ENHANCEMENTS OF CLARKE-WRIGHT SAVINGS HEURISTICS FOR THE CAPACITATED VEHICLE ROUTING PROBLEM

by

TAMER DOYURAN

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

SABANCI UNIVERSITY

Spring 2008

ENHANCEMENTS OF CLARKE-WRIGHT SAVINGS HEURISTICS FOR THE CAPACITATED VEHICLE ROUTING PROBLEM

APPROVED BY:

Associate Prof. Bülent Çatay..
(Thesis Supervisor)

Assistant Prof. Tonguç Ünlüyurt...

Assistant Prof. Kerem Bülbül..

Assistant Prof. Güvenç Şahin..

Assistant Prof. Hüsnü Yenigün...

DATE OF APPROVAL: ………………………………………………………….

ACKNOWLEDGEMENTS

I would like to thank my thesis advisor Dr. Bülent Çatay for his invaluable guidance, motivation, encouragement and patience. It has been a great honour for me to study under his supervision.

I would also like to thank the committee members of my thesis, Dr. Kerem Bülbül, Dr. Tonguç Ünlüyurt, Dr. Güvenç Şahin and Dr. Hüsnü Yenigün. For the financial support during my graduate studies, I thank TUBITAK (The Scientific and Research Council of Turkey).

Special thanks go to my friends Salih Sarı, Mustafa Barbaros Özkök and Çiğdem Bağbakan. I appreciate their friendship. I am indebted to them for providing me a pleasant working environment in their dormitory room and apartment which made my studies and life very joyful.

I also present sincere thanks to my friend Emre Şahin for his support to implement the algorithm in C++.

Finally, I would express my gratitude to my family. Without their encouragement and spiritual support, it would be very hard to accomplish my studies.

ENHANCEMENTS OF CLARKE-WRIGHT SAVINGS HEURISTICS FOR THE CAPACITATED VEHICLE ROUTING PROBLEM

Tamer Doyuran

IE, M.S. Thesis,2008

Thesis Supervisor: Doç. Dr. Bülent Çatay

Keywords: Vehicle routing problem, Clarke-Wright algorithm, heuristics

ABSTRACT

We address the Clarke-Wright (CW) savings algorithm proposed for the Capacitated Vehicle Routing Problem (CVRP). In the past,

Gaskell (1967), Yellow (1970), Paessens (1988), and Altınel and Öncan (2005) proposed modifications on the CW function by either parameterizing it or by adding new parameterized terms. The primary objective of all these approaches is to obtain short tours with least computational effort. In this study, we propose several enhancements to the two- and three-term versions of CW savings function. Our aim is to further improve the solution quality without bringing additional computational burden to the existing approaches. To test the performance of our savings functions, we conduct an extensive computational study on a large set of well-known instances from the literature. These instances were also used by the earlier savings algorithms that we benchmark our approach with. The results show that the proposed savings functions provide shorter distances in many instances and the average performance is better than those of the previous approaches reported in the literature.

KAPASİTE KISITLI ARAÇ ROTALAMA PROBLEMLERİ İÇİN KULLANILAN CLARKE-WRIGHT TASARRUF YÖNTEMİNİN PERFORMANSININ ARTTIRILMASI

Tamer Doyuran

IE, Yüksek Lisans Tezi,2008

Tez Danışmanı: Doç. Dr. Bülent Çatay

Anahtar Kelimeler: Araç rotalama problemi, Clarke-Wright algoritması, sezgisel yöntemler

ÖZET

Bu çalışmamızda, Kapasite Kısıtlı Araç Rotalama Problemleri için kullanılan Clarke-Wright (CW) tasarruf algoritmasına değiniyoruz. Geçmişte, Gaskell (1967), Yellow (1970), Paessens (1988) ve Altınel-Öncan (2005) tasarruf formülündeki terimleri parametrize ederek ve bu formüle yeni terimler ekleyerek CW algoritmasına yeni açılımlar getirmişlerdir. Tüm bu yeni açılımların başlıca amacı, hızlı hesaplama süresinde kısa rotalar bulmaktır. Çalışmamızda, CW tasarruf formülünün iki ve üç terimli versiyonlarının performansını arttıran çeşitli yaklaşımlar önerdik. Amacımız, ek bir hesaplama yükü getirmeden çözüm kalitesini arttırmaktır. Algoritmayı, literatürdeki en bilinen örnek setleri üzerinde test ettik. Bu örnek setleri, daha once önerilmiş ve bizim de karşılatırmalı değerlendirme yaptığımız tasarruf formüllerini test etmek için de kullanılmıştır. Algoritmamız, birçok problem için literatürdeki diğer sezgisel yöntemlerden daha iyi sonuçlar vermektedir.

Anahtar Kelimeler: Araç rotalama problemi, Clarke-Wright algoritması, sezgisel yöntemler

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The Capacitated Vehicle Routing Problem (CVRP) is a well-known NP-hard problem (Lenstra and Rinnooy, 1981) introduced first by Dantzig and Ramser (1959). It has attracted a lot of attention since then because of its applicability to many practical settings, and various variants have been proposed for different environments such as VRP with time-windows, VRP with pick-up and delivery, stochastic VRP, etc.

In the classical CVRP, n customers are serviced with a fleet of K identical vehicles. The vehicles are based at a central depot and have capacity C. Each customer has a non-negative demand d_i, which is known in advance and cannot be split. The objective is to determine a set of vehicle routes originating and terminating at the depot such that all customers are visited exactly once, the total demand of the customers assigned to one route do not violate the capacity of the vehicle, and the total distance travelled by all vehicles is minimized. In some cases, a maximum length restriction D may be associated with each vehicle route.

In the literature, exact algorithms, heuristics, and metaheuristics have been proposed for solving the CVRP. Exact algorithms such as column generation, dynamic programming, and relaxation-based methods are inefficient since the computational time grows exponentially as the size of the problem increases and/or as new constrains are introduced. Since these algorithms cannot be applied to large practical instances, significant research effort has been spent on heuristic methods to find good quality solutions fast. Until 1990s, the researchers have extensively studied heuristic methods. Heuristics are easy to implement; they produce fairly good quality solutions, and require lower computing times. Furthermore, they are flexible and can easily be adjusted to handle various constraints in real-life applications. On the other hand, due to the limited exploration of the search space, the solution quality of the heuristics is poorer compared to the solution quality of metaheuristics. Hence, the research on metaheuristic approaches for CVRP problems has gained momentum in recent years with the development of new and efficient techniques. With metaheuristics, the exploration of the solution space is performed deeply by the integration of complex neighbourhood search methods and memory structures. Moreover, the metaheuristics do not only accept the solutions improving the objective function but they also allow some infeasible moves in order to escape from the local minima. Nevertheless, better solution quality of the metaheuristics is achieved at the expense of more computational efforts. As reported in Laporte et al. (2000), Toth and Vigo (2001), and Cordeau et al. (2004) metaheuristcs such as tabu search, genetic algorithm, simulated annealing, deterministic annealing, ant colony systems, neural networks have been successfully applied to CVRP and its variants. Since metaheuristic approaches are beyond the scope of this study, further discussion is skipped and the interested readers are referred to the above mentioned papers.

An extensive study about the classical heuristics for the CVRP can be found in Laporte and Semet (2001). Laporte and Semet classified the classical heuristics in three categories: constructive heuristics, two-phased heuristics, and improvement heuristics. In the constructive heuristics, a feasible solution is constructed iteratively by checking the routing cost. In the two-phased heuristics, the customers are first clustered into feasible routes and then the final tours are constructed by considering the clustered routes. Finally, in the improvement heuristics the feasible solution is improved by exchanging edges or group of edges within or between the tours. Among constructive heuristics, the well-known Clarke and Wright (CW) savings algorithm (1964) uses a saving criterion for combining the routes. First, it calculates the cost savings of using one vehicle rather than two for servicing a pair of customers and sorts them in the non-increasing order. Then, the routes are constructed by merging customer pairs starting from the top of the savings list. The algorithm continues to merge feasible customers until all customers are assigned to a route.

CW algorithm is widely used even in the commercial packages due to its simplicity, accuracy, flexibility, and speed. Several enhancements of CW algorithm have been proposed in the literature by parameterizing the savings formula and adding new terms to it. In this study, we propose several enhancements to the two- and three-term versions of CW savings function. Our aim is to further improve the solution quality without bringing additional computational burden to the existing approaches. The thesis is organized as follows. Chapter 2 reviews the related literature about CW algorithm and its enhancements. The proposed enhancements to the two-term version of CW savings function and the computational study are given in Chapter 3. In Chapter 4 the drawbacks of the Altınel and Öncan's formula are discussed, new enhancements for this three-term version of CW savings function and the computational study are provided. Finally, contributions of the proposed savings functions are summarized, and concluding remarks are given in the last chapter.

CHAPTER 2

LITERATURE REVIEW

In this section, the metaheuristics and classical heuristics for the CVRP are discussed. In particular, Clarke-Wright savings heuristic and its enhancements, which are the primary focus of this paper, are explained in detail.

The CW algorithm is one of the earliest and most widely used heuristics for solving the CVRP. Two versions of the CW algorithm are proposed in the literature: parallel and sequential. The best feasible merge of sub-tours are performed in the parallel approach whereas the route extension is considered in the sequential approach. The parallel version dominates the sequential savings method, as stated in Laporte and Semet (2001). So, the parallel approach is adapted to the implementations in this study. The algorithm starts with computing the cost savings of using one vehicle rather than two for servicing all customer pairs i and j. The savings formula is as follows:

$$\begin{aligned} S_{ij} &= (c_{0i} + c_{i0} + c_{0j} + c_{j0}) - (c_{0i} + c_{ij} + c_{j0}) \\ &= c_{i0} + c_{0j} - c_{ij} \end{aligned} \quad (1)$$

where c_{i0} (c_{0j}) is the distance of customer i (j) to the depot and c_{ij} represents the distance between the customer i and j.

The algorithm calculates the savings values between the customer pairs using formula (1), and sorts them in the non-increasing order. It initially places each customer into the route of a separate vehicle. Since this requires one vehicle for each customer, implying a fleet of n vehicles, it is not a desirable solution. The next phase attempts to eliminate one vehicle at each iteration by serving customers i and j with the same vehicle if the total demand of the customers does not violate the capacity constraint and the total distance traversed by the vehicle does not exceed the route length constraint, if any. In the parallel version of CW algorithm, the customer pair i and j providing the highest saving value is considered and the feasibility of merging the two feasible routes, one starting with $(0,j)$ and another ending with $(i,0)$ is checked. If feasible, these two routes are combined by deleting $(0,j)$ and $(i,0)$ and inserting (i,j). In the sequential version, the route extension is considered: given an existing route $(0,i,...,j,0)$ the algorithm determines the first saving S_{ki} or S_{jl} that can be used to feasibly merge this route with another route ending with $(k,0)$ or starting with $(0,l)$. If a feasible merge exists it is performed; otherwise, the next route is considered following the same procedure. In both versions, the algorithm runs until no further route combination is possible.

CW savings algorithm is eager to construct good quality routes at the early stages. The reason is that the savings values are high at the top of savings list due to smaller distances between customers relative to their distance to the depot. However, this characteristic of the formula restrains the exploration ability of the algorithm. Gaskell (1967) and Yellow (1979) highlighted this weakness and proposed a parameterized savings formula:

$$S_{ij} = c_{i0} + c_{0j} - \lambda c_{ij} \quad (2)$$

The motivation in using the parameter λ is to avoid circumferenced formation of routes that are usually produced by the original CW algorithm. In other words, this parameter helps to reshape the routes by taking only non-negative values in order to find better quality solutions.

Paessens (1988) introduced a new term to the Gaskell and Yellow's formula in an attempt to collect more information about the distribution. The proposed savings function is the following:

$$S_{ij} = c_{i0} + c_{0j} - \lambda c_{ij} + \mu \left| c_{i0} - c_{0j} \right| \quad (3)$$

where μ is a positive constant. The inclusion of the new term in (3) may exploit the asymmetry information between customers i and j regarding their distances to the depot. Nevertheless, this information adds an unfair savings to certain customer pairs in some cases such as a case in which one customer is very close to the depot and another is very distant. Altınel and Öncan (2005) propose an enhancement to Paessens' formula by introducing a third term which considers the demand of customer pairs and the overall average demand. Inspired from the well-known *first fit decrease* idea of Martello and Toth (1990), originally used for the bin packing problem (BPP), they adopt *put first larger item* approach which gives priority to the customers with large demands. The new formula is follows:

$$S_{ij} = c_{i0} + c_{0j} - \lambda c_{ij} + \mu \left| c_{i0} - c_{0j} \right| + \nu \frac{d_i + d_j}{\bar{d}} \quad (4)$$

Here d_i (d_j) is the demand of customer i (j), $\bar{d}$ is the average demand, and ν is the new non-negative parameter. Since this algorithm is based on three parameters it requires more computational effort.

