A robust enhancement to the Clarke-Wright savings algorithm

Tamer Doyuran* and Bülent Çatay
Sabanci University, Faculty of Engineering and Natural Sciences
Tuzla, Istanbul, 34956, Turkey

Abstract: We address the Clarke and Wright (CW) savings algorithm proposed for the Capacitated Vehicle Routing Problem (CVRP). We first consider a recent enhancement which uses the put first larger items idea originally proposed for the bin packing problem and show that the conflicting idea of putting smaller items first has a comparable performance. Next, we propose a robust enhancement to the CW savings formulation. The proposed formulation is normalized to efficiently solve different problems, independent from the measurement units and parameter intervals. To test the performance of the proposed savings function, we conduct an extensive computational study on a large set of well-known instances from the literature. Our results show that the proposed savings function provides shorter distances in the majority of the instances and the average performance is significantly better than previously presented enhancements.

Keywords: Vehicle routeing, Clarke-Wright savings algorithm, heuristics.

1. Introduction

The capacitated vehicle routing problem (CVRP) is a well-known NP-hard problem 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. (Toth and Vigo, 2002). Since the exact algorithms proposed for solving CVRP are not practical for large instances significant research efforts have been spent on heuristic methods to find good quality solutions fast. An extensive study about the classical heuristics proposed in the literature can be found in Laporte and Semet (2001).

Among these heuristics, the well-known Clarke and Wright (1964) (CW) algorithm is one of the earliest and most widely used heuristics due to its speed, simplicity, and ease of adjustment to handle various constraints in real-life applications. It is based on the feasible

* *Corresponding author*: Tel.: +90 216.483.9531; fax: +90 216.483.9550.
E-mail addresses: tamerdoyuran@su.sabanciuniv.edu (T. Doyuran), catay@sabanciuniv.edu (B. Çatay)

merging of sub-tours using a savings criterion, which refers to the cost saving achieved by combining two routes and using one vehicle rather than two. Several enhancements of the CW algorithm have been proposed in the literature by parameterizing the savings formula and adding new terms to it. The first was introduced by Gaskell (1967) and Yellow (1979) who parameterized the CW savings formula with the aim of expanding the exploration ability of the algorithm. Paessens (1988) added a second term to Gaskell's and Yellow's formula in an attempt to collect more information about the distribution of the customers. In a recent paper, Altınel and Öncan (2005) introduced a third term and combined the distance and customer demand information in the savings function.

This paper is motivated by the works of Paessens (1988) and Altınel and Öncan (2005). We first address the "put first larger items" idea of Altınel and Öncan and present two modified savings functions to show that the conflicting idea of *putting first smaller items* has a comparable performance. Next, we propose a new and robust enhancement to improve the performance of Paessens' and Altınel and Öncan's savings heuristics. The remainder of this paper is organized as follows: Section 2 provides a brief overview of the recent enhancements of the CW algorithm. In Section 3, we present two modified savings functions utilizing the conflicting "put first smaller items" idea and test their performance. In Section 4 we propose an enhanced three-parameter savings function as a robust alternative to Altınel and Öncan's formulation. Section 5 provides the experimental analysis comparing our saving function to that of Paessens' and Altınel and Öncan's using the well-known benchmark instances from the literature. Finally, we provide our concluding remarks in the last section.

2. Overview of the recent enhancement of Clarke-Wright savings heuristics

Two versions of the CW algorithm are proposed in the literature: parallel and sequential. The best feasible merges of sub-tours are performed in the parallel approach whereas the route extension is considered in the sequential approach. As pointed out in Laporte and Semet (2001) the parallel version dominates the sequential savings method. Since the CW algorithm is a well-known algorithm in the literature we refer the reader to Laporte and Semet (2001) and Altınel and Öncan (2005) for further details. The CW savings function is the following:

$$S_{ij} = [c_{0i} + c_{i0} + c_{0j} + c_{j0}] - [c_{0i} + c_{ij} + c_{j0}] = c_{i0} + c_{0j} - c_{ij} \qquad (1)$$

where c_{i0} is the distance of customer i to the depot, c_{0j} is the distance of the depot to customer j, and c_{ij} is the distance between customers i and j.

The CW algorithm is eager to construct good quality routes at the early stages. In the case when the distances of customers i and j to the depot are long whereas the distance between them is short the corresponding savings value will be large, placing it at the top of the savings list. In other words, the outermost customers (i.e. customers with shorter distance between relative to their distances to the depot) are forced to be placed in the same route at the early stages. Eventually, the algorithm constructs circular shaped routes beginning from the outermost customers and proceeds towards the inner customers. Having noticed this weakness of CW method, which prevents the merging of possible less expensive routes, Gaskell (1967) and Yellow (1979) parameterized the savings formulation as follows:

$$S_{ij} = \left[c_{i0} + c_{0j} - \lambda c_{ij}\right] \tag{2}$$

Their motivation in using the positive 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 second term to the Gaskell's and Yellow's formula in an attempt to collect more information about the distribution. The proposed savings function is the following:

$$S_{ij} = \left[c_{i0} + c_{0j} - \lambda c_{ij}\right] + \left[\mu\left|c_{i0} - c_{0j}\right|\right] \tag{3}$$

where μ in the second term 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 the certain customer pairs in many cases, a customer very close to the depot and another one very distant from the depot as such.

