

Optimized Crossover Genetic Algorithm for Vehicle Routing Problem with Time Windows

H. Nazif and L.S. Lee

Department of Mathematics, Faculty of Science, University Putra Malaysia,
43400 UPM Serdang, Selangor, Malaysia

Abstract: Problem statement: In this study, we considered the application of a genetic algorithm to vehicle routing problem with time windows where a set of vehicles with limits on capacity and travel time are available to service a set of customers with demands and earliest and latest time for serving. The objective is to find routes for the vehicles to service all the customers at a minimal cost without violating the capacity and travel time constraints of the vehicles and the time window constraints set by the customers. **Approach:** We proposed a genetic algorithm using an optimized crossover operator designed by a complete undirected bipartite graph that finds an optimal set of delivery routes satisfying the requirements and giving minimal total cost. Various techniques have also been introduced into the proposed algorithm to further enhance the solutions quality. **Results:** We tested our algorithm with benchmark instances and compared it with some other heuristics in the literature. The results showed that the proposed algorithm is competitive in terms of the quality of the solutions found. **Conclusion/Recommendations:** This study presented a genetic algorithm for solving vehicle routing problem with time windows using an optimized crossover operator. From the results, it can be concluded that the proposed algorithm is competitive when compared with other heuristics in the literature.

Key words: Genetic algorithm, vehicle routing problem, time windows

INTRODUCTION

The Vehicle Routing Problem with Time Windows (VRPTW) which is an extension of Vehicle Routing Problems (VRPs) arises in a wide array of practical decision making problems. Instances of the VRPTW occur in rail distribution, school bus routing, mail and newspaper delivery, airline and railway fleet routing and etc. In general the VRPTW is defined as follows: given a set of identical vehicles $V = \{1, 2, \dots, K\}$, a central depot node, a set of customer nodes $C = \{0, 1, 2, \dots, N\}$ and a directed network connecting the depot and customers. Each arc in the network represents a connection between two nodes and also indicates the direction it travels. The depot is denoted as customer 0, which uses K independent delivery vehicles, with delivery capacity q_k , $k = 1, 2, \dots, K$, to service demands m_i from n customers, $i = 1, 2, \dots, N$. The time window constraint is denoted by a predefined time interval $[a_i, b_i]$, a_i and b_i describe earliest arrival time and latest arrival time for customer i , respectively. The vehicles must arrive at the customers not later than the latest arrival time, if vehicles arrive earlier than the

earliest arrival time, waiting occurs. Each customer also imposes a service time s_i , taking consideration of the loading/unloading time of goods. A non-negative cost (distance or travel time) matrix $C = (c_{ij})$ between customers i and j is defined on the network. A solution for the VRPTW would be a partition R_1, R_2, \dots, R_K , representing the routes of the vehicles, each route R_k is a permutation of the customers in C specifying the order of visiting them, starting and ending at the depot. The cost of the problem solution is the sum of the costs of its routes R_k , defined as follows:

$$\text{Cost} = \sum_{k=1}^K \text{Cost}(R_k) \quad (1)$$

The VRPTW consists in determining a set of a maximum of K routes (i) of minimum total cost (Eq. 1); (ii) starting and ending at the depot denoted with customer 0 and such that (iii) each customer is visited exactly once by exactly one vehicle; subject to the restrictions (iv) the total demand of any route R_k does not exceed q_k ; (v) each route R_k must be completed within a total route time, which is essentially the time

window of the depot; (vi) the vehicles must visit the customers within predefined time windows.

A survey of the VRPTW is given by Desrosiers *et al.* (1995); Cordeau *et al.* (2002) and Bräysy and Gendreau (2005). Significant improvements in Solomon's benchmark problem instances were established by Rochat and Taillard (1995) using a tabu search metaheuristic method. Some studies on tabu searches for VRPTW can be found in Taillard *et al.* (1997); Chiang and Russell (1997) and Cordeau *et al.* (2001). Gambardella *et al.* (1999) proposed an ant colony optimisation while, Liu and Shen (1999) applied a route-neighborhood-based metaheuristic to solve VRPTW. Rousseau *et al.* (1999) used constraint-based operators with variable neighborhood search to solve the problem.