Very recently, Battara et al. (2007) have applied a genetic algorithm to a small subset of representative CVRP problems to determine a robust set of parameters. In the next stage, they used this set of parameters as the starting points for a fast local search procedure. By doing so, they were able to more deeply investigate the space of parameter sets for each problem to be solved. They tested their method in Altınel and Öncan (2005)'s algorithm using the same instances and achieved approximately same solution quality in a very short computational time.

Apart from the savings algorithm related methodologies, a method proposed by Schneider (2001) attempts to find a set of reasonable good solutions after a number of independent trials. After these trials, algorithm finds the sub-tours identical in all of the routes obtained by these trials, and create route backbones to be used in the second phase of the algorithm. Without destroying the set of backbones, the algorithm constructs different configurations of routes. The performances of all these routes are tested, and new longer backbones are found to be used for configuring new set of routes. These procedures are repeated until the algorithm finds the same quality of solutions for all configurations of the routes.

A more recent approach is Algetect Electrostatic Algorithm proposed by Faulin and Valle (2007). Inspired by the Coulomb's Law, this algorithm imitates the customer demands to the behavior of two electrostatic charges. The primary objective of the algorithm is to minimize the number of routes. Thus, in most of the instances it finds longer vehicle routes compared to the CW algorithm, though with less number of vehicles.

CHAPTER 3

ENHANCEMENTS ON THE TWO-PARAMETER SAVINGS FUNCTION

3.1 Proposed Enhancements

In this chapter, we study the two-parameter CW savings function and proposed some enhancements on it. As mentioned in the previous section, the CW savings formulation of Paessens (1988) aims at collecting more information about the spatial distribution of the customers. The second term proposed in this formulation increases the savings value of the customer pairs if the difference of the displacements of the customers to the depot is high. The disadvantage of this approach is it disregards the distance between the customers and gives an unfair savings to some customer pairs. On the other hand, the contribution of the new term is controlled by a parameter and the negative effect is partially eliminated.

The proposed enhancement attempts to weaken the effect of this unfair contribution of the second term by multiplying it with the cosine value of the polar coordinate angles of the customers with the depot. This coefficient provides positive savings value to the customer pairs when this angle is acute. This positive contribution increases as the angle gets more acute, implying that the customers are closer in the polar coordinate. On the other hand, if the angle between the customers is greater than 90 degrees, our term has a negative contribution to the savings of this particular customer pair, since the cosine value of the angle is negative. Thus, as the angle gets more obtuse, the effect of this negative contribution increases due to decreasing negative cosine value. The approach is similar to that of the sweep algorithm. The sweep algorithm is firstly mentioned in a paper by Wren and Holliday (1972). This algorithm applies to planar instances of VRP. By rotating a ray centered at the depot, feasible clusters are initially formed. Then, final routes are obtained by solving a TSP for each cluster. The idea of the sweep algorithm is integrated to the Paessens' savings formula by proposing the following savings function:

$$S_{ij} = c_{i0} + c_{0j} - \lambda c_{ij} + \mu \cos(\theta_{ij}) \left| c_{i0} - c_{0j} \right| \qquad (5)$$

In the above function, θ_{ij} is the angle formed by the two rays originating from the depot and crossing the customers i and j. By utilizing the cosine theorem, $\cos(\theta_{ij})$ is simply calculated as $(c_{i0}^2 + c_{0j}^2 - c_{ij}^2)/2c_{i0}.c_{0j}$. This new savings expression (5) can be regarded as a more general enhancement to the Gaskell and Yellow's and Paessens' savings formulae.

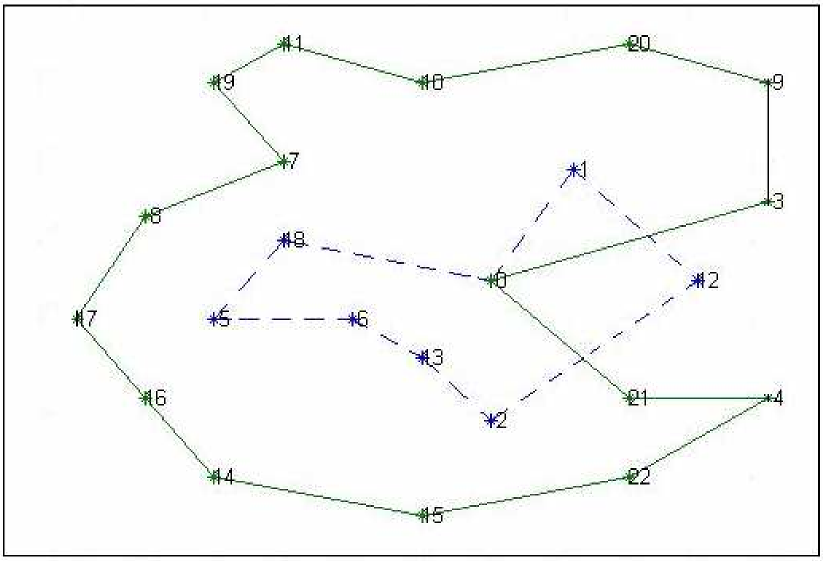

Figure 3.1. Classical Clark-Wright algorithm

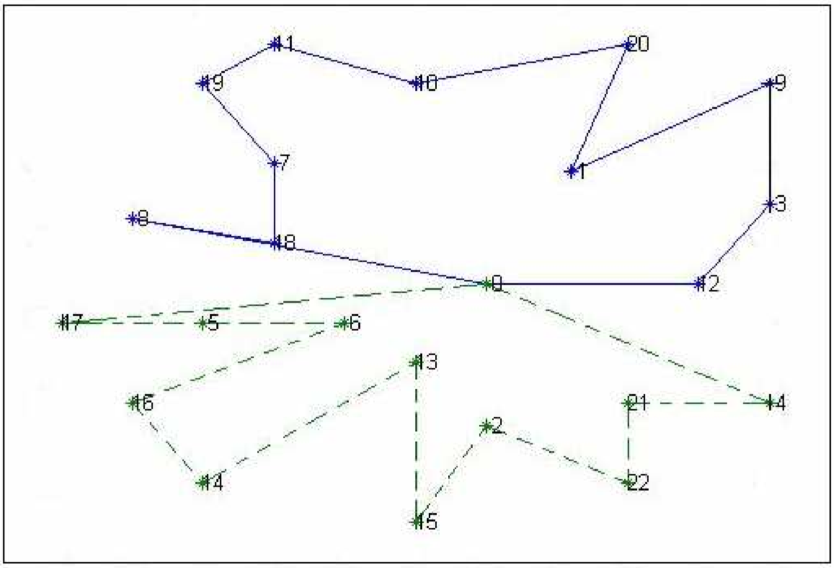

Figure 3.2. Sweep algorithm

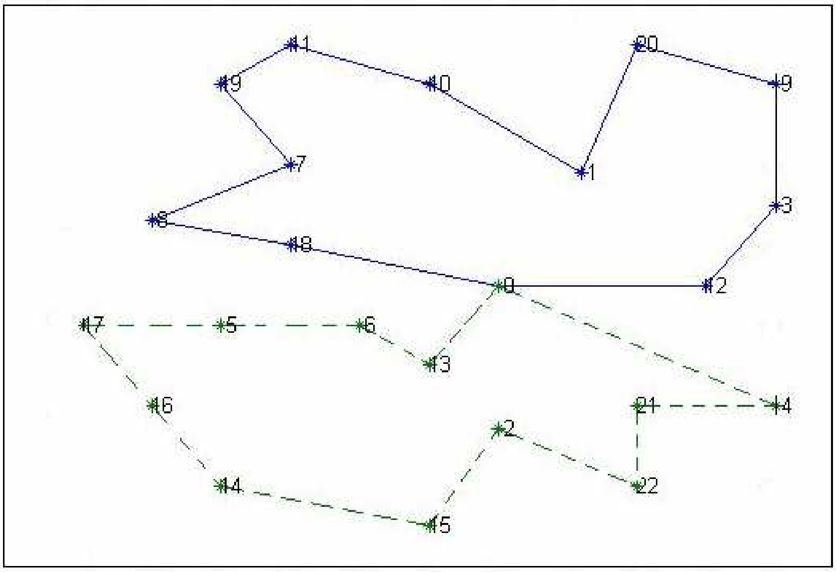

Figure 3.3. Proposed algorithm with $\lambda = \mu = 1$

To illustrate the effect of the proposed formula, an example consisting of 22 customers is used. Figure 3.1 shows the two routes obtained by using the classical CW algorithm. The depot is denoted as 0. Classical CW algorithm provides a total distance of 324.87. Figure 3.2 depicts the solution given by the sweep algorithm. It corresponds to a total distance of 358.45. The result of the proposed approach is illustrated in Figure 3.3. Total distance of 298.87 is obtained by setting $\lambda = \mu = 1$. Note that we selected these parameter values for simplicity and a better solution maybe obtained by tuning the parameters.

Figures 3.1-3.3 illustrates that the classical CW algorithm constructs more circumferenced routes since the savings are high at the top of the savings list due to smaller distances between customers relative to their distance to the depot. This deficiency of the classical CW limits the shape of the routes to be constructed and restricts the exploration ability of the algorithm leading relatively high cost. On the other hand, the sweep algorithm takes into account only the polar angles of the customers with the depot. This algorithm ignores the distance between the customers and the distances of the customers to the depot. Consequently, the total routing cost becomes highest due to the lack of information. Our formula, however, takes the advantage of the information used both in the sweep and CW heuristics, and may provide shorter distance for the sample. Figure 3.3 shows how the routes are reshaped and their circumferenced characteristics disappear by integrating the cosine value. Moreover, the greedy feature of the sweep algorithm is eliminated by preserving the idea of classical savings algorithm.

Next, we present a two-term savings function where the second term is completely new. In this function, the second term includes the absolute value of the difference between the maximum distance among all customer pairs and the average of the distances between customers i and j and the depot as well as the cosine of the angle associated with customers i and j. The adjusting parameter μ is preserved. The proposed new savings formula is as follows:

$$S_{ij} = c_{i0} + c_{0j} - \lambda c_{ij} + \mu \cos(\theta_{ij}) \left| c^{\max} - (c_{i0} + c_{0j})/2 \right| \quad (6)$$

where $c^{\max}$ represents the largest distance among all distances between the customers. Note that $c^{\max}$ is usually greater than $(c_{i0} + c_{0j})/2$ unless the customers are accumulated at one side of the depot, which is rarely the case in real world problems. In order to handle such exceptional cases, we use the absolute value of the term. Our motivation in using this approach is to give an early placement priority to the customers located near the depot. The last term in formula (6) will contribute more to the savings value of customer pair if the average distance of customer i and customer j to the depot is low.

3.2 Experimental Analysis

The performance of the proposed savings functions is tested against that of Paessen's formula on a large number of experimental problem instances. The computational effort in formula (5) is slightly higher due to the inclusion of "cos θ_{ij}". In formula (6), the additional computational effort arises from the calculation of the average distance of the customer pair to the depot and finding of $c^{\max}$ as well as the inclusion of the "cos θ_{ij}" term. The search effort, on the other hand, is the same since we use the same parameters. Our algorithm stores the savings in a vector of size (n*(n+1)/2) and uses vector of size n to keep track of the assignment of the customers to the vehicle routes.

The results are reported in Tables 3.1-3.6. Tables 3.1-3.5 consist of only capacity restricted instances whereas the instances in Table 3.6 include maximum route length restriction as well. For Tables 3.1-3.3, the instances are chosen from Augerat's (1995) test sets. Table 3.4 consists of Christofides and Eilon's (1969) data set, and Christofides et al.'s (1979) test set is included in Table 3.5. In Table 3.6, a unique data set of Christofides et al.'s with maximum route length constraint is reported. The data sets and best known results reported in Tables 3.1-3.4 are obtained from http://www.branchandcut.org. The best known results for Christofides et al.'s data set are obtained from Laporte et al. (2000).