Recently, Altınel and Öncan (2005) proposed an enhancement to Paessens' formula by introducing a third term which considers the demands of customer pairs and the overall average demand. Inspired from the *first fit decrease* idea of Martello and Toth (1990) originally used for the bin packing problem (BPP) they adopt a *put first larger items* approach which gives priority to the customers with large demands. The new formula is as follows:

$$S_{ij} = \left[c_{i0} + c_{0j} - \lambda c_{ij}\right] + \left[\mu\left|c_{i0} - c_{0j}\right|\right] + \left[\upsilon\frac{d_i + d_j}{\bar{d}}\right] \tag{4}$$

In the last (third) term, d_i (d_j) denotes the demand of customer i (j), $\bar{d}$ is the average demand, and v is the new non-negative parameter. The third term in this function gives a placement priority to customers with larger demands, which are normalized with the average demand.

In the next section, we present two modified savings functions based on the conflicting idea of putting first smaller items and test their performance.

3. Proposed modifications to Altınel and Öncan's savings function

In our first modification, we penalize the customers with larger demands by subtracting the last term. This penalty-based formulation aims at promoting the customers with smaller demands by penalizing the customer pairs with larger demands more than the customer pairs with smaller demands. The formulation is as follows:

$$S_{ij} = [c_{i0} + c_{0j} - \lambda c_{ij}] + [\mu |c_{i0} - c_{0j}|] - \left[v \frac{d_i + d_j}{\bar{d}} \right] \tag{5}$$

In our second modification, we give placement priority to customers with smaller demands by using the ratio of the average demand to the sum of the demands of customer *i* and customer *j* instead of using the ratio of the sum of demands of customer *i* and customer *j* to the average demand in (4). In other words, in the last term we use the inverse of the demand information of Altınel and Öncan's function parameterized with the same *v* value. The formulation is as follows:

$$S_{ij} = [c_{i0} + c_{0j} - \lambda c_{ij}] + [\mu |c_{i0} - c_{0j}|] + \left[v \frac{\bar{d}}{d_i + d_j} \right] \tag{6}$$

To test the performance of the proposed two modified savings functions we repeat the computational study of Altınel and Öncan (2005) using the same parameter setting: λ, μ, and v, respectively, are adjusted in the intervals [0.1, 2], [0, 2], and [0, 2], respectively, with an increment of 0.1. The parallel version of the savings algorithm is implemented. The code is written in C++. The test instances include Augerat's (1995) data sets A, B, and P, Christofides and Eilon's (1969) data set, and Christofides et al.'s (1979) test set C. All of the data sets are available at http://neo.lcc.uma.es/radi-aeb/WebVRP. In all instances, distances and customer demands are integer numbers. The number of customers varies between 15 and 199.

***** INSERT TABLE 1 ABOUT HERE *****

The results are summarized in Table 1. The instances are identified in the first column: Aug, ChrEil, and Chr denote the test sets of Augerat et al., Christofides and Eilon, and Christofides et al., respectively. NEG and INV refer to the results obtained using the savings functions (5) and (6), respectively, and AÖ denotes the results found by using Altınel and Öncan's savings function (4). "# of Prob" column shows the number of problem instances and

"Avg %Dev" column gives the average deviation of distances in NEG and INV from AÖ and is calculated as (NEG/AÖ – 1) and (INV/AÖ – 1), respectively. Note that a negative deviation indicates that NEG (INV) finds a shorter distance than AÖ. "# better" column reports the number of instances in which NEG (INV) finds shorter distance than AÖ and "# better or equal" column reports the number of instances in which NEG (INV) finds shorter distance than or same distance as AÖ.

The results show that INV gives a better average distance than AÖ in one problem set (Aug B) whereas NEG outperforms AÖ in three problem sets (Aug B, Aug P, and ChrEil). The average deviation values do not reveal any significant difference in employing either approach. If we make a comparison on the number of instances NEG and INV perform better than AÖ, we see that NEG gives the best distance in 44% of the problems while AÖ performs better in 42% and INV provides the best distance in 36% of the problems while AÖ finds the best distance in 34%. The performance of NEG is significantly better than AÖ in Aug B (48% vs. 39%), Aug P (46% vs. 17%), and ChrEil (50% vs. 25%). We also observe that NEG and INV perform better than or as good as AÖ in 66% and 58% of the test instances, respectively. These results indicate that both NEG and INV have a comparable performance to that of AÖ and NEG performs slightly better than INV.

4. New enhancement on the three-parameter savings function

These results in Section 3 confirm that the idea of putting smaller items first works as well as putting first larger items idea and even better in some instances, particularly in the case of savings function (6). Therefore, an approach that gives a higher placement priority to customers with large demands or small demands together may, in fact, provide improved solution quality.

Furthermore, the above mentioned effect of the unfair contribution of the second term in Paessens' function may be weakened by utilizing the cosine value of the polar coordinate angles of the customers with the depot as a coefficient (Doyuran and Çatay, 2008). The idea is similar to that of the well-known sweep algorithm (Wren and Holiday, 1972). 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, the new 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. This new approach basically ensures the customers to be placed in the same route if they are radially close to each other.

***** INSERT FIGURE 1 ABOUT HERE *****

Fig. 1 illustrates the effect of multiplying the second term of Paessens' savings function (3) by the cosine value of the angle formed by the two rays originating from the depot and crossing the customers i and j. Fig. 1(a) shows the two routes obtained by applying the classical CW algorithm to an instance of 22 customers. The depot is denoted as 0. The classical CW algorithm provides a total distance of 324.87. Fig. 1(b) depicts the solution given by the sweep algorithm. It corresponds to a total distance of 358.45. The result of the angle-based approach is illustrated in Fig. 1(c). 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 may be obtained by tuning the parameters. We observe that the classical CW algorithm forms routes that are more circumferenced since the savings are high at the top of 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 distances between the customers and the distance of the customers to the depot. Consequently, the total routing cost becomes highest due to lack of information used. The proposed approach, however, takes advantage of the information used in both the sweep and CW heuristics and provides the shortest distance. Fig. 1(c) shows how the routes are reshaped and their circumferenced characteristics disappear by integrating the cosine value.