Blanton and Wainwright (1993) were the first to apply a Genetic Algorithm (GA) to VRPTW. They hybridized a GA with a greedy heuristic. GAs have been proposed for the problem by Thangiah (1995); Potvin and Bengio (1996); Berger *et al.* (1998; 1999); Berger and Barkaoui (2000; 2004); Bräysy *et al.* (2000) and Tan *et al.* (2001a). Gehring and Homberger (2002) proposed a parallelization of a two-phase metaheuristic for solving VRPTW.

In this study, we consider the vehicle routing problem with time windows and propose an Optimized Crossover Genetic Algorithm (OCGA) for this problem.

MATERIALS AND METHODS

Optimized Crossover Genetic Algorithm (OCGA): GA, originally developed by Holland and Holland (1975) is an adaptive heuristic search method that mimics evolution through natural selection. It works by combining selection, crossover and mutation operations. The selection pressure drives the population toward better solutions while crossover uses genes of selected parents to produce offspring that will form the next generation. Mutation is used to escape from local minima.

Optimized crossover was proposed by Aggarwal *et al.* (1997) for the independent set problem. They applied optimized crossover within genetic algorithm and produced two new children: The Optimum child (O-child) and Exploratory child (E-child). The O-child is constructed in such a way that has the best objective function value from a feasible set of children, while the E-child is constructed so as to maintain the diversity of the search space.

In the remainder, we describe the proposed optimized crossover and discuss how various steps of

the GA are implemented for the proposed algorithm. The general framework of OCGA can be shown as follows:

Algorithm OCGA:

begin

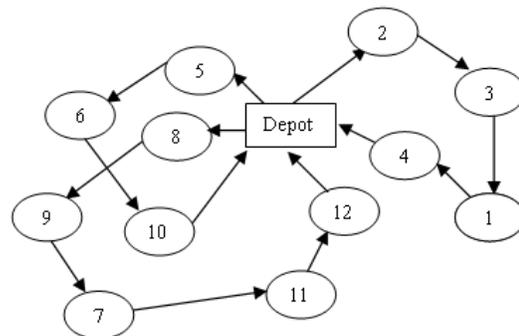
```

Initialize population (randomly generated);
Fitness evaluation;
repeat
    Selection (probabilistic binary tournament selection);
    Optimized crossover;
        Swap node (if the optimized crossover is not applied);
    Mutation (inversion and swap sequence);
    Fitness evaluation;
    Elitism replacement with filtration;
until the end condition is satisfied;
return the fittest solution found;
    
```

end

Individual representation and selection mechanism:

The representation of a solution we use here is an integer string of length N, where N is the number of customers, applied by Tan *et al.* (2001b). Each gene in the string is the integer node number assigned to that customer originally. The sequence of the genes in the string is the order of visiting these customers. For example, if we have the following solution represented in Fig. 1.



```

Route No. 1: 0 → 2 → 3 → 1 → 4 → 0
Route No. 2: 0 → 5 → 6 → 10 → 0
Route No. 3: 0 → 8 → 9 → 7 → 11 → 12 → 0
    
```

The coded integer string is then:

```
2 → 3 → 1 → 4 → 5 → 6 → 10 → 8 → 9 → 7 → 11 → 12
```

Fig. 1: A solution to a vehicle routing problem

This representation is unique and one string can only be decoded to one solution. It is a 1-to-1 relation. The last customer visited in route i is linked with the first customer visited in route $i+1$ to form a string of all the routes involved. Note that we do not put any bit in the string to indicate the end of a route now, because such delimiters in a string greatly restrain the validity of children produced by optimized crossover operator later. To decode the string into route configurations, the gene values are inserted into the new routes sequentially.

There is a chance that we may not get back the original routes after decoding, but it is generally assumed that minimizing the number of routes helps in minimizing the total travel cost, therefore, packing a route to its maximum capability implies a potential good solution as a result (Tan *et al.*, 2001b).