Table 3.1 Relative deviations on Augerat et al.'s test set P

Instance	Best	P	λ,μ	%Dev	SAVE1	λ,μ	%Dev	SAVE2	λ,μ	%Dev
P-n16-k8	450	451.94	0.5,2.0	0.43	451.94	0.5,2.0	0.43	451.94	0.1,0.8	0.43
P-n19-k2	212	220.64	0.5,0.9	4.08	220.64	0.6,0.8	4.08	220.64	0.1,1.3	4.08
P-n20-k2	216	233.99	0.7,0.2	8.33	229.43	2.0,1.0	6.22	228.24	1.7,1.0	5.67
P-n21-k2	211	236.18	0.7,0.2	11.93	231.54	2.0,1.0	9.74	212.71	0.4,1.5	0.81
P-n22-k2	216	219.89	0.7,1.9	1.80	219.89	0.7,1.9	1.80	217.87	0.2,1.5	0.87
P-n22-k8	603	589.39	0.0,0.9	-2.26	589.39	0.0,0.9	2.26	588.79	0.3,0.7	-2.36
P-n23-k8	529	536.71	0.0,1.2	1.46	536.34	0.2,1.4	1.39	536.34	0.1,0.9	1.39
P-n40-k5	458	468.20	1.0,1.2	2.23	468.20	1.0,1.2	2.23	485.63	0.6,0.6	6.03
P-n45-k5	510	523.91	0.7,1.9	2.73	523.91	0.7,1.8	2.73	531.23	0.5,0.6	4.16
P-n50-k10	696	712.77	0.1,1.2	2.41	712.77	0.1,1.2	2.41	712.77	0.9,0.3	2.41
P-n50-k7	554	578.94	0.4,1.7	4.50	578.43	0.7,2.0	4.41	577.73	0.7,0.8	4.28
P-n50-k8	631	646.54	0.2,1.4	2.46	646.54	0.3,1.3	2.46	646.54	0.7,0.7	2.46
P-n51-k10	741	754.97	0.3,1.1	1.89	754.97	0.4,0.8	1.89	755.59	0.6,0.9	1.97
P-n55-k10	694	716.06	0.3,1.4	3.18	716.06	0.3,1.4	3.18	713.23	0.7,1.2	2.77
P-n55-k15	989	963.32	0.5,1.4	-2.60	963.32	0.5,1.4	2.60	971.55	0.3,1.1	-1.76
P-n55-k7	568	589.54	0.1,1.4	3.79	589.54	0.1,1.4	3.79	585.63	0.4,1.4	3.10
P-n55-k8	576	594.84	0.5,1.4	3.27	594.84	0.5,1.4	3.27	596.28	0.9,0.5	3.52
P-n60-k10	744	769.27	0.5,1.5	3.40	768.12	0.5,1.6	3.24	785.25	0.3,1.0	5.54
P-n60-k15	968	1006.94	0.0,0.8	4.02	1006.94	0.0,0.8	4.02	1000.63	0.5,1.2	3.37
P-n70-k10	834	853.94	0.4,0.6	2.39	853.05	1.0,0.9	2.28	855.10	0.4,0.3	2.53
P-n76-k4	593	643.14	0.8,1.7	8.46	643.14	0.8,1.6	8.46	634.92	0.9,0.9	7.07
P-n76-k5	627	655.03	0.7,2.0	4.47	655.03	0.5,1.7	4.47	665.18	1.3,0.4	6.09
P-n65-k10	792	829.17	1.0,0.3	4.69	831.78	0.0,1.2	5.02	831.78	1.1,0.1	5.02
P-n101-k4	681	722.83	0.2,1.2	6.14	716.8	0.6,1.2	5.26	714.52	1.6,0.4	4.92
Average				**3.47**			**3.25**			**3.10**

Table 3.2 Relative deviations on Augerat *et al.*'s test set A

Instance	Best	P	λ,μ	%Dev	SAVE1	λ,μ	%Dev	SAVE2	λ,μ	%Dev
A-n32-k5	784	828.70	0.8;0.6	5.70	827.96	0.1;0.4	5.61	833.39	0.4;1.3	6.30
A-n33-k5	661	679.72	1.4;0.8	2.83	679.72	1.2;0.8	2.83	676.10	0.3;1.0	2.28
A-n33-k6	742	747.32	1.6;0.5	0.72	747.32	1.5;0.4	0.72	746.98	0.2;1.8	0.67
A-n34-k5	778	793.05	0.7;0.1	1.93	793.05	0.7;0.1	1.93	793.05	0.6;0.2	1.93
A-n36-k5	799	806.78	0.9;0.0	0.97	806.78	0.9;0.0	0.97	806.78	0.6;0.4	0.97
A-n37-k5	669	695.08	0.7;0.9	3.90	695.08	0.6;0.9	3.90	699.15	0.7;0.9	4.51
A-n37-k6	949	976.01	1.0;0.1	2.85	976.01	1.0;0.1	2.85	976.60	1.0;0.0	2.91
A-n38-k5	730	755.94	1.4;0.3	3.55	755.94	1.3;0.3	3.55	755.94	0.6;0.8	3.55
A-n39-k5	822	851.25	1.6;0.1	3.56	851.25	1.6;0.1	3.56	848.25	0.6;1.2	3.19
A-n39-k6	831	849.55	0.8;0.2	2.23	849.55	0.6;1.0	2.23	857.69	0.1;0.9	3.21
A-n44-k7	937	968.84	2.0;0.9	3.40	973.20	1.3;0.2	3.86	971.85	0.7;0.7	3.72
A-n45-k6	944	957.05	1.1;0.1	1.38	957.05	1.1;0.1	1.38	972.72	0.9;0.2	3.04
A-n45-k7	1146	1169.00	1.9;0.9	2.01	1169.00	1.9;0.9	2.01	1179.81	0.2;1.7	2.95
A-n46-k7	914	933.66	1.1;0.1	2.15	933.66	1.1;0.1	2.15	934.41	0.9;0.3	2.23
A-n48-k7	1073	1104.23	1.7;0.7	2.91	1104.23	1.7;0.7	2.91	1104.41	0.9;0.3	2.93
A-n53-k7	1010	1045.98	1.5;0.6	3.56	1043.86	1.3;1.4	3.35	1050.89	0.2;1.8	4.05
A-n54-k7	1167	1188.64	1.7;0.9	1.85	1188.30	1.7;0.9	1.83	1197.92	1.0;0.0	2.65
A-n55-k9	1073	1099.55	1.3;0.2	2.47	1099.83	1.0;0.0	2.50	1099.84	1.0;0.0	2.50
A-n60-k9	1354	1389.59	1.6;1.0	2.63	1385.35	1.6;1.0	2.32	1394.56	0.8;1.0	3.00
A-n61-k9	1034	1051.37	1.1;0.0	1.68	1051.37	1.1;0.0	1.68	1051.37	0.9;0.3	1.68
A-n62-k8	1288	1351.11	1.2;0.2	4.90	1351.11	1.2;0.2	4.90	1345.65	0.8;0.4	4.48
A-n63-k10	1314	1349.58	2.0;1.2	2.71	1349.58	2.0;1.2	2.71	1348.18	1.0;0.1	2.60
A-n64-k9	1401	1442.44	1.1;0.5	2.96	1434.63	1.4;0.5	2.40	1451.92	0.7;0.7	3.64
A-n63-k9	1616	1648.92	1.6;0.6	2.04	1648.82	1.6;0.6	2.03	1663.06	0.2;0.6	2.91
A-n65-k9	1174	1224.71	1.0;0.2	4.32	1216.72	1.0;0.2	3.64	1209.85	0.7;0.4	3.05
A-n69-k9	1159	1185.08	1.3;0.0	2.25	1185.08	1.3;0.0	2.25	1183.11	1.1;0.2	2.08
A-n80-k10	1763	1818.64	1.8;0.7	3.16	1818.64	1.8;0.7	3.16	1815.07	0.6;1.0	2.95
Average				**2.76**			**2.71**			**2.96**

Table 3.3 Relative deviations on Augerat *et al*.'s test set B

Instance	Best	P	λ,μ	%Dev	SAVE1	λ,μ	%Dev	SAVE2	λ,μ	%Dev
B-n31-k5	672	679.43	0.0,0.9	1.11	679.02	1.0,1.5	1.05	677.34	0.8,0.1	0.80
B-n34-k5	788	789.84	0.0,1.2	0.23	789.84	0.0,1.2	0.23	789.84	1.0,0.3	0.23
B-n35-k5	955	978.32	0.2,0.8	2.44	978.32	0.2,0.8	2.44	975.38	0.5,1.3	2.13
B-n38-k6	805	824.00	0.4,1.4	2.36	824.00	0.4,1.4	2.36	825.55	0.8,0.6	2.55
B-n39-k5	549	554.99	0.3,1.4	1.09	554.51	0.5,1.6	1.00	554.44	1.0,0.2	0.99
B-n41-k6	829	867.42	0.5,0.6	4.63	861.67	0.5,0.6	3.94	861.02	0.3,0.3	3.86
B-n43-k6	742	754.04	0.6,1.4	1.62	754.04	0.6,1.4	1.62	757.95	0.7,0.3	2.15
B-n44-k7	909	932.32	0.8,1.8	2.57	932.32	0.8,1.8	2.57	937.73	1.0,0.0	3.16
B-n45-k5	751	757.16	0.0,1.0	0.82	757.16	0.0,1.0	0.82	756.60	0.6,0.6	0.75
B-n45-k6	678	713.24	0.6,0.9	5.20	716.15	0.3,0.6	5.63	719.05	0.7,0.4	6.06
B-n50-k7	741	747.92	0.0,1.1	0.93	747.92	0.0,1.1	0.93	747.92	0.9,0.3	0.93
B-n50-k8	1312	1339.44	0.7,1.6	2.09	1344.60	0.4,1.2	2.49	1339.89	0.9,0.2	2.13
B-n51-k7	1032	1050.00	0.0,1.5	1.74	1050.00	0.0,1.5	1.74	1049.57	1.3,0.4	1.70
B-n52-k7	747	763.96	0.2,1.1	2.27	763.96	0.4,1.3	2.27	764.89	0.8,0.3	2.40
B-n56-k7	707	723.76	0.1,0.7	2.37	723.76	0.1,0.7	2.37	722.61	0.3,0.2	2.21
B-n57-k7	1153	1148.97	0.8,1.8	-0.35	1148.97	0.8,1.8	-0.35	1154.26	0.2,1.6	0.11
B-n57-k9	1598	1619.71	0.0,0.9	1.36	1619.71	0.0,0.9	1.36	1619.71	0.8,0.2	1.36
B-n63-k10	1496	1562.59	0.0,0.9	4.45	1557.74	1.3,1.1	4.13	1552.36	0.9,0.1	3.77
B-n64-k9	861	919.37	0.8,1.7	6.78	919.49	0.4,0.5	6.79	910.39	0.2,0.1	5.74
B-n66-k9	1316	1372.09	0.4,1.4	4.26	1364.70	0.5,1.4	3.70	1371.54	0.6,0.2	4.22
B-n67-k10	1032	1090.18	0.2,0.8	5.64	1090.18	0.2,0.8	5.64	1096.19	0.9,0.4	6.22
B-n68-k9	1272	1317.77	0.0,1.0	3.60	1317.66	1.0,2.0	3.59	1316.22	0.7,0.5	3.48
B-n78-k10	1221	1263.05	0.1,1.0	3.44	1263.05	0.1,1.5	3.44	1264.36	0.8,0.2	3.55
Average				**2.64**			**2.60**			**2.63**