The second term of the proposed savings function 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. Our motivation is to give an early placement priority to the customers located near the depot. Keeping the customer pairs in the vicinity of the depot together may enable a vehicle to visit more customers before the route ends at the depot. The last term is demand-based as it is the case in Altınel and Öncan; however, the underlying idea is quite different: parameter ν is allowed to take both positive and negative values (after having observed the performance of NEG in Section 3). As far as the positive values are concerned, the saving 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 saving list. The proposed formulation is as follows:

$$S_{ij} = \left[\frac{c_{i0} + c_{0j} - \lambda c_{ij}}{c^{max}}\right] + \left[\mu \frac{cos\theta_{ij}\left|c^{max} - (c_{i0} - c_{0j})/2\right|}{c^{max}}\right] + \left[v \frac{\left|\bar{d} - (d_i + d_j)/2\right|}{d^{max}}\right] \qquad (7)$$

where θ_{ij} in the second term is the angle formed by the two rays originating from the depot and crossing the customers i and j. c^{max} represents the longest distance among all customer pairs, and d^{max} denotes the maximum demand among all 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, the absolute value of the term is utilized.

In CVRP, one of the most challenging aspects in using the 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. Following the results in Section 3, the last term in (7) aims at increasing the possibility of customers having small demands and large demands to be fitted 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 the ones with small demands or large demands will be penalized most. In this case, the former customer pairs move towards the top of the saving list while the latter ones go downwards. However, the idea of keeping customer pairs having small demands and large demands close in the saving list is preserved. Keeping the customer pairs having smaller and larger demands close near the bottom of the savings list improves the capacity utilization particularly towards the end of the route.

A drawback of the savings function (4) 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 well. Hence, it will need to be readjusted in a new search interval, requiring additional computational effort. Therefore, we propose a normalized savings function where the distances are divided by the maximum distance and the demands are divided by the maximum demand. (7) is a robust formulation independent from the measurement units since all distances and demands are represented within a unit measure.

In what follows is a detailed experimental analysis to investigate the performance of the proposed enhancement on the well-known benchmark instances utilized by Altınel and Öncan (2005).

5. Experimental analysis

To make a fair comparison, we have conducted our experiments the same way Altınel and Öncan did. We adopted the parallel version and used the same number of parameter values. Since the search effort is the same we do not report the computation times. The algorithm is coded in C++. Note 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, longer, or the same distances with different parameter values. The reason is that some instances are not very sensitive to changing parameter values and two or more parameter triplets may provide the same best distance and/or the implementation of the algorithm on the computer code in the code may cause this difference. 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 by our code using Altınel and Öncan's formula.

***** INSERT TABLE 2 ABOUT HERE *****

The data set and the notation are the same as in Section 3. To make an overall assessment of the performance of the three methods we report in Table 2 the average deviations with respect to different data sets as well as the number of instances in which ROBUST performs "better than" and "better than or same as" P and AÖ. Here, ROBUST refers to the proposed enhancement whereas P and AÖ are Paessens' and Altınel and Öncan's algorithms, respectively. The detailed results are given in the Appendix (Tables A1-A6).

The results in Table 2 show that the average performance of ROBUST is better than that of P and AÖ in all of the benchmarked problem sets. The difference is particularly significant for Aug P, Chr C and CD test sets: ROBUST outperforms P (AÖ) by 1.32% (0.99%), 1.26% (0.56%), and 0.75% (0.59%), respectively. Overall, the average performance of ROBUST is 0.75% and 0.42% better than that of P and AÖ, respectively. The results also show that ROBUST outperforms P in 73% of the instances and AÖ in 57%. Moreover, ROBUST provides "better or equal quality" solutions in 83% and 71% of the problems, respectively.

AÖ gives placement priority to the customers with high demands. At the early phase of the route construction, this approach disregards the customers with low demands that can otherwise be fitted into the routes, increasing the capacity utilization of the vehicles. We observe that our formulation which attempts to keep customers with similar demands together in the savings list extends the exploration ability of the algorithm and is able to find better combinations of routes. Furthermore, the second term in P and AÖ emphasizes the construction of routes starting from the outermost customers radially distant from the depot. However, the idea of early placement priority of the customers near the depot in our approach enables these customers to be inserted into the routes as soon as possible. By doing so, the algorithm eliminates the additional routes that would be constructed by innermost customers close to the depot, and hence may obtain tours with shorter distance.

To further evaluate the contribution of the ideas implemented through the second and third terms in the proposed savings function we investigated the values that the parameters μ and v reported in the Appendix take. If the parameter value is zero, then the associated idea does not contribute to the solution obtained. The tables in the Appendix indicate that v in ROBUST is zero in only 4 out of 96 problems (compared to 27 problems in the case of AÖ). Similarly, μ is zero in only 5 problems (compared to 25 problems in the case of AÖ). These results reveal that both terms in ROBUST play an integral role in the solution quality. Note here that AÖ becomes P when v=0 whereas ROBUST reduces to a new two-term savings formulation. In fact, in two instances where v=0 (namely, A-n38-k5and B-n63-k10) this two-term reduction of ROBUST still provides the best distance. Finally, we see that in almost half of the problems (49 out of 96) the best distance was obtained with a negative v. This actually confirms the contribution of our third term and supports the underlying idea behind it, as explained in Section 4.