After choosing the representation, we uniformly randomly generate an initial population using a random number generator. Moreover we use a probabilistic binary tournament selection scheme to select individuals from the population to be the parents for the OCGA with a given selection probability $p_s = 0.75$. In other words, we give a 75% chance for the fitter individual to be selected as the parent compared to the less fit individual which only has a 25% chance to be selected.

Optimized crossover: We propose an optimized crossover operator within a GA for the VRPTW. The proposed optimized crossover using a complete undirected bipartite graph finds the two new children which are called O-child and E-child. We will now explain the optimized crossover strategy on determining the O-child and E-child for the VRPTW.

Step 1: Given two parents P_1 and P_2 . Construct a complete undirected bipartite graph $G = (U \cup V, E_1 \cup E_2)$ where $U = \{u_1, u_2, \dots, u_n\}$ representing customers, $V = \{v_1, v_2, \dots, v_n\}$ representing nodes, E_1 and E_2 representing the arc sets in the graph in which, $\{u_j, v_i\} \in E_1$ if and only if, customers j of parent P_1 is located at node i and $\{u_j, v_i\} \in E_2$ if and only if, customers j of parent P_2 is located at node i .

Step 2: Determine all perfect matchings in graph G . Suppose that there are k cycles corresponding to different locations of customers in the two parents. There will be exactly 2^k perfect matchings in graph G . Note that each perfect matching representing a temporary offspring.

Step 3: Select a temporary offspring with the least objective function value as O-child.

Step 4: Generate E-child using $((P_1 \cup P_2) \setminus O\text{-child}) \cup (P_1 \cap P_2)$.

We restricted the temporary offspring in graph G in every case to 2^5 even if the cycles are more than 5. In addition, the proposed optimized crossover is applied based on a crossover probability, $p_c = 0.75$.

Swap node: In this study, we use a swap node operator instead of reproduction to produce two new offspring when the optimized crossover is not applied to the parents. This operator could be regarded as a giant mutation where the elements in the parent are randomly reassigned. The process of swap node operator is as follows: two nodes from a parent are randomly selected and swapped. It is repeated for the second parent to create a second offspring. An example of swap node operator is given in Fig. 2.

Mutation: The mutation operator used in the proposed algorithm consists of applying with equal probability two different mutation operators called inversion and swap sequence operators. The inversion operator reverses the visiting order of the customers between two randomly selected points, while the swap sequence operator consists of randomly selecting two sub-strings of customers and exchanging them. The examples of the two mutation operators are given in Fig. 3.

Elitism replacement with filtration: Elitism replacement scheme is applied in this study as follows: Both parent and offspring population are combined into a single population and sorted in a non-increasing order of their associated fitness value. Then, the first half of the combined population is selected as the individuals of the new population for the next generation. In order to avoid premature convergence and to add diversity to the new population, we propose the filtration strategy in which, identical individuals are identified from the new population and, they are removed and replaced by uniformly randomly generated new individuals. As the filtration strategy requires a certain amount of computational time, the procedure will only be invoked after every 50 generations.

Parent 1	5 2 1 3 6 4 7	Parent 2	1 5 2 7 6 3 4
Offspring 1	5 2 4 3 6 1 7	Offspring 2	1 7 2 5 6 3 4

Fig. 2: An example of swap node

Offspring (before)	Inversion 7 2 5 4 6 1 3	Offspring (before)	Swap sequence 7 2 5 4 6 1 3
Offspring (after)	7 2 1 6 4 5 3	Offspring (after)	7 4 6 1 5 2 3

Fig. 3: Examples of the two mutation operators

RESULTS

We present the results of the computational experiments of our proposed algorithm on VRPTW. The proposed algorithm is coded in C language and implemented on a Pentium 4, 2.0 GHz computer with 2.0 GB RAM.

The OCGA has been tested on all six classes of the benchmark of Solomon (1987) which are 100-customer problem sets. The classes R1 and R2 have customers randomly disposed, while the classes C1 and C2 have customers clustered. The classes RC1 and RC2 contain a subset of customers randomly disposed and the other part clustered. The distance between two customers is the simple Euclidean distance. One unit of time is necessary to run one unit of distance by one vehicle. Each customer *i* has a time window [*a_i*, *b_i*], which represents the time interval to arrive in that customer. Different capacity constraints are considered for the vehicle in each class of instance, as well as the demands from the customers.