Table 3.4 Relative deviations on Christofides and Eilon's test set

Instance	**Best**	**P**	***λ,μ***	**%Dev**	**SAVE1**	***λ,μ***	**%Dev**	**SAVE2**	***λ,μ***	**%Dev**
E-n22-k4	375	375.28	0.6,1.5	0.08	375.28	0.9,1.8	0.08	378.66	1.3,0.5	0.98
E-n23-k3	569	573.01	0.5,1.7	0.71	573.01	0.5,1.8	0.71	573.01	0.3,1.4	0.71
E-n30-k4	503	506.67	0.3,1.3	0.73	506.67	0.3,1.3	0.73	507.51	0.6,1.1	0.90
E-n33-k4	835	843.09	0.1,0.9	0.97	842.83	0.1,0.9	0.94	843.09	0.8,0.2	0.97
E-n76-k14	1021	1052.3	0.1,1.1	3.07	1052.3	0.1,1.1	3.07	1051.67	1.0,0.2	3.00
E-n76-k8	735	783.12	0.3,1.2	6.55	779.42	0.5,1.0	6.04	775.73	0.7,0.3	5.54
E-n76-k7	682	718.88	0.8,1.7	5.41	718.88	0.8,1.7	5.41	717.49	0.4,1.0	5.20
E-n101-k14	1071	1133.99	0.5,0.7	5.88	1131.2	0.5,0.6	5.62	1132.55	1.4,0.3	5.75
Average				**2.92**			**2.82**			**2.88**

Table 3.5 Relative deviations on Christofides *et al.*'s test set

Instance	**Best**	**P**	***λ,μ***	**%Dev**	**SAVE1**	***λ,μ***	**%Dev**	**SAVE2**	***λ,μ***	**%Dev**
C50	524.60	566.10	0.9,0.8	7.91	566.10	0.4,0.8	7.91	537.29	1.0,1.4	2.42
C75	835.30	866.29	0.1,1.0	3.72	866.29	0.1,1.0	3.72	866.29	0.7,0.3	3.72
C100a	826.10	865.60	0.4,1.5	4.78	865.60	0.3,1.5	4.78	877.24	1.2,0.0	6.19
C150	1028.00	1101.81	0.7,2.0	7.14	1098.80	0.8,0.8	6.84	1097.52	1.0,0.1	6.72
C199	1291.00	1370.04	0.2,1.4	6.09	1370.04	0.2,1.4	6.09	1367.53	1.3,0.2	5.89
C120	1042.00	1066.40	0.3,1.3	2.33	1065.60	0.4,1.4	2.25	1068.14	1.0,0.0	2.50
C100b	819.60	826.00	0.4,1.2	0.79	826.00	0.4,1.2	0.79	829.88	0.6,0.5	1.26
Average				**4.68**			**4.62**			**4.10**

Table 3.6 Relative deviations on Christofides *et al*.'s distance restricted test set

Instance	**Best**	**P**	***λ,μ***	**%Dev**	**SAVE1**	***λ,μ***	**%Dev**	**SAVE2**	***λ,μ***	**%Dev**
CD50	555.40	595.31	0.3,1.3	7.18	595.31	0.3,1.3	7.18	593.26	0.8,1.0	6.81
CD75	909.60	942.98	0.7,2.0	3.67	942.98	0.7,1.9	3.67	953.78	0.7,0.9	4.85
CD100a	865.90	942.69	0.3,1.7	8.86	942.69	0.3,1.7	8.86	922.76	0.7,0.1	6.56
CD150	1163.00	1222.10	0.0,1.3	5.12	1222.10	0.0,1.3	5.12	1217.62	1.2,0.2	4.74
CD199	1396.00	1485.50	0.6,1.9	6.42	1485.50	0.5,1.8	6.42	1478.12	1.2,0.4	5.89
CD120	1541.00	1583.24	0.1,0.7	2.73	1577.87	0.4,0.7	2.38	1586.90	1.1,0.1	2.97
CD100b	866.40	869.61	0.2,1.2	0.37	869.61	0.6,2.0	0.37	873.75	0.9,0.2	0.85
Average				**4.91**			**4.86**			**4.67**

In the first columns of tables 3.1-3.4, A, B, P and E represents the name of the data set, the numbers shows the total nodes including all customers and the depot. For Table 3.5-3.6, C and CD represent the name of the data sets and the number of customers are represented next to the name of datas.In tables 3.1-3.6, the instances are represented in the first column and best-known results are reported in the second column. The third, fourth, and fifth columns show the results obtained from Paessen's savings algorithm and represents total distance, parameters λ and μ, and relative deviations, respectively. Similarly, the total distance, parameters λ and μ and the relative deviations from best-known results are represented in the sixth, seventh, and eighth columns, respectively for our SAVE1 heuristic expressed with formula (5). And for our SAVE 2 heuristic expressed with formula (6), the total distance, parameters λ and μ and the relative deviations from best-known results are represented in ninth, tenth, and eleventh columns. In all instances, distances and customer demands are integer numbers. The number of customers in the problems varies between 15 and 199. All the instances are symmetric. The savings values are calculated by increasing λ and μ with an increment of 0.1 in the interval [0.1, 2] and [0, 2], respectively. The distances reported for Paessen's, SAVE1 and SAVE2 algorithms are the best values obtained among the 20 x 21 = 420 possible parameter combinations. Finally, we note that the algorithms are coded in C++.

In all benchmark instances, SAVE1 outperforms the average solution quality of Paessens (1988). SAVE2 also outperforms the savings function of Paessens (1988) in all problems except Augerat *et al.*'s test set A. However, SAVE2 performs significantly better than SAVE1 especially in Christofides et al.'s both test sets (0.53% and 0.21% additional improvement as compared with SAVE1 for unrestricted and restricted route length instances) and Augerat *et al.*'s test set P (0.31%). For three instances, we have negative values in relative deviations, implying a better distance compared to the reported best solutions. These results are obtained at the cost of an additional vehicle and do not represent any better solutions. Although significant improvements are achieved in certain instances the reductions in the average deviations are not dramatic. Nevertheless, the proposed savings functions are capable of providing shorter distances with almost no additional computational effort.

CHAPTER 4

ENHANCEMENTS ON THE THREE-PARAMETER SAVINGS FUNCTION

4.1 On the Enhancement Proposed by Altınel and Öncan (2005)

In this chapter, we first investigate the improvements achieved by the savings function of Altınel and Öncan (2005), and then propose some enhancements on the three-parameter CW savings function. We claim that Altınel and Öncan's approach benefits mostly from the extended ability of searching the solution space by using a third term rather than the *put first larger items* idea. To justify this claim two modified savings functions based on the conflicting idea of *putting first smaller item* are proposed. The first savings function is the same as (4); however, parameter v is used as a "non-positive" parameter. This penalty-based formulation promotes the customers with smaller demands by penalizing the customer pairs with larger demand more than the customer pairs with smaller demands. The second savings function uses the inverse of the third term giving priority to the customers with small demands and expressed as follows:

$$S_{ij} = c_{i0} + c_{0j} - \lambda c_{ij} + \mu \left| c_{i0} - c_{0j} \right| + v \frac{\bar{d}}{d_i + d_j} \quad (7)$$

In the literature, Gaskell (1967) and Yellow (1979) parameterize the distance between the customer pairs in the original CW formula and expand the search space of the algorithm. The original CW algorithm calculates the saving values of customer pairs once and constructs routes according to a single savings list. However, in formula (2) the shaping parameter λ is adjusted in the interval [0.1, 2] with an increment of 0.1. This implies that a best route is obtained using 20 different saving lists at the end of 20 iterations. Consequently, the computational effort increases in parallel with the number of iterations. Similarly, in Paessens' formula (3), the parameter μ changes within the interval [0, 2] with an increment of 0.1. Thus, this approach requires a total of 20 x 21 = 420 iterations. Formulation (4) proposed by Altınel and Öncan (2005) includes a third parameter v which changes with an increment of 0.1 in [0, 2], resulting in 20 x 21 x 21= 8820 iterations. When compared to Paessens' formula, it involves 8400 additional iterations implying a deeper search of the solution space achieved by different combinations of three parameters. To make a fair comparison, experiments are conducted the same way as Altınel and Öncan (2005) did. The parallel version is adopted and the same parameter values are used. Since the search effort is the same, the computational time is not reported.

The detailed results are given in the Appendix where the instances are represented in the first column and best-known results are included in the second column. The results obtained using Altınel and Öncan's savings function are denoted as AÖ and reported with the corresponding parameter values. NEG shows the results obtained using the same function with parameter *v* changing in [-2, 0] and INV shows the results obtained using formula (7) with the same parameter values as in AÖ. These results are reported with the corresponding parameter values as well. The % Dev column gives the deviations of distances in NEG and INV from AÖ and is calculated as (NEG/AÖ – 1) and (INV/AÖ – 1), respectively. Hence, a negative deviation value represents a shorter distance than that of AÖ.

Table 4.1 Performance comparison in terms of the number of problems

Test set	# of Prob	NEG vs. AÖ						INV vs. AÖ					
		# "<"	Perc (%)	# "="	Perc (%)	# ">"	Perc (%)	# "<"	Perc (%)	# "="	Perc (%)	# ">"	Perc (%)
Aug (*P*)	24	8	33.3	9	37.5	7	29.2	11	45.8	9	37.5	4	16.7
Aug (*A*)	27	9	33.3	4	14.8	14	51.9	11	40.7	5	18.5	11	40.7
Aug (*B*)	23	11	47.8	3	13.0	9	39.1	11	47.8	3	13.0	9	39.1
ChrEil	8	2	25.0	4	50.0	2	25.0	4	50.0	2	25.0	2	25.0
Chr (*C*)	7	2	28.6	0	0.0	5	71.4	2	28.6	1	14.3	4	57.1
Chr (*CD*)	7	4	57.1	1	14.3	2	28.6	4	57.1	2	28.6	1	14.3
Total	96	36	**37.5**	21	**21.9**	39	**40.6**	43	**44.8**	22	**22.9**	31	**32.3**

We observe that the average distances provided by both NEG and INV are shorter than those of AÖ on the benchmark instances of Augerat *et al*'s test set *B* and Christofides *et al*'s route length restricted test set. Furthermore, in Augerat *et al*'s test set *P* and Christofides and Eilon's test set INV outperforms AÖ as well. Only in two test sets, the average performance of AÖ is better than the performance of both NEG and INV. In addition, we observe that the performance of NEG and INV is significantly better in the existence of route length restriction with 0.689 % and 0.496% improvement, respectively. The results show that there is not significant difference in employing either approach. It is important to note here that although the majority of our results match those of Altınel and Öncan (2005) in the implementation of their savings function there are certain instances for which we find shorter or longer distances with different parameter values. This may be due to the implementation of the algorithm on the computer code. Variability in the numerical results reported for different savings heuristics is also pointed out in Laporte *et al* (2000). For consistency, we compare our distances and the corresponding parameter values with those we obtained in our code using Altınel and Öncan's (2005) formula.

To compare the number of instances in which NEG and INV find shorter distances than ("<"), same distance as ("="), or longer distance than (">") AÖ, we provide Table 4.1. In this table, Aug, ChrEil, and Chr, respectively, denote the test sets of Augerat *et al,* Christofides and Eilon, and Christofides *et al,* respectively. The results show that NEG gives the best distance in 38% of the problems whereas AÖ performs better in 41%. Moreover, we observe that in 45% of the problems INV provides the best distance while AÖ finds the best distance only in 32%. The performance of INV is significantly better than AÖ for Augerat *et al*'s test set *P* (46% vs. 17%), the test set of Christofides and Eilon (50% vs. 25%), and Christofides *et al*'s route length restricted test set (57% vs. 14%). AÖ outperforms INV in only Christofides *et al*'s test set with no route length restriction. These results confirm that the idea of *put first smaller items* works as good as the *put first larger items* idea and even better in many instances, particularly in the case of INV formulation. In sum, we believe that the good performance of both approaches is more likely due to the ability of better searching the solution space using the newly introduced third parameter rather than the *put first larger items* idea as claimed in Altınel and Öncan (2005).

In addition, another drawback of (7) is that the first two terms consist of a distance measure whereas the third term is the ratio of demands and is unitless. Thus, if the distance measure changes, the relative weight of the third term will also change. That is, for instance, if the distances are switched from kilometers to meters the same value of v will not work as good. Hence, it will need to be readjusted in a new search interval, requiring additional computational effort.

To overcome these shortcomings of Altınel and Öncan's algorithm and improve its performance, we present in the next section two new enhanced savings formulations which offer a more robust solution quality.