We also investigated the effect of increasing the computational effort twice by extending the interval of parameter v to [-0.2, 0.2]. However, this extended interval has only a contribution of 0.14% to the average distance of all benchmark instances. Furthermore, to test the sensitivity of Altınel and Öncan's algorithm to varying measurement units we multiplied all distances by 1000 (e.g. converting kilometers to meters) and conducted a computational study on a subset of instances. The best solutions were obtained with v=0, as expected. Thus, the third term does not have any impact on the algorithm and the savings function, in fact, reduces to that of Paessens' unless a new parameter interval is investigated. Since our savings functions consist of normalized terms, the same parameter intervals can still be used and the results are not affected by the change in the measurement units. In sum, these results show that the proposed savings function with a normalization procedure and newly integrated demand idea is robust and capable of providing shorter distances.

6. Conclusion

In this study, we discussed several enhancements proposed for the CW algorithm. One of those enhancements is the three-term savings function proposed by Altınel and Öncan's (2005). This paper uses put first larger items idea originally proposed for the BPP. We show that an alternative approach which puts smaller items first works as well as the idea of Altınel and Öncan. Then, we proposed a robust enhancement to CW savings function. Instead of the idea of putting first larger items, our enhancement aims at increasing the possibility of customers having small and large demands to be fitted into the same route together and reducing capacity losses. In addition, it tries to place the customers near the depot into routes first. Furthermore, our algorithm utilizes normalized distance and demand values and is independent from the 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 the proposed savings function outperforms that of both Paessens (1988) and Altınel and Öncan (2005) in many instances and provide shorter average distance an all of the benchmark data sets. The better solution quality is achieved with "negligible" additional computational effort in calculating the saving values as compared to Altınel and Öncan's algorithm.

Appendix

Tables A1-A5 consist of only capacity restricted instances whereas the instances in Table A6 include a unique data set CD of Christofides et al.'s with maximum route length constraint (referred to as Chr CD). This data set was not reported in Altınel and Öncan (2005) but we prefer to test it to observe the performance of our algorithm when a maximum route length is imposed. In all the tables, the instances are represented in the first column and best-known results and the results given by classical CW method are included in the second and third column, respectively. The results obtained using Paessens' and Altınel and Öncan's savings function are denoted as P and AÖ, respectively, and reported with the corresponding parameter values (λ, μ,) and (λ, μ, v), respectively. ROBUST column shows the results obtained using the proposed savings function (7) along with the corresponding parameter values as well. Our experiments revealed that the parameter v 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 P, AÖ, and ROBUST, respectively, in comparison with CW and is calculated as (CW-P)/CW, (CW-AÖ)/CW, and (CW-ROBUST)/CW, respectively.

***** INSERT TABLES A1-A6 ABOUT HERE *****

References

Altınel I K and Öncan T (2005). A new enhancement of the Clarke and Wright savings heuristic for the capacitated vehicle routing problem. *J Opl Res Soc* **56**: 954-961.

Christofides N and Eilon S (1969). An algorithm for the vehicle dispatching problem. *Opl Res Quart* **20**: 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 and Wright J W (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Opns Res* **12**: 568-581.

Dantzig G B and Ramser J H (1959). The truck dispatching problem. *Man Sci* **6**: 80-91.

Doyuran T and Çatay B (2008). Two enhanced savings functions for the Clark-Wright algorithm. In: Blecker T, Kersten W and Gertz C (eds). *Management in Logistics Networks and Nodes: Concepts, Technology and Applications*. Series on Operations and Technology Management 8. Erich Schmidt Verlag, Berlin, pp 245-258

Gaskell T J (1967). Bases for vehicle fleet scheduling. *Opns Res* **18**: 281-295.

Laporte G, Gendreau M, Potvi J and Semet F (2000). Classical and modern heuristics for the vehicle routing problem. *Int Trans Opns Res* **7**: 285-300.

Laporte G and 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.

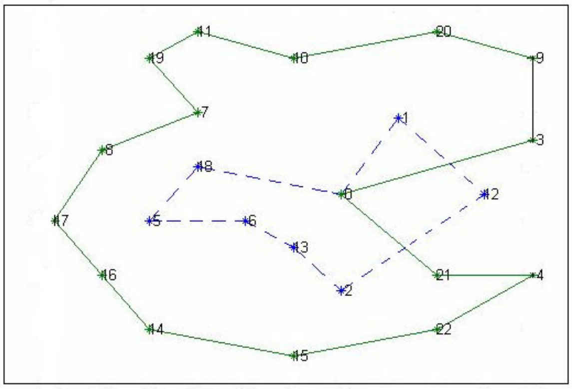
Martello S and Toth P (1990). *Knapsack problems: algorithms and computer implementations*. John Wiley & Sons: New York, NY.

Paessens H (1988). The savings algorithm for the vehicle routing problem. *Eur J Opl Res* **34**: 336-344.

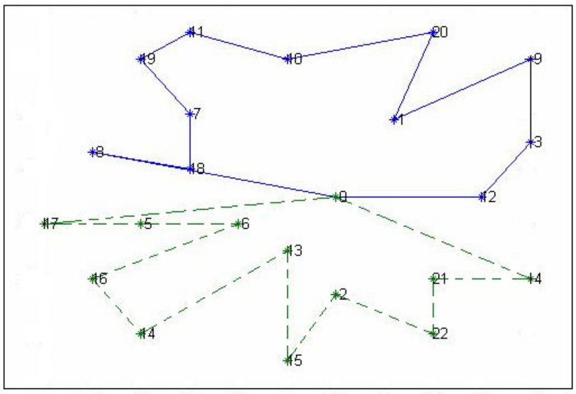
Toth P and Vigo D (2002). *The Vehicle Routing Problem*. SIAM Monographs on Discrete Mathematics and Applications. SIAM Publishing: Philadelphia, PA.

