRP	58005	depot, 2, 6, 9, 7, depot depot, 19, 18, 12, 11, depot depot, 10, 15, depot depot, 14, 3, depot depot, 17, 13, 16, depot depot, 5, 4, 8, depot depot, 1, depot	57927	depot, 19, 18, 12, 10, depot depot, 1, 4, 8, depot depot, 5, 2, 6, 9, depot depot, 7, 17, 16, depot depot, 14, 15, depot depot, 3, 13, 11, depot

CHAPTER SIX
CONCLUSION

The main objective of the VRP is minimizing the transportation cost. While achieving this objective, all constraints must be considered. In this study, adding traffic condition to VRP is the main concern because the traffic is daily struggle. Literature review was done, and most suitable methods were found to reflect traffic condition.

Proposed methods for CVRP, PDVRP and their variants, that consider traffic conditions, contribute both to the literature and daily life. Integer linear programming formulation as exact method and a metaheuristic method that combines 2-opt, Lin-Kernighan and Genetic Algorithm was developed.

Different data sets were studied with proposed method. Developing integer linear programing formulation has been a guide for metaheuristic method. A small problem was solved with integer linear problem and its results were sufficient for the problem studied. For larger problems, metaheuristic method was developed because VRP is a NP-Hard problem. To achieve optimum or near optimum solutions for larger problems in acceptable time, heuristic methods should be used. When metaheuristic method was developed, parallel programing method was used. In this way, speed and the efficiency of the method are increased.

Computational studies for CVRP were performed. The results of these studies are promising. Speed of the application and the results are acceptable. Parameter settings can be done via user interface. This allows to improve the efficiency of the application according to problem. Results of PDVRP are not as good as results of CVRP. To improve results for PDVRP parameter settings can be done.

For testing traffic condition effect on the CVRP and simultaneous PDVRP real life data was obtained. Results were suitable for proposed method. Traffic volume data is deterministic, and it only changes according to time. If real-time data can be obtained for traffic conditions, the results will be closer to real life. These problems are both in real life and the literature have importance in the transportation systems. Therefore, in this thesis, solution approaches are proposed for CVRP, PDVRP and their variants considering traffic conditions.

REFERENCES

Archetti, C., & Speranza, M. G. (2008). The Split Delivery Vehicle Routing Problem: A Survey. *Operations Research Computer Science Interfaces*, 103-122.

Augerat, P., Belenguer, J. M., Benavent, E., Corberan, A., Naddef, D., & Rinaldi, G. (1995). *Computational results with a branch and cut code for the capacitated vehicle routing problem. Research report 949-M.* Grenoble, France: Universite Joseph Fourier.

Balas, E., & Toth, P. (1983). *Branch and bound methods for the traveling salesman problem.* Pittsburg: Carnegie-Mellon University.

Bell, J. E., & Griffis, S. E. (2010). Swarm Intelligence: Application of the ant colony optimization algorithm to logistics-oriented vehicle routing problems. *Journal of Business Logistics*, *31*, 157-175.

Braekers, K., Ramaekers, K., & Nieuwenhuyse, I. V. (2016, September). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering*, *99*, 300-313.

Breedam, A. V. (1996). An analysis of the effect of local improvement operators in genetic algorithms and simulated annealing for the vehicle routing problem. *KPMG Orinoco*.

Carpaneto, G., & Toth, P. (1980). Some new branching and bounding criteria for the asymmetric travelling salesman problem. *Management Science*, *7*, 736-743.

Cordeau, J., Laporte, G., Savelsberg, M. W., & Vigo, D. (2007). Vehicle Routing. In J. Cordeau, G. Laporte, M. W. Savelsberg, D. Vigo, C. Barnhart, & G. Laporte (Eds.), *Handbooks in Operations Research and Management Science: Transportation*, *14*, 367-428. Amsterdam, Netherlands: Elsevier.

Croes, G. A. (1958). A method for solving traveling salesman problems. *Operations Research*, *6*, 791-812.

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, *6*, 80-91.

Desrochers, M., Desrosiers, J., & Solomon, M. (1992). A new optimization algorithm for the vehicle routing problem with time windows. *Operations research, 40*, 342-354.

Eksioglu, B., Vural, A. V., & Reisman, A. (2009). The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering, 57*, 1472-1483.

Grefenstette, J. J., Gopal, R., Gucht, D. V., & Rosmaita, B. (1985). Genetic Algorithms for the Traveling Salesman Problem. *Proceedings of the 1st International Conference on Genetic Algorithms.*

Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence.* Oxford: U Michigan Press.

Kumar, S. N., & Panneerselvam, R. (2012). A Survey on the Vehicle Routing Problem and Its Variants. *Intelligent Information Management, 4*, 66-74.

Laporte, G., Mercure, H., & Nobert, Y. (1986). An exact algorithm for the asymmetrical capacitated vehicle routing problem. *Networks, 14*, 161-172.

Laporte, G., Toth, P., & Vigo, D. (2013). Vehicle routing: historical perspective and recent contributions. *EURO Journal on Transportation and Logistics, 2*, 1-4.

Lin, S., & Kernighan, B. W. (1973). An effective heuristic algorithm for the traveling-salesman problem. *Operation Research, 21*, 498-516.

Martí, R., & Reinelt, G. (2011). *The linear ordering problem: exact and heuristic methods in combinatorial optimization, 175*. Berlin: Springer-Verlag Berlin Heidelberg.

Nagy, G., & Salhi, S. (2005). Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries. *European Journal of Operational Research, 162*, 126-141.

Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research, 34*, 2403-2435.

Sörensen, K., & Glover, F. (2013). Metaheuristics. *Encyclopedia of Operations Research and Management Science*, 960–970.

Surekha, P., & Sumathi, S. (2011). Solution to multi-depot vehicle routing problem using genetic algorithms. *World Applied Programming, 1*, 118-131.

Tasan, A. S., & Gen, M. (2012). A genetic algorithm based approach to vehicle routing problem with simultaneous pick-up and deliveries. *Computers & Industrial Engineering, 62*, 755-761.

Yandex. (2012). *Araştırmalar ve İnfografikler / İstanbul'da Trafik Durumu*. Retrieved June 13, 2012 from Yandex: https://yandex.com.tr/company/press_center/infographics/traffic2012