4.2 Proposed Enhancements

Our first enhancement attempts to improve Paessens's two-term savings function by introducing a third term that aims at eliminating vehicle capacity losses due to inefficient routing. Our third term is also demand-based as is the case in Altınel and Öncan (2005); however, the underlying idea is quite different. The proposed formula is as follows:

$$S_{ij} = \frac{c_{i0} + c_{0j} - \lambda c_{ij}}{c^{\max}} + \mu \frac{\left| c_{i0} - c_{0j} \right|}{c^{\max}} + v \frac{\left| \overline{d} - (d_i + d_j)/2 \right|}{d^{\max}} \qquad (8)$$

where d^{max} denotes the maximum demand among all customers.

First, we normalize the savings function by dividing the distances by the maximum distance and the demands by the maximum demand. Hence, all distances and demands are represented within a unit measure. Therefore, the parameter intervals do not need to be readjusted for different measures. Second, v is allowed to take both positive and negative values. As far as the positive values are concerned, the savings value increases as the average demand of a customer pair diverges from the overall average. In other words, two customers both having low or high demands are rewarded the most and ranked closer in the savings list. In CVRP, one of the most challenging aspects of using savings algorithms is the losses in capacity utilization. Especially, if a vehicle visits customers with larger demands at the beginning of the tour, its remaining capacity cannot be usually utilized by nearby customers having lower demands. Formulation (8) aims at increasing the possibility of customers having low and high demands to be fit into the same route together and thus, minimizing the capacity losses. On the other hand, if v takes negative values, customer pairs having an average demand close to the overall average will be penalized the least and ones with large or small demands will be penalized most. In this case, the former customer pairs move towards the top of the savings list, while the latter ones go downwards. However, the idea of keeping customer pairs having lower and higher demands close in the savings list is preserved.

As discussed in the previous chapter, the idea of giving early placement priority to the customers located near the depot and utilizing the polar angles between the customer pairs in the savings functions provide shorter distances. Thus, in the second enhancement, we incorporate the same approach in the second term and propose the following savings function:

$$S_{ij} = \frac{c_{i0} + c_{0j} - \lambda c_{ij}}{c^{\max}} + \mu \frac{\cos(\theta_{ij}) \left| c^{\max} - (c_{i0} + c_{0j})/2 \right|}{c^{\max}} + v \frac{\left| \overline{d} - (d_i + d_j)/2 \right|}{d^{\max}} \qquad (9)$$

The distances and the demands are also normalized in (9) to have a robust formula independent from the measurement units.

4.3 Experimental Analysis

For fair comparison, we conduct our experiments on the same data set the same way as Altınel and Öncan (2005). We adopt the parallel version and use the same number of iterations. Since the search effort is the same we do not report the computation times. The algorithm is coded in C++. The results are given in Tables 4.2-4.7. In all the tables, the instances are reported in the first column and best-known results and the results given by classical CW method are included in the second and third columns, respectively. The results obtained using Altınel and Öncan's savings function are denoted as AÖ and reported with the corresponding parameter values (λ, μ, ν). DÇ1 and DÇ2 columns show the results obtained using savings functions (8) and (9), respectively, reported with the corresponding parameter values as well. Our experiments showed that the parameter ν changing within the interval [-0.1, 0.1] with an increment of 0.01 works well. The % Imp column gives the improvements in the distances obtained by AÖ, DÇ1, and DÇ2, respectively, in comparison with CW and is calculated as (CW-AÖ)/CW, (CW-DÇ1)/CW, and (CW-DÇ2)/CW, respectively.

We observe that both DÇ1 and DÇ2 provide shorter average distances than AÖ for all benchmark instances. To compare the number of instances in which DÇ1 and DÇ2 perform "better than AÖ" and "better than or same as AÖ" we provide Table 4.8. The results show that DÇ1 outperforms AÖ in 58% of the problems whereas DÇ2 outperforms AÖ in 55% of the problems. Moreover, DÇ1 and DÇ2 provide "better or equal quality" solutions in 74% and 68% of the problems, respectively. We also investigated the effect of increasing the computational effort twice by extending the interval of parameter ν to [-0.2, 0.2]. However, this extended interval has only a negligible contribution of 0.14% to the average distance of all benchmark instances. We should also point out that we tested formulas in which the demand terms are the inverse of the last term used in formula (8) and (9). The inverse of the newly integrated demand idea provide no improvement.

These results indicate that the newly integrated demand idea works better in many instances and proposed savings functions are capable of providing shorter distances with almost no additional computational effort. It is worth noting that the parameter tuning procedure proposed by Battara *et al.* (2007) is also applicable to our new three-parameter versions of CW algorithm to reduce the intensive computational requirement of parameter adjusting.

Table 4.2 Relative deviations on Augerat *et al.*'s test set P

Instance	**Best**	**CW**	**AÖ**	***λ,μ,ν***	**% Imp**	**DÇ1**	***λ,μ,ν***	**% Imp**	**DÇ2**	***λ,μ,ν***	**% Imp**
P-n16-k8	450	478.77	451.94	1.8,0.7,1.5	5.604	451.94	1.7,0.4,-0.10	5.604	451.95	0.1,1.6, 0.04	5.603
P-n19-k2	212	237.89	220.64	0.9,0.5,0.0	7.252	220.64	0.6,0.7,-0.03	7.252	220.64	0.1,1.3,-0.10	7.252
P-n20-k2	216	234.00	232.86	1.2,1.0,1.7	0.485	224.01	0.4,0.9, 0.09	4.267	224.13	0.1,1.4, 0.09	4.215
P-n21-k2	211	236.19	231.54	1.4,1.0,2.0	1.967	223.90	1.9,1.1,-0.10	5.201	212.71	0.8,1.4, 0.01	9.939
P-n22-k2	216	239.50	219.89	1.8,0.2,0.8	8.188	219.89	1.4,0.6,-0.10	8.188	217.87	0.2,1.5,-0.04	9.031
P-n22-k8	603	590.62	589.39	0.9,0.0,0.0	0.208	589.10	0.8,0.0, 0.06	0.257	588.79	0.1,0.9, 0.03	0.309
P-n23-k8	529	539.48	536.71	1.4,0.2,1.3	0.513	536.35	1.2,0.0, 0.10	0.580	536.35	0.1,0.8, 0.05	0.580
P-n40-k5	458	518.37	468.20	1.1,1.0,0.3	9.679	468.20	1.1,0.8,-0.07	9.678	470.20	0.7,1.4, 0.07	9.293
P-n45-k5	510	572.95	522.41	1.5,0.1,0.7	8.821	522.76	1.4,0.3, 0.10	8.760	521.31	1.2,1.3,-0.10	9.014
P-n50-k10	696	739.84	712.77	1.2,0.1,0.0	3.659	711.22	1.0,0.2, 0.07	3.868	712.77	0.8,0.4,-0.04	3.659
P-n50-k7	554	597.03	577.73	1.7,0.4,1.9	3.233	576.73	1.4,0.5, 0.10	3.401	577.73	0.7,0.8,-0.01	3.232
P-n50-k8	631	674.34	646.55	1.2,0.2,0.8	4.121	646.55	1.2,0.1,-0.10	4.121	646.55	0.6,0.7,-0.04	4.121
P-n51-k10	741	790.97	754.98	0.9,0.4,0.1	4.550	748.62	1.4,0.3, 0.10	5.354	747.25	0.7,0.6, 0.09	5.528
P-n55-k10	694	736.45	715.21	1.2,0.1,1.7	2.884	713.56	1.3,0.1, 0.10	3.107	709.33	1.8,0.8, 0.05	3.683
P-n55-k15	989	978.07	963.32	1.6,0.9,0.0	1.508	963.33	1.6,0.8,-0.10	1.507	959.93	0.2,1.2, 0.08	1.854
P-n55-k7	568	618.68	587.44	1.4,0.4,1.3	5.049	584.23	1.5,0.0, 0.07	5.569	584.23	1.4,0.1, 0.10	5.569
P-n55-k8	576	631.67	588.04	1.3,0.3,1.7	6.907	592.03	1.4,0.5,-0.08	6.276	594.30	1.3,0.1,-0.10	5.917
P-n60-k10	744	800.20	768.12	1.7,0.5,0.1	4.008	764.38	1.9,0.6, 0.08	4.476	765.08	0.6,0.8, 0.09	4.388
P-n60-k15	968	1016.96	1002.77	0.9,0.0,0.5	1.395	1002.78	0.8,0.0, 0.01	1.394	996.87	0.5,1.2,-0.10	1.975
P-n70-k10	834	896.86	853.94	0.6,0.4,0.0	4.786	853.94	0.7,0.3, 0.02	4.786	855.10	0.3,0.4, 0.01	4.656
P-n76-k4	593	688.34	641.78	1.9,0.8,0.4	6.765	620.81	1.8,1.3, 0.07	9.810	616.30	1.0,0.8,-0.05	10.466
P-n76-k5	627	709.38	652.93	1.6,0.3,0.9	7.957	651.42	1.8,0.8, 0.02	8.171	647.31	0.6,0.9,-0.09	8.749
P-n65-k10	792	844.61	825.92	1.9,0.7,0.7	2.213	816.17	1.9,1.0,-0.07	3.367	815.96	0.3,1.3,-0.04	3.392
P-n101-k4	681	765.38	711.03	0.6,1.0,0.0	7.101	707.25	1.6,1.0,-0.10	7.595	702.04	1.7,0.3,-0.10	8.275
Average					**4.536**			**5.108**			**5.446**

Table 4.3 Relative deviations on Augerat *et al.*'s test set A

Instance	Best	CW	AÖ	λ,μ,ν	% Imp	DÇ1	λ,μ,ν	% Imp	DÇ2	λ,μ,ν	% Imp
A-n32-k5	784	843.69	828.70	0.8,0.6,0.0	1.777	828.70	0.7,0.4,0.05	1.777	828.70	0.3,0.5,0.03	1.777
A-n33-k5	661	712.05	676.10	2.0,1.0,1.6	5.049	679.73	1.2,0.8,0.03	4.539	676.10	0.3,0.9,-0.01	5.049
A-n33-k6	742	776.26	743.21	1.2,0.0,1.0	4.258	747.22	1.9,1.1,0.09	3.741	746.99	0.1,1.8,-0.08	3.771
A-n34-k5	778	810.41	793.05	0.6,0.3,1.1	2.142	793.05	0.6,0.4,0.06	2.142	793.05	0.6,0.2,-0.06	2.142
A-n36-k5	799	828.47	806.78	0.8,0.0,0.1	2.618	806.78	0.8,0.0,0.01	2.618	806.78	0.6,0.4,-0.02	2.618
A-n37-k5	669	707.81	694.43	1.5,0.3,0.9	1.890	694.44	1.3,0.1,-0.1	1.889	694.44	1.0,0.3,-0.09	1.889
A-n37-k6	949	976.61	974.56	1.0,0.0,0.4	0.210	972.51	1.0,0.1,-0.1	0.420	976.61	0.8,0.3,-0.04	0.000
A-n38-k5	730	768.13	756.11	1.4,0.3,0.0	1.565	755.94	1.5,0.5,0.01	1.587	755.94	0.6,0.8,0.00	1.587
A-n39-k5	822	901.99	848.24	1.2,0.2,0.3	5.959	848.18	1.6,0.4,0.1	5.966	843.23	0.3,1.5,0.09	6.514
A-n39-k6	831	863.08	849.56	0.8,0.2,0.0	1.566	849.56	0.8,0.2,-0.01	1.566	849.90	0.5,0.4,-0.09	1.527
A-n44-k7	937	976.04	959.43	1.6,0.4,2.0	1.702	957.03	1.5,0.3,-0.1	1.948	957.03	0.7,0.8,-0.04	1.948
A-n45-k6	944	1006.45	957.06	1.0,0.0,1.4	4.907	957.06	1.1,0.1,0.01	4.907	957.06	1.0,0.1,0.01	4.907
A-n45-k7	1146	1199.98	1166.39	1.5,0.2,2.0	2.799	1167.66	1.6,0.4,0.07	2.693	1168.97	1.1,0.6,0.05	2.584
A-n46-k7	914	939.74	933.66	1.1,0.1,0.0	0.647	929.42	1.1,0.3,0.1	1.098	929.42	0.8,0.1,0.08	1.098
A-n48-k7	1073	1112.82	1104.24	1.7,0.7,0.0	0.771	1103.99	1.0,0.0,-0.04	0.793	1103.99	0.7,0.5,-0.04	0.793
A-n53-k7	1010	1099.45	1045.47	0.7,0.0,1.5	4.910	1043.86	1.8,1.6,0.07	5.056	1048.79	0.8,0.2,-0.02	4.608
A-n54-k7	1167	1197.92	1173.77	1.1,0.1,0.9	2.016	1172.27	1.1,0.1,-0.03	2.141	1172.27	0.8,0.4,-0.02	2.141
A-n55-k9	1073	1099.84	1098.51	0.9,0.1,1.1	0.121	1098.91	1.2,0.2,0.05	0.085	1099.56	1.1,0.0,0.06	0.025
A-n60-k9	1354	1421.88	1376.20	1.4,0.0,0.9	3.213	1384.19	1.8,1.0,0.09	2.651	1379.86	0.9,0.8,-0.10	2.955
A-n61-k9	1034	1102.23	1051.10	1.1,0.0,0.1	4.639	1051.06	1.1,0.0,0.05	4.642	1051.06	0.9,0.3,0.07	4.642
A-n62-k8	1288	1352.81	1347.87	1.0,0.0,0.2	0.365	1326.52	1.2,0.2,-0.07	1.943	1326.54	0.7,0.4,-0.01	1.942
A-n63-k10	1314	1352.48	1348.17	1.5,0.4,0.2	0.319	1346.84	1.3,0.3,0.05	0.417	1347.30	1.0,0.1,-0.04	0.383
A-n64-k9	1401	1486.92	1439.75	1.9,0.9,0.1	3.172	1435.25	1.4,0.5,-0.04	3.475	1442.66	1.0,0.1,0.07	2.977
A-n63-k9	1616	1687.96	1649.14	1.6,0.6,0.1	2.300	1648.44	1.5,0.7,-0.04	2.341	1652.42	0.3,1.5,0.04	2.106
A-n65-k9	1174	1239.42	1202.08	0.9,0.1,0.3	3.013	1205.84	1.4,0.4,-0.08	2.709	1197.49	0.7,0.4,0.02	3.383
A-n69-k9	1159	1210.78	1185.08	1.3,0.0,0.0	2.123	1183.88	1.3,0.0,-0.05	2.222	1181.91	1.1,0.2,-0.05	2.384
A-n80-k10	1763	1860.94	1816.78	1.8,1.4,1.5	2.373	1811.43	1.4,0.6,-0.08	2.660	1811.56	0.6,0.9,-0.03	2.653
Average					**2.460**			**2.520**			**2.534**