Wren A and Holiday A (1972). Computer scheduling of vehicles from one or more depots to a number of delivery points. *Opl Res Quart* **23**: 333-344.

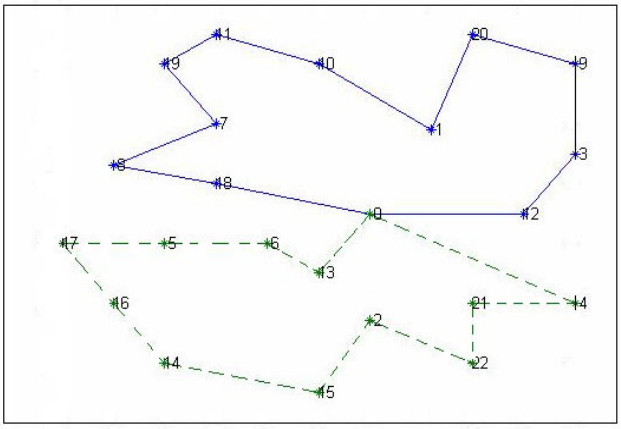
Yellow P (1970). A computational modification to the savings method of vehicle scheduling. *Opl Res Quart* **21**: 281-283.



(a) Classical CW algorithm



(b) Sweep algorithm



(c) Angle-based approach

Figure 1. Comparison of the routes formed by three approaches. (a) Classical CW algorithm, (b) Sweep algorithm, (c) Proposed approach

Table 1. Comparison of NEG and INV vs. AÖ

Test set	# of Prob	NEG vs. AÖ					INV vs. AÖ				
		Avg %Dev	# better	%	# better or equal	%	Avg %Dev	# better	%	# better or equal	%
Aug (A)	27	0.142	11	40.7	16	59.3	0.214	9	33.3	13	48.1
Aug (B)	23	-0.015	11	47.8	14	60.9	-0.038	11	47.8	14	60.9
Aug (P)	24	-0.042	11	45.8	20	83.3	0.192	8	33.3	17	70.8
ChrEil	8	-0.069	4	50.0	6	75.0	0.030	2	25.0	6	75.0
Chr (C)	7	0.088	2	28.6	3	42.9	0.410	2	28.6	2	28.6
All	**89**	**0.021**	**39**	**43.8**	**59**	**66.3**	**0.162**	**32**	**36.0**	**52**	**58.4**

Table 2. Comparison of the proposed approach to that of Paessens (1988) and Altınel and Öncan (2005)

Test set	# of Prob	ROBUST vs. P					ROBUST vs. AÖ				
		Avg %Dev	# better	%	# better or equal	%	Avg %Dev	# better	%	# better or equal	%
Aug (*A*)	27	-0.364	19	70.4	20	74.1	-0.075	12	44.4	17	63.0
Aug (*B*)	23	-0.439	18	78.3	20	87.0	-0.168	16	69.6	17	73.9
Aug (*P*)	24	-1.321	16	66.7	21	87.5	-0.987	16	66.7	21	87.5
ChrEil	8	-0.383	5	62.5	7	87.5	-0.152	3	37.5	5	62.5
Chr (C)	7	-1.261	7	100.0	7	100.0	-0.561	3	42.9	3	42.9
Chr (CD)	7	-0.745	5	71.4	5	71.4	-0.592	5	71.4	5	71.4
All	**96**	**-0.752**	**70**	**72.9**	**80**	**83.3**	**-0.423**	**55**	**57.3**	**68**	**70.8**