We compare our proposed algorithm against the best known solutions reported in the literature, although Table 1: Computational results for the benchmark of Solomon (1987)

Best-known solution				OCGA				Best-known solution				OCGA			
Problems	Ref.	NV	TD	NV	TD	Problems	Ref.	NV	TD	NV	TD	NV	TD		
R101	Czarnas <i>et al.</i> (2004)	19	1650.80	20	1667.40	R201	Gehring and Homberger (2002)	4	1252.37	8	1248.87				
R102	Rochat and Taillard (1995)	17	1486.12	18	1480.73	R202	Rochat and Taillard (1995)	4	1088.07	6	1079.36				
R103	Rochat and Taillard (1995)	14	1213.62	14	1254.62	R203	Mester (2002)	3	939.54	5	965.79				
R104	Rochat and Taillard (1995)	10	982.01	11	1005.34	R204	Bent and Van Hentenryck (2004)	2	825.52	4	813.90				
R105	Rochat and Taillard (1995)	14	1377.11	16	1372.66	R205	Rousseau <i>et al.</i> (1999)	3	994.42	6	994.88				
R106	Mester (2002)	12	1251.98	13	1263.54	R206	Schrimpf <i>et al.</i> (2000)	3	906.14	5	928.96				
R107	Shaw (1997)	10	1104.66	11	1095.11	R207	Rochat and Taillard (1995)	3	814.78	4	847.54				
R108	Berger and Barkaoui (2004)	9	960.88	10	972.13	R208	Mester (2002)	2	726.75	4	725.42				
R109	Gehring and Homberger (2002)	11	1194.73	13	1201.51	R209	Czarnas <i>et al.</i> (2004)	3	909.16	6	890.27				
R110	Rochat and Taillard (1995)	11	1080.36	12	1119.05	R210	Mester (2002)	3	939.34	6	946.55				
R111	Rousseau <i>et al.</i> (1999)	10	1096.72	12	1088.48	R211	Bent and Van Hentenryck (2004)	2	892.71	5	888.73				
R112	Rochat and Taillard (1995)	10	953.63	11	994.22										
C101	Rochat and Taillard (1995)	10	828.94	10	828.94	C201	Rochat and Taillard (1995)	3	591.56	3	591.56				
C102	Rochat and Taillard (1995)	10	828.94	10	828.94	C202	Rochat and Taillard (1995)	3	591.56	3	591.56				
C103	Rochat and Taillard (1995)	10	828.06	10	828.15	C203	Rochat and Taillard (1995)	3	591.17	3	591.17				
C104	Rochat and Taillard (1995)	10	824.78	10	825.19	C204	Rochat and Taillard (1995)	3	590.60	3	590.60				
C105	Rochat and Taillard (1995)	10	828.94	10	828.94	C205	Rochat and Taillard (1995)	3	588.88	3	588.88				
C106	Rochat and Taillard (1995)	10	828.94	10	828.94	C206	Rochat and Taillard (1995)	3	588.49	3	588.49				
C107	Rochat and Taillard (1995)	10	828.94	10	828.94	C207	Rochat and Taillard (1995)	3	588.29	3	588.29				
C108	Rochat and Taillard (1995)	10	828.94	10	828.94	C208	Rochat and Taillard (1995)	3	588.32	3	588.32				
C109	Rochat and Taillard (1995)	10	828.94	10	828.94										
RC101	Rochat and Taillard (1995)	15	1623.58	17	1652.09	RC201	Mester (2002)	4	1406.91	8	1387.55				
RC102	Rochat and Taillard (1995)	13	1477.54	14	1496.23	RC202	Rochat and Taillard (1995)	4	1165.57	7	1176.98				
RC103	Shaw (1998)	11	1261.67	11	1306.12	RC203	Czech and Czarnas (2002)	3	1049.62	5	1082.94				
RC104	Cordeau <i>et al.</i> (2001)	10	1135.48	10	1177.63	RC204	Mester (2002)	3	798.41	4	836.13				
RC105	Berger and Barkaoui (2004)	13	1629.44	16	1618.55	RC205	Mester (2002)	4	1297.19	7	1270.69				
RC106	Rochat and Taillard (1995)	12	1384.92	13	1382.27	RC206	Czarnas <i>et al.</i> (2004)	3	1146.32	6	1149.73				

few works have considered Solomon's benchmark problems minimizing Travel Distance (TD).