Table 4.4 Relative deviations on Augerat *et al.*'s test set B

Instance	**Best**	**CW**	**AÖ**	***λ,μ,ν***	**% Imp**	**DÇ1**	***λ,μ,ν***	**% Imp**	**DÇ2**	***λ,μ,ν***	**% Imp**
B-n31-k5	672	681.16	677.34	0.9,0.0,0.1	0.561	676.50	0.9,0.0,-0.1	0.684	676.50	0.3,1.0,-0.03	0.684
B-n34-k5	788	794.33	789.85	1.2,0.0,0.0	0.564	789.85	1.2,0.0,-0.01	0.564	789.85	1.0,0.3,0.00	0.564
B-n35-k5	955	978.33	975.48	1.1,0.1,1.7	0.291	975.34	1.0,0.0,-0.1	0.306	973.27	0.7,0.9,-0.05	0.517
B-n38-k6	805	832.09	824.00	1.4,0.4,0.0	0.972	824.60	1.5,0.4,-0.01	0.900	820.31	0.5,1.0,0.03	1.416
B-n39-k5	549	566.71	555.00	1.4,0.3,0.0	2.066	554.16	1.5,0.4,0.0	2.215	554.35	1.1,0.0,-0.04	2.181
B-n41-k6	829	898.09	867.42	0.6,0.4,0.1	3.415	861.67	0.8,0.7,0.01	4.055	852.95	0.3,0.3,-0.06	5.026
B-n43-k6	742	781.96	754.92	0.9,0.1,0.4	3.458	754.04	1.1,0.2,-0.01	3.571	756.07	0.7,0.3,0.03	3.311
B-n44-k7	909	937.74	934.68	1.9,0.9,1.8	0.326	932.25	1.8,0.8,0.1	0.585	930.99	0.6,0.8,-0.03	0.720
B-n45-k5	751	757.16	754.71	1.1,0.0,0.8	0.324	756.52	1.6,0.6,0.03	0.085	756.60	0.5,0.7,-0.01	0.074
B-n45-k6	678	727.84	713.24	0.9,0.6,0.0	2.006	708.52	0.9,0.6,0.08	2.654	717.24	0.2,0.5,0.10	1.456
B-n50-k7	741	748.80	745.37	1.0,0.0,0.2	0.458	744.77	1.1,0.0,-0.05	0.538	744.77	0.9,0.3,-0.03	0.538
B-n50-k8	1312	1354.03	1338.34	1.9,0.9,0.8	1.159	1336.72	1.8,0.8,-0.01	1.278	1337.13	0.9,0.2,-0.05	1.248
B-n51-k7	1032	1059.86	1050.00	1.5,0.0,0.0	0.930	1043.58	1.2,0.0,0.04	1.536	1043.58	1.2,0.0,0.04	1.536
B-n52-k7	747	764.90	756.90	1.3,0.0,1.5	1.046	762.16	1.3,0.0,-0.06	0.358	762.16	1.0,0.5,-0.06	0.358
B-n56-k7	707	733.74	722.61	0.8,0.0,0.2	1.517	723.77	0.7,0.0,-0.01	1.359	722.62	0.2,0.2,0.01	1.516
B-n57-k7	1153	1239.78	1148.98	1.1,0.0,0.5	7.324	1148.48	1.7,0.8,-0.04	7.364	1150.77	1.1,0.0,0.05	7.179
B-n57-k9	1598	1653.42	1619.72	0.9,0.0,0.0	2.038	1613.27	0.9,0.0,0.02	2.428	1613.27	0.8,0.2,0.01	2.428
B-n63-k10	1496	1598.18	1562.59	0.9,0.0,0.0	2.227	1559.81	0.8,0.0,0.01	2.401	1552.36	0.9,0.1,0.00	2.867
B-n64-k9	861	921.56	910.07	1.1,0.8,2.0	1.247	897.51	1.5,0.8,0.04	2.610	907.30	0.5,0.7,0.05	1.547
B-n66-k9	1316	1416.42	1358.32	1.9,1.1,1.0	4.102	1340.00	1.6,0.9,0.09	5.395	1357.17	0.3,0.8,0.06	4.183
B-n67-k10	1032	1099.95	1070.30	0.8,0.0,1.8	2.696	1071.72	0.8,0.0,-0.1	2.566	1066.79	0.7,0.2,-0.09	3.015
B-n68-k9	1272	1317.77	1316.07	1.1,0.1,0.4	0.129	1315.76	1.0,0.0,-0.03	0.153	1315.76	0.9,0.2,-0.03	0.153
B-n78-k10	1221	1264.56	1261.35	1.0,0.1,0.9	0.254	1260.99	1.0,0.0,0.06	0.282	1260.50	1.0,0.1,-0.05	0.321
Average					**1.700**			**1.908**			**1.863**

Table 4.5 Relative deviations on Christofides and Eilon's test set

Instance	Best	CW	AÖ	λ,μ,ν	% Imp	DÇ1	λ,μ,ν	% Imp	DÇ2	λ,μ,ν	% Imp
E-n22-k4	375	388.77	375.28	1.1,0.9,1.1	3.470	375.28	1.3,0.6,0.05	3.470	375.28	0.1,1.1,0.02	3.470
E-n23-k3	569	621.09	573.01	1.7,0.5,0.0	7.741	573.01	1.7,0.5,-0.10	7.741	573.01	0.2,1.4,0.01	7.741
E-n30-k4	503	534.45	506.67	1.3,0.3,0.0	5.198	506.67	1.3,0.3,-0.01	5.198	507.51	0.6,1.0,-0.02	5.041
E-n33-k4	835	843.10	843.10	0.9,0.1,0.0	0.000	842.83	0.8,0.1,0.06	0.032	842.83	0.7,0.4,0.08	0.032
E-n76-k14	1021	1054.60	1045.04	1.3,0.0,1.0	0.907	1052.20	1.2,0.0,-0.02	0.228	1049.31	0.7,0.3,0.08	0.502
E-n76-k8	735	794.74	779.42	1.0,0.5,0.1	1.928	768.82	1.3,0.1,0.06	3.262	768.05	1.2,0.1,0.05	3.358
E-n76-k7	682	738.13	718.88	1.6,0.8,0.2	2.608	712.14	1.9,0.8,0.07	3.521	716.48	0.3,1.0,-0.04	2.933
E-n101-k14	1071	1139.07	1126.39	0.8,0.6,0.3	1.113	1120.59	0.7,0.6,0.00	1.622	1127.01	1.3,0.3,-0.07	1.059
Average					**2.871**			**3.134**			**3.017**

Table 4.6 Relative deviations on Christofides *et al.*'s test set

Instance	Best	CW	AÖ	λ,μ,ν	% Imp	DÇ1	λ,μ,ν	% Imp	DÇ2	λ,μ,ν	% Imp
C50	524.61	584.64	555.55	1.7,0.2,0.6	4.976	550.24	0.3,0.9,0.08	5.884	537.29	1.0,1.4,-0.02	8.099
C75	835.26	907.39	860.21	1.2,0.2,0.7	5.200	862.81	1.1,0.3,-0.06	4.913	864.29	0.5,0.4,-0.06	4.750
C100a	826.14	889.00	867.35	1.2,0.6,0.1	2.435	862.18	1.8,0.5,-0.05	3.017	854.49	1.6,0.5,-0.05	3.882
C150	1028.42	1140.42	1094.06	1.3,0.1,0.3	4.065	1090.24	1.0,0.7,0.06	4.400	1089.78	0.4,0.6,-0.03	4.440
C199	1291.45	1395.74	1359.78	1.3,0.0,1.1	2.576	1363.93	1.6,0.1,-0.07	2.279	1367.53	1.3,0.2,0.00	2.021
C120	1042.11	1068.14	1057.80	1.1,0.1,0.3	0.968	1060.87	1.1,0.1,-0.03	0.681	1059.87	0.9,0.2,0.02	0.774
C100b	819.56	833.51	824.66	1.4,0.4,0.6	1.062	825.76	1.1,0.0,0.03	0.930	825.76	1.1,0.0,0.03	0.930
Average					**3.040**			**3.158**			**3.557**

Table 4.7 Relative deviations on Christofides *et al.*'s distance restricted test set

Instance	**Best**	**CW**	**AÖ**	***λ,μ,ν***	**% Imp**	**DÇ1**	***λ,μ,ν***	**% Imp**	**DÇ2**	***λ,μ,ν***	**% Imp**
CD50	555.43	618.39	589.43	1.6,0.4,2.0	4.683	582.9	1.5,1.1,0.1	5.739	582.52	1.0,1.4,-0.08	5.801
CD75	909.63	975.46	942.98	2.0,0.7,0.0	3.330	941.78	1.9,0.7,0.05	3.453	944.14	0.9,0.3,0.07	3.211
CD100a	865.94	973.94	942.69	0.7,0.3,0.0	3.209	903.91	0.7,1.2,-0.01	7.190	922.77	0.7,1.0,-0.01	5.254
CD150	1162.55	1287.64	1222.1	1.3,0.0,0.0	5.090	1218.55	1.6,0.1,0.06	5.366	1216.15	1.3,0.2,0.06	5.552
CD199	1395.85	1538.66	1485.53	1.9,0.6,0.0	3.453	1488.46	1.8,0.3,0.02	3.263	1482.89	1.7,0.3,0.01	3.625
CD120	1541.14	1592.26	1582.2	0.8,0.0,0.2	0.632	1571.14	0.9,0.4,-0.02	1.326	1572.81	0.8,0.4,-0.06	1.222
CD100b	866.37	875.75	869.61	1.2,0.2,0.0	0.701	871.32	1.3,0.3,-0.02	0.506	872.6	0.8,0.4,0.03	0.360
Average					**3.014**			**3.835**			**3.575**

Table 4.8 Performance comparison with respect to the number of problems

Test set	**# of Prob**	**DÇ1 vs. AÖ**				**DÇ2 vs. AÖ**			
		# better	**Perc (%)**	***# better or equal***	**Perc (%)**	***# better***	**Perc (%)**	***# better or equal***	**Perc (%)**
Augerat *et al* (*P*)	24	15	63	22	92	16	67	21	88
Augerat *et al* (*A*)	27	12	44	19	70	11	41	17	63
Augerat *et al* (*B*)	23	17	74	18	78	16	70	18	78
Christofides and Eilon	8	4	50	7	88	3	38	5	63
Christofides *et al* (*C*)	7	3	43	3	43	3	43	3	43
Christofides *et al* (*CD*)	7	5	71	5	71	5	71	5	71
Total	96	56	**58**	74	**74**	54	**55**	69	**68**

CHAPTER 5

CONCLUSION

Metaheuristics are proven to provide the best results for vehicle routing problems. However, they usually necessitate long computational times because of deeper exploration of the search space. On the other hand, CW savings method is very fast, and easy to implement, and can be simply adapted to various real-world situations due to its flexibility. These advantages make the method still attractive for many applications, including the commercial software. Thus, improving the performance of the CW algorithm, while preserving its simplicity, flexibility, and speed, is of interest to many researchers and practitioners.