Table A1. Relative deviations on Augerat *et al.*'s test set P

Instance	Best	CW	P	λ,μ	% Imp	AÖ	λ,μ,ν	% Imp	ROBUST	λ,μ,ν	% Imp
P-n16-k8	450	478.77	451.94	2.0,0.5	5.604	451.94	1.8,0.7,1.5	5.604	451.94	0.1,1.6, 0.04	5.604
P-n19-k2	212	237.89	220.64	0.9,0.5	7.251	220.64	0.9,0.5,0.0	7.251	220.64	0.1,1.3,-0.10	7.251
P-n20-k2	216	234.00	233.99	0.2,0.7	0.004	232.86	1.2,1.0,1.7	0.487	224.13	0.1,1.4, 0.09	4.218
P-n21-k2	211	236.19	236.18	0.2,0.7	0.004	231.54	1.4,1.0,2.0	1.969	212.71	0.8,1.4, 0.01	9.941
P-n22-k2	216	239.50	219.89	1.9,0.7	8.188	219.89	1.8,0.2,0.8	8.188	217.87	0.2,1.5,-0.04	9.031
P-n22-k8	603	590.62	589.39	0.9,0.0	0.208	589.39	0.9,0.0,0.0	0.208	588.79	0.1,0.9, 0.03	0.310
P-n23-k8	529	539.48	536.71	1.2,0.0	0.513	536.71	1.4,0.2,1.3	0.513	536.35	0.1,0.8, 0.05	0.580
P-n40-k5	458	518.37	468.20	1.2,1.0	9.678	468.20	1.1,1.0,0.3	9.678	470.20	0.7,1.4, 0.07	9.293
P-n45-k5	510	572.95	523.91	1.9,0.7	8.559	522.41	1.5,0.1,0.7	8.821	521.31	1.2,1.3,-0.10	9.013
P-n50-k10	696	739.84	712.77	1.2,0.1	3.659	712.77	1.2,0.1,0.0	3.659	712.77	0.8,0.4,-0.04	3.659
P-n50-k7	554	597.03	578.94	1.7,0.4	3.030	577.73	1.7,0.4,1.9	3.233	577.73	0.7,0.8,-0.01	3.233
P-n50-k8	631	674.34	646.54	1.4,0.2	4.123	646.55	1.2,0.2,0.8	4.121	646.55	0.6,0.7,-0.04	4.121
P-n51-k10	741	790.97	754.97	1.1,0.3	4.551	754.98	0.9,0.4,0.1	4.550	747.25	0.7,0.6, 0.09	5.527
P-n55-k10	694	736.45	716.06	1.4,0.3	2.769	715.21	1.2,0.1,1.7	2.884	709.33	1.8,0.8, 0.05	3.683
P-n55-k15	989	978.07	963.32	1.4,0.5	1.508	963.32	1.6,0.9,0.0	1.508	959.93	0.2,1.2, 0.08	1.855
P-n55-k7	568	618.68	589.54	1.4,0.1	4.710	587.44	1.4,0.4,1.3	5.049	584.23	1.4,0.1, 0.10	5.568
P-n55-k8	576	631.67	594.84	1.4,0.5	5.831	588.04	1.3,0.3,1.7	6.907	594.30	1.3,0.1,-0.10	5.916
P-n60-k10	744	800.20	769.27	1.5,0.5	3.865	768.12	1.7,0.5,0.1	4.009	765.08	0.6,0.8, 0.09	4.389
P-n60-k15	968	1016.96	1006.94	0.8,0.0	0.985	1002.77	0.9,0.0,0.5	1.395	996.87	0.5,1.2,-0.10	1.975
P-n70-k10	834	896.86	853.94	0.6,0.4	4.786	853.94	0.6,0.4,0.0	4.786	855.10	0.3,0.4, 0.01	4.656
P-n76-k4	593	688.34	643.14	1.7,0.8	6.567	641.78	1.9,0.8,0.4	6.764	616.30	1.0,0.8,-0.05	10.466
P-n76-k5	627	709.38	655.03	2.0,0.7	7.662	652.93	1.6,0.3,0.9	7.958	647.31	0.6,0.9,-0.09	8.750
P-n65-k10*	792	844.61	829.17	0.3,1.0	1.828	825.92	1.9,0.7,0.7	2.213	815.96	0.3,1.3,-0.04	3.392
P-n101-k4*	681	765.38	722.83	1.2,0.2	5.559	711.03	0.6,1.0,0.0	7.101	702.04	1.7,0.3,-0.10	8.276
Average					**4.227**			**4.536**			**5.446**

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

Table A2. Relative deviations on Augerat *et al.*'s test set A

Instance	Best	CW	P	λ,μ	% Imp	AÖ	λ,μ,ν	% Imp	ROBUST	λ,μ,ν	% Imp
A-n32-k5	784	843.69	828.70	0.8,0.6	1.777	828.70	0.8,0.6,0.0	1.777	828.70	0.3,0.5, 0.03	1.777
A-n33-k5	661	712.05	679.72	1.4,0.8	4.540	676.10	2.0,1.0,1.6	5.049	676.10	0.3,0.9,-0.01	5.049
A-n33-k6	742	776.26	747.32	1.6,0.5	3.728	743.21	1.2,0.0,1.0	4.258	746.99	0.1,1.8,-0.08	3.771
A-n34-k5	778	810.41	793.05	0.7,0.1	2.142	793.05	0.6,0.3,1.1	2.142	793.05	0.6,0.2,-0.06	2.142
A-n36-k5	799	828.47	806.78	0.9,0.0	2.618	806.78	0.8,0.0,0.1	2.618	806.78	0.6,0.4,-0.02	2.618
A-n37-k5	669	707.81	695.08	0.7,0.9	1.799	694.43	1.5,0.3,0.9	1.890	694.44	1.0,0.3,-0.09	1.889
A-n37-k6	949	976.61	976.01	1.0,0.1	0.061	974.56	1.0,0.0,0.4	0.210	976.61	0.8,0.3,-0.04	0.000
A-n38-k5	730	768.13	755.94	1.4,0.3	1.587	756.11	1.4,0.3,0.0	1.565	755.94	0.6,0.8, 0.00	1.587
A-n39-k5	822	901.99	851.25	1.6,0.1	5.625	848.24	1.2,0.2,0.3	5.959	843.23	0.3,1.5, 0.09	6.514
A-n39-k6	831	863.08	849.55	0.8,0.2	1.568	849.56	0.8,0.2,0.0	1.566	849.90	0.5,0.4,-0.09	1.527
A-n44-k7	937	976.04	968.84	2.0,0.9	0.738	959.43	1.6,0.4,2.0	1.702	957.03	0.7,0.8,-0.04	1.948
A-n45-k6	944	1006.45	957.05	1.1,0.1	4.908	957.06	1.0,0.0,1.4	4.907	957.06	1.0,0.1, 0.01	4.907
A-n45-k7	1146	1199.98	1169.00	1.9,0.9	2.582	1166.39	1.5,0.2,2.0	2.799	1168.97	1.1,0.6, 0.05	2.584
A-n46-k7	914	939.74	933.66	1.1,0.1	0.647	933.66	1.1,0.1,0.0	0.647	929.42	0.8,0.1, 0.08	1.098
A-n48-k7	1073	1112.82	1104.23	1.7,0.7	0.772	1104.24	1.7,0.7,0.0	0.771	1103.99	0.7,0.5,-0.04	0.793
A-n53-k7	1010	1099.45	1045.98	1.5,0.6	4.863	1045.47	0.7,0.0,1.5	4.910	1048.79	0.8,0.2,-0.02	4.608
A-n54-k7	1167	1197.92	1188.64	1.7,0.9	0.775	1173.77	1.1,0.1,0.9	2.016	1172.27	0.8,0.4,-0.02	2.141
A-n55-k9	1073	1099.84	1099.55	1.3,0.2	0.026	1098.51	0.9,0.1,1.1	0.121	1099.56	1.1,0.0, 0.06	0.025
A-n60-k9	1354	1421.88	1389.59	1.6,1.0	2.271	1376.20	1.4,0.0,0.9	3.213	1379.86	0.9,0.8,-0.10	2.955
A-n61-k9	1034	1102.23	1051.37	1.1,0.0	4.614	1051.10	1.1,0.0,0.1	4.639	1051.06	0.9,0.3, 0.07	4.642
A-n62-k8	1288	1352.81	1351.11	1.2,0.2	0.126	1347.87	1.0,0.0,0.2	0.365	1326.54	0.7,0.4,-0.01	1.942
A-n63-k10	1314	1352.48	1349.58	2.0,1.2	0.214	1348.17	1.5,0.4,0.2	0.319	1347.30	1.0,0.1,-0.04	0.383
A-n64-k9	1401	1486.92	1442.44	1.1,0.5	2.991	1439.75	1.9,0.9,0.1	3.172	1442.66	1.0,0.1, 0.07	2.977
A-n63-k9	1616	1687.96	1648.92	1.6,0.6	2.313	1649.14	1.6,0.6,0.1	2.300	1652.42	0.3,1.5, 0.04	2.106
A-n65-k9	1174	1239.42	1224.71	1.0,0.2	1.187	1202.08	0.9,0.1,0.3	3.013	1197.49	0.7,0.4, 0.02	3.383
A-n69-k9	1159	1210.78	1185.08	1.3,0.0	2.123	1185.08	1.3,0.0,0.0	2.123	1181.91	1.1,0.2,-0.05	2.384
A-n80-k10	1763	1860.94	1818.64	1.8,0.7	2.273	1816.78	1.8,1.4,1.5	2.373	1811.56	0.6,0.9,-0.03	2.653
Average					**2.180**			**2.460**			**2.534**