DISCUSSION

The results are presented in Table 1 together with the best known solution in the literature. The numerical results are computed after making 50 independent runs for statistical significance. The objective function applied is only the total TD. Results in Table 1 represent the best known solution is found or improved (better TD) by OCGA in 32 problems. It is clear to see that the proposed algorithm continues to be competitive in terms of total TD.

Following, Table 2 shows the results by problem class, including results from other relevant works in the literature. We have selected the heuristic methods proposed by Taillard *et al.* (1997); Rousseau *et al.* (1999); Gehring and Homberger (2002); Tan *et al.* (2001b) and Berger and Barkaoui (2004). We compute the percentage Deviation (D) in terms of travel distance between our best solution (sol) and the published best (best) for each problem class with the following equation:

$$D = \left[\frac{|\text{best} - \text{sol}|}{\text{best}} \right] \times 100\% \tag{2}$$

RC107	Shaw (1997)	11	1230.48	13	1243.20	RC207	Bent and Van Hentenryck (2004)	3	1061.14	6	1053.58
RC108	Taillard <i>et al.</i> (1997)	10	1139.82	11	1128.46	RC208	Ibaraki <i>et al.</i> (2001)	3	828.14	5	816.10

NV: Number of Vehicles; TD: Travel Distance

Table 2: Computational results between VRPTW algorithms

		R1	C1	RC1	R2	C2	RC2
Published best	NV	12.250	10.000	11.870	2.910	3.000	3.370
	TD	1196.050	828.380	1360.370	935.350	589.860	1094.160
Taillard <i>et al.</i> (1997)	NV	12.170	10.000	11.500	2.820	3.000	3.380
	TD	1209.350	828.380	1389.220	980.270	589.860	1117.440
Rousseau <i>et al.</i> (1999)	NV	12.830	10.000	12.500	3.180	3.000	3.750
	TD	1201.100	828.380	1370.260	966.940	594.010	1113.290
Tan <i>et al.</i> (2001b)	NV	13.830	10.110	13.630	5.640	3.250	7.000
	TD	1260.710	859.810	1447.060	1058.520	617.100	1169.410
Gehring and Homberger (2002)	NV	12.000	10.000	11.630	2.730	3.000	3.250
	TD	1226.380	828.380	1392.570	969.950	589.860	1144.430
Berger and Barkaoui (2004)	NV	11.920	10.000	11.500	2.730	3.000	3.250
	TD	1221.100	828.480	1389.890	975.430	589.930	1159.370
OCGA	NV	13.420	10.000	13.120	5.360	3.000	6.000
	TD	1209.560	828.430	1375.570	939.110	589.860	1096.710
	D%	1.165	0.006	1.117	0.402	0.000	0.233

CONCLUSION

This study presents a genetic algorithm that uses an optimized crossover operator to solve the vehicle routing problem with time windows. The proposed algorithm has been tested against the best known solutions reported in the literature, using 56 Solomon's problems with 100 customers. The computational results showed that the proposed algorithm is competitive in terms of the quality of the solutions found (in terms of total travel distance). As for future work, it may be interesting to test OCGA with additional benchmarks of VRPTW.

REFERENCES

Aggarwal, C.C., J.B. Orlin and R.P. Tai, 1997. Optimized crossover for the independent set problem. *Operat. Res.*, 45: 226-234. DOI: 10.1287/opre.45.2.226

Bent, R. and P. Van Hentenryck, 2004. A two-stage hybrid local search for the vehicle routing problem with time windows. *Transport. Sci.*, 38: 515-530. DOI: 10.1287/trsc.1030.0049