In this thesis, firstly we address Paessens' (1988) two-term savings algorithm and propose two different enhancements. Unlike the previous ones, our first enhancement "promotes" or "penalizes" the savings value between two customers by considering the cosine value of the angle formed by the two rays originating from the depot and crossing two customers. Our computational results show that this approach provides shorter distances in many instances with improved average solution quality for all problem sets. The second enhancement enforces the customers closer to the depot to be placed into routes first. This is achieved by considering the difference between the maximum distance among the customers and the average distances of customer pairs to the depot. This approach also improves the distances in many instances and provides significantly better average solution quality in three problem sets.

Secondly, we investigate the performance of Altınel and Öncan's (2005) savings function based on the *put first larger items* idea originally proposed for Bin Packing Problem. This function considers the customer demands as well and improves the accuracy of the classical CW heuristic remarkably. However, we claim that the improvement achieved by the new function is the result of better searching the solution space using a parameterized third term rather than the "put first larger items" idea. Two modified savings functions based on the conflicting "put first smaller items" idea is presented in order to justify our claim. The computational results reveal that our savings functions work as good and even better in many problem instances.

Then, we present two alternative enhancements to Altınel and Öncan's (2005) three-term savings function. Instead of the idea of putting first larger items of Altınel and Öncan, the first enhancement aims at increasing the possibility of customers having small and large demands to be fit into the same route together and reducing capacity losses. In the second enhancement, the second term of the proposed formulation is replaced with a new term in an attempt to enforce the customers closer to the depot to be placed into routes first. Furthermore, both formulations utilize normalized distance and demand values; hence, they are independent from measurement units. Thus, the parameter intervals are robust and do not need to be readjusted for different data in different units. The computational study reveals that both approaches provide enhanced solution quality in many instances as compared to the solutions of Altinel and Öncan. Moreover, the average distances are shorter in all problem sets. The better solution quality is achieved with "slightly" more computational effort in calculating the savings values. However, the search effort for tuning the parameter values is the same, and in fact, this constitutes the most significant portion of the total computation time.

REFERENCES

Altınel, I.K., Öncan, T. (2005). A new enhancement of the Clarke and Wright savings heuristic for the capacitated vehicle routing problem. Journal of the Operational Research Society, Vol. 56, No. 8, pp. 954-961(8)

Augerat,P. (1995): VRP problem instances. Available at http://www.branchandcut.org /VRP/data (last accessed in April 2008)

Christofides N., Eilon S. (1969). An algorithm for the vehicle dispatching problem. Operational Research, Vol. 20, No. 3, pp. 309-318

Christofides N., Mingozzi A. and Toth P. (1979). The vehicle routing problem. In: Christofides N, Mingozzi A, Toth P and Sandi C (eds). Combinatorial Optimization. Wiley, Chichester, pp. 315–338.

Clarke, G., Wright, J.W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. Operations Research, Vol. 12, No. 4, pp. 568-581

Cordeau, J.F., Gendreau, M., Hertz, A., Laporte, G., Sormany, J.S. (2004). New heuristics for the vehicle routing problem. Logistic Systems: Design and Optimization. Springer US, pp. 279-297.

Dantzig, G.B., Ramser, J.H. (1959). The truck dispatching problem. Management Science, Vol. 6, No. 1, pp. 80-91

Faulin, J., Valle, A.G. (2007). Solving the capacitated vehicle routing problem using the ALGELECT electrostatic algorithm. Journal of the Operational Research Society, 10.1057/palgrave.jors. 2602502

Gaskell, T.J. (1967). Bases for vehicle fleet scheduling. Operations Research, Vol. 18, No. 3, pp. 281-295

Lenstra, J., Rinnooy, A. (1981). Complexity of vehicle routing and scheduling problems. Networks, Vol. 11, No. 2 , pp. 221 - 227.

Laporte, G., Gendreau, M., Potvin, J., Semet, F. (2000). Classical and modern heuristics for the vehicle routing problem. International Transactions in Operational Research, Vol. 7, No. 4-5, pp. 285-300.

Laporte, G., Semet, F. (2001). Classical heuristics for the vehicle routing problem. In: Toth P and Vigo D (Eds). The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications. SIAM Publishing, Philadelphia, PA, pp. 109–128.

Martello, S., Toth, P. (1990). Knapsack Problems: Algorithms and Computer Implementations. John Wiley & Sons: New York.

Paessens, H. (1988). The savings algorithm for the vehicle routing problem. European Journal of Operational Research, Vol. 34, No. 3, pp. 336-344.

Schneider, J. (2001). Searching for backbones-a high performance parallel algorithm for solving combinatorial optimization problems. Vol. 19, No. 1, pp. 121-131

Toth, P., Vigo, D. (2001). The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications. SIAM Publishing, Philadelphia, PA,

Wren, A. (1971). Computers in Transport Planning and Operation. Ian Allan, London.

Wren, A., Holiday, A. (1972) Computer scheduling of vehicles from one or more depots to a number of delivery points. Operational Research Quarterly, Vol. 23, No. 3, pp. 333-344

Yellow, P. (1970). A computational modification to the savings method of vehicle scheduling. Operational Research Quarterly, Vol. 21, No. 2, pp. 281-283

APPENDIX: Computational Results for NEG and INV for CVRP

Table A.1 Results for Augerat *et al*'s test set *P*

		AÖ		NEG			INV		
Instance	**Best**	**Dist**	**(λ, μ, ν)**	**Dist**	**(λ, μ, ν)**	**% Dev**	**Dist**	**(λ, μ, ν)**	**% Dev**
P-n16-k8	450	451.94	1.8,0.7,1.5	459.47	0.1,0.9,0.1	1.666	451.94	1.8,0.6,-0.4	0.000
P-n19-k2	212	220.64	0.9,0.5,0.0	220.64	0.2,0.1,0.0	0.000	220.64	0.3,0.7,-2.0	0.000
P-n20-k2	216	232.86	1.2,1.0,1.7	229.43	0.3,1.0,0.0	-1.473	231.48	0.5,1.1,-1.3	-0.593
P-n21-k2	211	231.54	1.4,1.0,2.0	231.54	0.2,1.0,0.0	0.000	234.43	0.2,0.5,-2.0	1.248
P-n22-k2	216	219.89	1.8,0.2,0.8	228.51	0.2,0.3,0.0	3.920	221.91	0.8,0.5,-2.0	0.919
P-n22-k8	603	589.39	0.9,0.0,0.0	589.1	0.8,0.0,1.3	-0.049	589.1	0.8,0.0,-2.0	-0.049
P-n23-k8	529	536.71	1.4,0.2,1.3	536.35	1.4,0.2,1.7	-0.067	536.35	1.1,0.0,-2.0	-0.067
P-n40-k5	458	468.2	1.1,1.0,0.3	468.2	1.4,0.8,0.0	0.000	468.2	1.4,0.8,-0.2	0.000
P-n45-k5	510	522.41	1.5,0.1,0.7	523.91	1.9,0.7,0.0	0.287	528.67	0.9,0.3,-2.0	1.198
P-n50-k10	696	712.77	1.2,0.1,0.0	712.77	1.2,0.1,0.0	0.000	712.77	1.2,0.1,-1.4	0.000
P-n50-k7	554	577.73	1.7,0.4,1.9	576.9	1.5,0.4,1.3	-0.144	577.15	1.5,0.3,-2.0	-0.100
P-n50-k8	631	646.55	1.2,0.2,0.8	646.55	1.4,0.2,0.0	0.000	646.55	1.4,0.2,-0.4	0.000
P-n51-k10	741	754.98	0.9,0.4,0.1	749.96	2.0,0.2,1.6	-0.665	749.96	1.0,0.2,-1.7	-0.665
P-n55-k10	694	715.21	1.2,0.1,1.7	713.02	1.3,0.9,1.9	-0.306	711.93	1.5,0.8,-2.0	-0.459
P-n55-k15	989	963.32	1.6,0.9,0.0	963.33	1.6,0.9,0.0	0.001	963.33	1.6,0.9,-0.6	0.001
P-n55-k7	568	587.44	1.4,0.4,1.3	589.55	1.2,0.0,0.0	0.359	586.17	1.5,0.5,-0.8	-0.216
P-n55-k8	576	588.04	1.3,0.3,1.7	594.84	1.5,0.4,0.0	1.156	592.03	1.7,0.5,-1.6	0.679
P-n60-k10	744	768.12	1.7,0.5,0.1	768.13	1.7,0.5,0.0	0.001	768.13	1.8,0.5,-0.2	0.001
P-n60-k15	968	1002.77	0.9,0.0,0.5	1006.44	0.5,0.3,0.0	0.366	999.54	1.5,0.3,-2.0	-0.322
P-n70-k10	834	853.94	0.6,0.4,0.0	853.94	0.7,0.3,0.7	0.000	853.94	0.7,0.3,-0.4	0.000
P-n76-k4	593	641.78	1.9,0.8,0.4	639.45	1.7,0.8,0.1	-0.363	630.89	1.4,1.0,-1.8	-1.697
P-n76-k5	627	652.93	1.6,0.3,0.9	651.42	1.7,0.8,0.1	-0.231	651.42	1.7,0.8,-0.6	-0.231
P-n65-k10*	792	825.92	1.9,0.7,0.7	827.08	1.6,1.1,1.9	0.140	820.54	1.6,1.3,-1.4	-0.651
P-n101-k4*	681	711.03	0.6,1.0,0.0	711.03	0.6,1.0,0.0	0.000	711.03	1.8,0.4,-0.5	0.000
Average						**0.192**			**-0.042**

* These two problems are not included in Altınel and Öncan (2005)