Table A3. Relative deviations on Augerat *et al.*'s test set B

Instance	Best	CW	P	*λ,μ*	% Imp	AÖ	*λ,μ,ν*	% Imp	ROBUST	*λ,μ,ν*	% Imp
B-n31-k5	672	681.16	679.43	0.9,0.0	0.254	677.34	0.9,0.0,0.1	0.561	676.50	0.3,1.0,-0.03	0.684
B-n34-k5	788	794.33	789.84	1.2,0.0	0.564	789.85	1.2,0.0,0.0	0.564	789.85	1.0,0.3, 0.00	0.564
B-n35-k5	955	978.33	978.32	0.8,0.2	0.001	975.48	1.1,0.1,1.7	0.291	973.27	0.7,0.9,-0.05	0.517
B-n38-k6	805	832.09	824.00	1.4,0.4	0.972	824.00	1.4,0.4,0.0	0.972	820.31	0.5,1.0, 0.03	1.416
B-n39-k5	549	566.71	554.99	1.4,0.3	2.068	555.00	1.4,0.3,0.0	2.066	554.35	1.1,0.0,-0.04	2.181
B-n41-k6	829	898.09	867.42	0.6,0.5	3.415	867.42	0.6,0.4,0.1	3.415	852.95	0.3,0.3,-0.06	5.026
B-n43-k6	742	781.96	754.04	1.4,0.6	3.571	754.92	0.9,0.1,0.4	3.458	756.07	0.7,0.3, 0.03	3.311
B-n44-k7	909	937.74	932.32	1.8,0.8	0.578	934.68	1.9,0.9,1.8	0.326	930.99	0.6,0.8,-0.03	0.720
B-n45-k5	751	757.16	757.16	1.0,0.0	0.000	754.71	1.1,0.0,0.8	0.324	756.60	0.5,0.7,-0.01	0.074
B-n45-k6	678	727.84	713.24	0.9,0.6	2.006	713.24	0.9,0.6,0.0	2.006	717.24	0.2,0.5, 0.10	1.456
B-n50-k7	741	748.80	747.92	1.1,0.0	0.118	745.37	1.0,0.0,0.2	0.458	744.77	0.9,0.3,-0.03	0.538
B-n50-k8	1312	1354.03	1339.44	1.6,0.7	1.078	1338.34	1.9,0.9,0.8	1.159	1337.13	0.9,0.2,-0.05	1.248
B-n51-k7	1032	1059.86	1050.00	1.5,0.0	0.930	1050.00	1.5,0.0,0.0	0.930	1043.58	1.2,0.0, 0.04	1.536
B-n52-k7	747	764.90	763.96	1.1,0.2	0.123	756.90	1.3,0.0,1.5	1.046	762.16	1.0,0.5,-0.06	0.358
B-n56-k7	707	733.74	723.76	0.7,0.1	1.360	722.61	0.8,0.0,0.2	1.517	722.62	0.2,0.2, 0.01	1.516
B-n57-k7	1153	1239.78	1148.97	1.8,0.8	7.325	1148.98	1.1,0.0,0.5	7.324	1150.77	1.1,0.0, 0.05	7.179
B-n57-k9	1598	1653.42	1619.71	0.9,0.0	2.039	1619.72	0.9,0.0,0.0	2.038	1613.27	0.8,0.2, 0.01	2.428
B-n63-k10	1496	1598.18	1562.59	0.9,0.0	2.227	1562.59	0.9,0.0,0.0	2.227	1552.36	0.9,0.1, 0.00	2.867
B-n64-k9	861	921.56	919.37	1.7,0.8	0.238	910.07	1.1,0.8,2.0	1.247	907.30	0.5,0.7, 0.05	1.547
B-n66-k9	1316	1416.42	1372.09	1.4,0.4	3.130	1358.32	1.9,1.1,1.0	4.102	1357.17	0.3,0.8, 0.06	4.183
B-n67-k10	1032	1099.95	1090.18	0.8,0.2	0.888	1070.30	0.8,0.0,1.8	2.696	1066.79	0.7,0.2,-0.09	3.015
B-n68-k9	1272	1317.77	1317.77	1.0,0.0	0.000	1316.07	1.1,0.1,0.4	0.129	1315.76	0.9,0.2,-0.03	0.153
B-n78-k10	1221	1264.56	1263.05	1.0,0.1	0.119	1261.35	1.0,0.1,0.9	0.254	1260.50	1.0,0.1,-0.05	0.321
Average					**1.435**			**1.700**			**1.863**