Berger, J. and M. Barkaoui, 2000. An improved hybrid genetic algorithm for the vehicle routing problem with time windows. *Proceeding of the International ICSC Symposium on Computational Intelligence, Part of the International ICSC Congress on Intelligent Systems and Applications*, Dec. 11-15, University of Wollongong, Wollongong, Australia. <http://www.proceedings.com/00444.html>

Berger, J. and M. Barkaoui, 2004. A parallel hybrid genetic algorithm for the vehicle routing problem with time windows. *Comput. Operat. Res.*, 31: 2037-2053. DOI: 10.1016/S0305-0548(03)00163-1

Berger, J., M. Salois and R. Begin, 1998. A hybrid genetic algorithm for the vehicle routing problem with time windows. *Lecture Notes Artif. Intell.*, 1418: 114-127. DOI: 10.1007/3-540-64575-6_44

Berger, J., M. Sassi and M. Salois, 1999. A hybrid genetic algorithm for the vehicle routing problem with time windows and itinerary constraints. *Proceedings of the Conference on Genetic and Evolutionary Computation*, July 13-17, Orlando, USA., pp: 44-51. <http://www.alibris.com/search/books/isbn/1558606114>

Blanton, J.L. and R.L. Wainwright, 1993. Multiple vehicle routing with time and capacity constraints using genetic algorithms. *Proceedings of the 5th International Conference on Genetic Algorithms, (ICGA'93)*, Morgan Kaufmann, San Francisco, pp: 452-459.

Bräysy, O. and M. Gendreau, 2005. Vehicle routing problem with time windows, Part II: Metaheuristics. *Transport. Sci.*, 39: 119-139. DOI: 10.1287/trsc.1030.0057

Bräysy, O., J. Berger and M. Barkaoui, 2000. A new hybrid evolutionary algorithm for the vehicle routing problem with time windows. *Proceedings of the International Workshop on Vehicle Routing and Related Areas*, Aug. 16-19, Skodsborg, Denmark. http://74.125.155.132/scholar?q=cache:0510XxcO08UJ:scholar.google.com/+Berger,+J.+and+M.+Barkaoui,+2000&hl=en&as_sdt=2000

Chiang, W.C. and R.A. Russell, 1997. A reactive tabu search metaheuristic for the vehicle routing problem with time windows. *Inform. J. Comput.*, 9: 417-430. DOI: 10.1287/ijoc.9.4.417