Table A.2 Results for Augerat *et al*'s test set *A*

		AÖ		NEG			INV		
Instance	**Best**	**Dist**	**(λ, μ, ν)**	**Dist**	**(λ, μ, ν)**	**% Dev**	**Dist**	**(λ, μ, ν)**	**% Dev**
A-n32-k5	784	828.70	0.8,0.6,0.0	828.70	0.8,0.4,0.9	0.000	828.70	0.8,0.5,-1.4	0.000
A-n33-k5	661	676.10	2.0,1.0,1.6	679.73	1.2,0.8,0.9	0.537	679.73	1.2,0.6,-1.9	0.537
A-n33-k6	742	743.21	1.2,0.0,1.0	747.32	1.5,0.5,1.9	0.553	747.32	1.4,0.4,-1.4	0.553
A-n34-k5	778	793.05	0.6,0.3,1.1	793.05	0.7,0.1,0.0	0.000	793.05	0.7,0.1,-0.3	0.000
A-n36-k5	799	806.78	0.8,0.0,0.1	806.78	0.9,0.0,0.0	0.000	806.78	0.9,0.0,-0.1	0.000
A-n37-k5	669	694.43	1.5,0.3,0.9	694.51	1.4,0.6,0.0	0.012	695.09	0.6,0.9,-0.3	0.095
A-n37-k6	949	974.56	1.0,0.0,0.4	973.96	1.1,0.2,0.3	-0.062	973.96	1.1,0.2,-0.3	-0.062
A-n38-k5	730	756.11	1.4,0.3,0.0	755.93	1.5,0.5,0.2	-0.024	755.93	1.9,0.9,-0.2	-0.024
A-n39-k5	822	848.24	1.2,0.2,0.3	851.26	1.3,0.0,1.3	0.356	848.25	1.1,0.1,-0.2	0.001
A-n39-k6	831	849.56	0.8,0.2,0.0	847.03	1.2,0.0,1.8	-0.298	847.01	1.4,0.0,-1.9	-0.300
A-n44-k7	937	959.43	1.6,0.4,2.0	969.63	1.9,0.9,0.0	1.063	969.63	1.9,0.9,-0.4	1.063
A-n45-k6	944	957.06	1.0,0.0,1.4	962.82	1.1,0.1,0.0	0.602	962.82	1.1,0.1,-0.8	0.602
A-n45-k7	1146	1166.39	1.5,0.2,2.0	1175.20	1.7,1.0,0.0	0.755	1175.59	1.4,0.5,-0.9	0.789
A-n46-k7	914	933.66	1.1,0.1,0.0	933.67	1.1,0.1,0.0	0.001	933.67	1.1,0.1,-0.2	0.001
A-n48-k7	1073	1104.24	1.7,0.7,0.0	1104.41	1.2,0.0,0.4	0.015	1104.05	1.9,0.7,-1.7	-0.017
A-n53-k7	1010	1045.47	0.7,0.0,1.5	1045.23	1.4,0.6,0.4	-0.023	1040.51	1.4,1.0,-1.3	-0.474
A-n54-k7	1167	1173.77	1.1,0.1,0.9	1185.58	1.8,0.9,0.3	1.006	1187.59	1.8,1.0,-0.6	1.177
A-n55-k9	1073	1098.51	0.9,0.1,1.1	1098.91	1.1,0.0,1.3	0.036	1099.84	1.0,0.0,-0.3	0.121
A-n60-k9	1354	1376.20	1.4,0.0,0.9	1377.89	1.4,0.1,1.9	0.123	1375.44	1.0,0.0,-0.8	-0.055
A-n61-k9	1034	1051.10	1.1,0.0,0.1	1051.33	1.1,0.0,0.9	0.022	1051.33	1.1,0.0,-0.6	0.022
A-n62-k8	1288	1347.87	1.0,0.0,0.2	1346.19	0.6,0.0,1.6	-0.125	1335.17	0.8,0.0,-2.0	-0.942
A-n63-k10	1314	1348.17	1.5,0.4,0.2	1347.95	1.9,1.1,0.0	-0.016	1347.45	1.9,1.0,-0.8	-0.053
A-n64-k9	1401	1439.75	1.9,0.9,0.1	1433.91	1.4,0.4,0.2	-0.406	1429.54	1.5,0.2,-1.6	-0.709
A-n63-k9	1616	1649.14	1.6,0.6,0.1	1648.44	1.5,0.7,0.0	-0.042	1648.44	1.5,0.7,-0.1	-0.042
A-n65-k9	1174	1202.08	0.9,0.1,0.3	1225.26	1.0,0.3,0.5	1.928	1220.58	1.1,0.3,-0.9	1.539
A-n69-k9	1159	1185.08	1.3,0.0,0.0	1185.09	1.3,0.0,0.0	0.001	1184.00	1.4,0.0,-0.8	-0.091
A-n80-k10	1763	1816.78	1.8,1.4,1.5	1812.24	1.6,0.7,1.2	-0.250	1818.65	1.8,0.7,-0.7	0.103
Average						**0.214**			**0.142**

Table A.3 Results for Augerat *et al*'s test set *B*

		AÖ		NEG			INV		
Instance	**Best**	**Dist**	**(λ, μ, ν)**	**Dist**	**(λ, μ, ν)**	**% Dev**	**Dist**	**(λ, μ, ν)**	**% Dev**
B-n31-k5	672	677.34	0.9,0.0,0.1	679.44	0.9,0.0,0.0	0.310	678.99	1.4,0.8,-0.8	0.244
B-n34-k5	788	789.85	1.2,0.0,0.0	789.85	1.2,0.0,0.0	0.000	789.85	1.4,0.0,-0.1	0.000
B-n35-k5	955	975.48	1.1,0.1,1.7	978.33	0.8,0.1,0.6	0.292	977.91	1.1,0.0,-1.2	0.249
B-n38-k6	805	824.00	1.4,0.4,0.0	824.60	1.5,0.4,0.0	0.073	823.84	1.3,0.5,-0.7	-0.019
B-n39-k5	549	555.00	1.4,0.3,0.0	554.16	1.5,0.4,0.0	-0.151	554.16	1.5,0.4,-0.1	-0.151
B-n41-k6	829	867.42	0.6,0.4,0.1	861.67	0.9,0.7,0.6	-0.663	867.42	0.6,0.5,-0.3	0.000
B-n43-k6	742	754.92	0.9,0.1,0.4	754.04	1.1,0.2,0.0	-0.117	754.04	1.0,0.1,-0.1	-0.117
B-n44-k7	909	934.68	1.9,0.9,1.8	931.95	1.7,0.7,0.4	-0.292	929.39	1.1,0.1,-0.2	-0.566
B-n45-k5	751	754.71	1.1,0.0,0.8	756.35	1.0,0.0,0.2	0.217	756.35	1.0,0.0,-0.1	0.217
B-n45-k6	678	713.24	0.9,0.6,0.0	713.24	0.9,0.6,0.0	0.000	718.37	1.0,0.0,-2.0	0.719
B-n50-k7	741	745.37	1.0,0.0,0.2	745.37	1.0,0.0,0.2	0.000	745.37	1.0,0.0,-0.1	0.000
B-n50-k8	1312	1338.34	1.9,0.9,0.8	1338.46	1.7,0.7,1.2	0.009	1335.98	1.8,0.8,-1.9	-0.176
B-n51-k7	1032	1050.00	1.5,0.0,0.0	1045.02	1.2,0.0,1.7	-0.474	1046.19	1.2,0.0,-1.1	-0.363
B-n52-k7	747	756.90	1.3,0.0,1.5	763.65	1.2,0.3,0.0	0.892	763.96	1.1,0.2,-0.1	0.933
B-n56-k7	707	722.61	0.8,0.0,0.2	723.12	1.0,0.1,1.0	0.071	721.11	1.1,0.1,-0.5	-0.208
B-n57-k7	1153	1148.98	1.1,0.0,0.5	1147.65	1.6,0.7,0.7	-0.116	1150.38	1.6,0.7,-0.1	0.122
B-n57-k9	1598	1619.72	0.9,0.0,0.0	1613.27	0.9,0.0,1.4	-0.398	1613.27	0.9,0.0,-1.1	-0.398
B-n63-k10	1496	1562.59	0.9,0.0,0.0	1559.81	0.8,0.0,0.1	-0.178	1552.13	1.0,0.0,-1.0	-0.669
B-n64-k9	861	910.07	1.1,0.8,2.0	898.02	1.8,1.1,0.3	-1.324	898.02	1.8,1.0,-0.5	-1.324
B-n66-k9	1316	1358.32	1.9,1.1,1.0	1347.59	1.6,0.9,0.0	-0.790	1347.59	1.7,1.0,-0.1	-0.790
B-n67-k10	1032	1070.30	0.8,0.0,1.8	1090.19	0.8,0.2,0.0	1.858	1090.19	0.8,0.2,-0.3	1.858
B-n68-k9	1272	1316.07	1.1,0.1,0.4	1317.23	1.0,0.0,1.0	0.088	1316.43	1.0,0.0,-0.9	0.027
B-n78-k10	1221	1261.35	1.0,0.1,0.9	1259.14	1.0,0.0,1.8	-0.175	1262.23	1.0,0.1,-1.5	0.070
Average						**-0.027**			**-0.008**

Table A.4 Results for Christofides and Eilon's test set

		AÖ		NEG			INV		
Instance	**Best**	**Dist**	**(*λ*, *μ*, *ν*)**	**Dist**	**(*λ*, *μ*, *ν*)**	**% Dev**	**Dist**	**(*λ*, *μ*, *ν*)**	**% Dev**
E-n22-k4	375	375.28	1.1,0.9,1.1	375.28	1.5,0.6,0.0	0.000	376.50	1.6,0.5,-0.4	0.325
E-n23-k3	569	573.01	1.7,0.5,0.0	573.01	1.7,0.5,0.0	0.000	573.01	1.7,0.5,-0.1	0.000
E-n30-k4	503	506.67	1.3,0.3,0.0	506.67	1.3,0.3,0.0	0.000	506.67	1.3,0.3,-0.1	0.000
E-n33-k4	835	843.10	0.9,0.1,0.0	842.84	0.8,0.1,0.5	-0.031	842.82	0.8,0.1,-0.7	-0.033
E-n76-k14	1021	1045.04	1.3,0.0,1.0	1052.33	1.1,0.1,0.0	0.698	1058.65	1.1,0.2,-0.7	1.302
E-n76-k8	735	779.42	1.0,0.5,0.1	780.08	1.0,0.5,0.0	0.085	771.82	1.1,0.5,-1.6	-0.975
E-n76-k7	682	718.88	1.6,0.8,0.2	718.88	1.8,0.8,0.0	0.000	714.18	1.7,0.8,-1.5	-0.654
E-n101-k14	1071	1126.39	0.8,0.6,0.3	1120.59	0.7,0.6,0.0	-0.515	1120.59	0.7,0.6,-0.1	-0.515
Average						**-0.005**			**-0.022**

Table A.5 Results for Christofides *et al*'s test set *C*

		AÖ		NEG			INV		
Instance	**Best**	**Dist**	**(*λ*, *μ*, *ν*)**	**Dist**	**(*λ*, *μ*, *ν*)**	**% Dev**	**Dist**	**(*λ*, *μ*, *ν*)**	**% Dev**
C50	525	555.55	1.7,0.2,0.6	554.17	0.7,0.8,1.9	-0.248	555.55	1.5,0.3,-1.5	0.000
C75	835	860.21	1.2,0.2,0.7	864.79	1.4,0.9,0.0	0.532	862.81	1.1,0.3,-1.9	0.302
C100a	826	867.35	1.2,0.6,0.1	865.06	1.4,0.3,0.4	-0.202	869.01	0.8,0.2,-0.9	0.191
C150	1028	1094.06	1.3,0.1,0.3	1102.89	0.3,1.0,0.1	0.807	1101.13	1.4,0.4,-1.8	0.646
C199	1291	1359.78	1.3,0.0,1.1	1370.17	1.1,0.0,0.3	0.764	1360.59	1.6,0.5,-2.0	0.060
C120	1042	1057.8	1.1,0.1,0.3	1067.18	1.1,0.1,0.0	0.887	1052.19	1.0,0.0,-1.5	-0.530
C100b	820	824.66	1.4,0.4,0.6	827.36	1.3,0.3,1.0	0.327	824.23	1.2,0.2,-2.0	-0.052
Average						**0.193**			**0.057**

Table A.6 Results for Christofides *et al*'s test set *CD*

		AÖ		**NEG**			**INV**		
Instance	**Best**	**Dist**	**(λ, μ, v)**	**Dist**	**(λ, μ, v)**	**% Dev**	**Dist**	**(λ, μ, v)**	**% Dev**
CD50	555	589.42	1.6,0.4,2.0	580.49	1.5,1.4,1.0	-1.515	590.38	1.3,0.7,-2.0	0.163
CD75	910	942.98	2.0,0.7,0.0	942.98	1.9,0.6,0.0	0.000	942.98	1.9,0.6,-2.0	0.000
CD100a	866	942.69	0.7,0.3,0.0	909.99	0.5,1.1,0.2	-3.469	917.66	0.7,1.2,-0.1	-2.655
CD150	1163	1222.10	1.3,0.0,0.0	1220.91	1.4,0.0,0.4	-0.097	1221.35	1.8,0.2,-1.4	-0.061
CD199	1396	1485.53	1.9,0.6,0.0	1484.98	1.8,0.3,1.0	-0.037	1481.78	1.3,0.0,-1.9	-0.252
CD120	1541	1582.20	0.8,0.0,0.2	1583.77	0.7,0.1,0.0	0.099	1571.65	0.9,0.2,-1.3	-0.667
CD100b	866	869.61	1.2,0.2,0.0	871.32	1.3,0.3,0.0	0.197	869.62	1.3,0.3,-1.0	0.001
Average						**-0.073**			**-0.176**