Table A4. Relative deviations on Christofides and Eilon's test set

Instance	Best	CW	P	*λ,μ*	% Imp	AÖ	*λ,μ,ν*	% Imp	ROBUST	*λ,μ,ν*	% Imp
E-n22-k4	375	388.77	375.28	1.5,0.6	3.470	375.28	1.1,0.9,1.1	3.470	375.28	0.1,1.1, 0.02	3.470
E-n23-k3	569	621.09	573.01	1.7,0.5	7.741	573.01	1.7,0.5,0.0	7.741	573.01	0.2,1.4, 0.01	7.741
E-n30-k4	503	534.45	506.67	1.3,0.3	5.198	506.67	1.3,0.3,0.0	5.198	507.51	0.6,1.0,-0.02	5.041
E-n33-k4	835	843.10	843.09	0.9,0.1	0.001	843.10	0.9,0.1,0.0	0.000	842.83	0.7,0.4, 0.08	0.032
E-n76-k14	1021	1054.60	1052.30	1.1,0.1	0.218	1045.04	1.3,0.0,1.0	0.907	1049.31	0.7,0.3, 0.08	0.502
E-n76-k8	735	794.74	783.12	1.2,0.3	1.462	779.42	1.0,0.5,0.1	1.928	768.05	1.2,0.1, 0.05	3.358
E-n76-k7	682	738.13	718.88	1.7,0.8	2.608	718.88	1.6,0.8,0.2	2.608	716.48	0.3,1.0,-0.04	2.933
E-n101-k14	1071	1139.07	1133.99	0.7,0.5	0.446	1126.39	0.8,0.6,0.3	1.113	1127.01	1.3,0.3,-0.07	1.059
Average					**2.643**			**2.871**			**3.017**

Table A5. Relative deviations on Christofides *et al.*'s test set

Instance	Best	CW	P	*λ,μ*	% Imp	AÖ	*λ,μ,ν*	% Imp	ROBUST	*λ,μ,ν*	% Imp
C50	524.61	584.64	566.10	0.8,0.9	3.171	555.55	1.7,0.2,0.6	4.976	537.29	1.0,1.4,-0.02	8.099
C75	835.26	907.39	866.29	1.0,0.1	4.529	860.21	1.2,0.2,0.7	5.200	864.29	0.5,0.4,-0.06	4.750
C100a	826.14	889.00	865.60	1.5,0.4	2.632	867.35	1.2,0.6,0.1	2.435	854.49	1.6,0.5,-0.05	3.882
C150	1028.42	1140.42	1101.81	2.0,0.7	3.386	1094.06	1.3,0.1,0.3	4.065	1089.78	0.4,0.6,-0.03	4.440
C199	1291.45	1395.74	1370.04	1.4,0.2	1.841	1359.78	1.3,0.0,1.1	2.576	1367.53	1.3,0.2, 0.00	2.021
C120	1042.11	1068.14	1066.40	1.3,0.3	0.163	1057.80	1.1,0.1,0.3	0.968	1059.87	0.9,0.2, 0.02	0.774
C100b	819.56	833.51	826.00	1.2,0.4	0.901	824.66	1.4,0.4,0.6	1.062	825.76	1.1,0.0, 0.03	0.930
Average					**2.375**			**3.040**			**3.557**

Table A6. Relative deviations on Christofides *et al.*'s distance restricted test set

Instance	Best	CW	P	*λ,μ*	% Imp	AÖ	*λ,μ,ν*	% Imp	ROBUST	*λ,μ,ν*	% Imp
CD50	555.43	618.39	595.31	1.3,0.3	3.732	589.43	1.6,0.4,2.0	4.683	582.52	1.0,1.4,-0.08	5.801
CD75	909.63	975.46	942.98	2.0,0.7	3.330	942.98	2.0,0.7,0.0	3.330	944.14	0.9,0.3, 0.07	3.211
CD100a	865.94	973.94	942.69	1.7,0.3	3.209	942.69	0.7,0.3,0.0	3.209	922.77	0.7,1.0,-0.01	5.254
CD150	1162.55	1287.64	1222.10	1.3,0.0	5.090	1222.10	1.3,0.0,0.0	5.090	1216.15	1.3,0.2, 0.06	5.552
CD199	1395.85	1538.66	1485.50	1.9,0.6	3.455	1485.53	1.9,0.6,0.0	3.453	1482.89	1.7,0.3, 0.01	3.625
CD120	1541.14	1592.26	1583.24	0.7,0.1	0.566	1582.20	0.8,0.0,0.2	0.632	1572.81	0.8,0.4,-0.06	1.222
CD100b	866.37	875.75	869.61	1.2,0.2	0.701	869.61	1.2,0.2,0.0	0.701	872.60	0.8,0.4, 0.03	0.360
Average					**2.869**			**3.014**			**3.575**