- Cordeau, J.F., G. Desaulniers, J. Desrosiers, M.M. Solomon and F. Soumis, 2002. The VRP with Time Windows. In: *The Vehicle Routing Problems*, SIAM Monographs on Discrete Mathematics and Applications, Society for Industrial and Applied Mathematics, Toth, P. and D. Vigo (Eds.). Philadelphia, USA., ISBN: 0-89871-498-2, pp: 157-193.
- Cordeau, J.F., G. Laporte and A. Mercier, 2001. A unified tabu search heuristic for vehicle routing problem with time windows. *J. Operat. Res. Soc.*, 52: 928-936. DOI: 10.1057/palgrave.jors.2601163
- Czarnas, P., Z.J. Czech and P. Gocyla, 2004. Parallel simulated annealing for bicriterion optimization problems. *Lecture Notes Comput. Sci.*, 3019: 233-240. DOI: 10.1007/b97218
- Czech, Z.J. and P. Czarnas, 2002. Parallel simulated annealing for the vehicle routing problem with time windows. *Proceedings of the 10th Euromicro Workshop on Parallel, Distributed and Network-Based Processing*, Jan. 9-11, Canary Island, Spain, pp: 376-383.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.100.8805&rep=rep1&type=pdf>
- Desrosiers, J., Y. Dumas, M.M. Solomon and F. Soumis, 1995. Time Constrained Routing and Scheduling. In: *Handbooks in Operations Research and Management Science, Network Routing*, Ball, M.O., T.L. Magnanti, C.L. Monma and G.L. Nemhauser (Eds.). Amsterdam, North-Holland, ISBN: 10: 0444821414, pp: 35-139.
- Gambardella, L.M., É. Taillard, and G. Agazzi, 1999. MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows. In: *New Ideas in Optimization*, Corne, D., M. Dorigo and F. Glover (Eds.). McGraw-Hill, London, ISBN: 0-07-709506-5, pp: 63-76.
- Gehring, H. and J. Homberger, 2002. Parallelization of a two-phase metaheuristic for routing problems with time windows. *J. Heurist.*, 8: 251-276. DOI: 10.1023/A:1015053600842
- Holland, J.H. and J.H. Holland, 1975. *Adaptations in Natural and Artificial Systems: an introductory analysis with applications to biology, control, and artificial intelligence*. The University of Michigan Press, Ann Arbor, ISBN: 0-262-58111-6, pp: 183.
- Ibaraki, T., M. Kubo, T. Masuda, T. Uno and M. Yagiura, 2001. Effective local search algorithms for the vehicle routing problem with general time windows Constraints. *Proceedings of the 4th international Conference*, July 16-20, Porto, Portugal, pp: 293-297.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.6.3356&rep=rep1&type=pdf>
- Liu, F.H. and S.Y. Shen, 1999. A route-neighborhood-based metaheuristic for vehicle routing problem with time windows. *Eur. J. Operat. Res.*, 118: 485-504. DOI: 10.1016/S0377-2217(98)00315-4
- Mester, D., 2002. An Evolutionary strategies algorithm for large scale vehicle routing problem with capacitate and time windows restrictions. Working Paper, Institute of Evolution, University of Haifa, Israel. <http://www.top.sintef.no/vrp/bknown8.html>
- Potvin, J.Y. and S. Bengio, 1996. The vehicle routing problem with time windows, Part II: Genetic search. *Inform. J. Comput.*, 8: 165-172. DOI: 10.1287/ijoc.8.2.165
- Rochat, Y. and É.D. Taillard, 1995. Probabilistic diversification and intensification in local search for vehicle routing. *J. Heurist.*, 1: 147-167. DOI: 10.1007/BF02430370
- Rousseau, L.M., M. Gendreau and G. Pesant, 1999. Using constraint-based operators with variable neighborhood search to solve the vehicle routing problem with time windows. *Proceedings of the 1st Workshop on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, Feb. 1999, University of Ferrara, Italy.
<http://citeseer.ist.psu.edu/old/406681.html>
- Schrimpf, G., J. Schneider, H. Stamm Wilbrandt and G. Dueck, 2000. Record breaking optimization results using the ruin and recreate principle. *J. Comput. Phys.*, 159: 139-171. DOI: 10.1006/JCPH.1999.6413
- Shaw, P., 1997. A new local search algorithm providing high quality solutions to vehicle routing problems. Working Paper, University of Strathclyde, Glasgow, Scotland.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.51.1273&rep=rep1&type=pdf>
- Shaw, P., 1998. Using constraint programming and local search methods to solve vehicle routing problems. *Proceedings of the 4th International Conference on Principles and Practice of Constraint Programming*, Oct. 26-30, Springer-Verlag, London, UK., pp. 417-431.
<http://portal.acm.org/citation.cfm?id=726320>
- Solomon, M.M., 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operat. Res.*, 35: 254-265. DOI: 10.1287/opre.35.2.254
- Taillard, É., P. Badeau, M. Gendreau, F. Guertin and J.Y. Potvin, 1997. A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportat. Sci.*, 31: 170-186. DOI: 10.1287/trsc.31.2.170

- Tan, K.C., L.H. Lee and K. Ou, 2001a. Hybrid genetic algorithms in solving vehicle routing problems with time window constraints. *Asia-Pacif. J. Operat. Res.*, 18: 121-130. <http://direct.bl.uk/bld/PlaceOrder.do?UIN=097067636&ETOC=RN&from=searchengine>
- Tan, K.C., L.H. Lee, Q.L. Zhu and K. Ou, 2001b. Heuristic method for vehicle routing problem with time windows. *Artifi. Intel. Eng.*, 15: 281-295. DOI: 10.1016/S0954-1810(01)00005-X
- Thangiah, S., 1995. Vehicle Routing with Time Windows Using Genetic Algorithms. In: *Application Handbook of Genetic Algorithms: New Frontiers*, Chambers, L. (Ed.). CRC Press, Boca Raton, ISBN: 0849325196, pp: 253-277.