Memetic Algorithm Based on a Constraint Satisfaction Technique for VRPTW

Marco A. Cruz-Chávez[1], Ocotlán Díaz-Parra[1], David Juárez-Romero[1], and Martín G. Martínez-Rangel[2]

1CIICAp, 2FCAeI, Autonomous University of Morelos State *
Av. Universidad 1001. Col. Chamilpa, C.P. 62110. Cuernavaca, Morelos,Mexico
(mcruz,odiazp,djuarez,mmtzr)@uaem.mx

Abstract. In this paper a Memetic Algorithm (MA) is proposed for solving the Vehicles Routing Problem with Time Windows (VRPTW) multi-objective, using a constraint satisfaction heuristic that allows pruning of the search space to direct a search towards good solutions. An evolutionary heuristic is applied in order to establish the crossover and mutation between sub-routes. The results of MA demonstrate that the use of Constraints Satisfaction Technique permits MA to work more efficiently in the VRPTW.

1 Introduction

One of the first forerunners of genetic algorithms was John Holland in 1960 [1][2].The mere structure of a GA (Genetic Algorithm) involves three types of operators: Selection, crossover and mutation[3][4].

By definition, the search carried out by a GA in the solution space of a problem is global (exploration in the search space). When global search is combined with local search (exploitation in the search space), a genetic hybrid algorithm, called a MA (Memetic Algorithm), is formed [5] This MA, because of the new characteristics contributed by the local search, is able to find better solutions than a simple GA because for each solution S obtained by the global search, the algorithm searches the neighborhood of S for the local optimum, which could turn out to be the global optimum. Researchers suggest that involving the technique of local search in GA, allows for results nearer to the global optimum to be found in combinatorial optimization problems [6][7].

In this paper a Memetic Algorithm is proposed called GA-PCP, which combines two techniques of search, local and global. For the local search, the algorithm used was one of constraint satisfaction PCP (Precedence Constraint Posting) proposed by Cheng and Smith [8]. For the global search, the simple crossover of a GA was used.

In order to prove the efficiency of the proposed algorithm, GA-PCP was applied to the Vehicles Routing Problem well-known like VRPTW (Vehicles Routing Problem with Time Windows) which is an NP-complete problem [9]

* This work was supported by project 160 of the Fideicomiso SEP-UNAM, 2006-2007

[10]. The VRPTW [11] [12] [13] [14] [15] [16] is a variant of the VRP with the additional restriction of the time window associated with each client. This window defines an interval within which the client has to be assisted. The objective is to reduce the number of the vehicles, the route time sum, and the necessary wait time to provide all clients the times of attention required.

Very little research of MA exists as applied to VRPTW. In [17], a Memetic is proposed using a GA for a constraint satisfaction model of VRPTW with rescheduling and optimization of Pareto. The algorithm includes three local searches, Route-exchange, mutation and lambda-exchange. In [18] a Memetic is proposed that combines TS (Tabu Search) and GA. TS is used for its excellent local search execution capacity which allows for exploitation of the solutions space, while GA is able to diversify these local searches, allowing for the exploration of several regions in the search space. In [19], a Memetic multi-objective is proposed that incorporates three heuristics of local exploration. The first heuristic, Intra_Route, generates two different numbers based on the sequence size of the route assignment of both vehicles. This heuristic chooses two routes randomly and exchanges two nodes of each route. The second heuristic, Lambda_Interchange, assumes that two routes A and B are selected, and begins by sweeping the nodes of route A and moving the feasible nodes into B route. The third heuristic, Shortest_pf, is a modification of the shortest path first method, which tries to change the order of the nodes of a particular route and uses the optimization concept of Pareto to solve multi-objective optimization in VRPTW.

In this paper, in order to apply PCP to VRPTW, the problem was treated as a CSP (Constraint Satisfaction Problem). The constraint satisfaction works with problems that have finite domains like VRPTW, which is a discreet optimization problem. A solution to a CSP is an assignment of values to all the variables such that all restrictions of the CSP are satisfied. The most common techniques in CSP management can be organized in three groups: **systematic search techniques, inference techniques and hybrid techniques** [20].

In this work, GA-PCP uses the **hybrid search constraint satisfaction technique** for a CSP using the PCP look-ahead algorithm.

The PCP local search algorithm involves the calculation of the shortest path, partially and globally, between a pair of nodes and among all the nodes respectively, in the graph that represents the VRPTW model [21]. PCP is applied specifically to disjunctive graphs models. PCP fixes the address of each edge based on the execution of certain rules and converts the disjunctive graph into a digraph. The shortest path of the digraph represents a feasible solution to VRPTW. The representation of results that is obtained by PCP is coupled with the model of the VRPTW, which is modeled by means of a digraph in order to represent the routes, clients, demand for the client and times of attention required by the client (time window). PCP carries out a series of transformations in order to establish the address of edges in a graph, the set of transformations that is carried out to change an edge is small, since every time that it returns only a small change the address of an edge is made. PCP behaves similarly to the ramification of a tree and a bounded solution space, which carries out a local search, where each transformation is considered near. These transformations are called local transformations and the method is known as local search [22].

The result obtained in this research is that the combination of PCP with GA applied to VRPTW improves the results for several benchmarks, depending on the percentage of PCP applied to the population used in GA.

The structure of the paper is as follows; section one is the introduction, section two explains the procedure of the proposed algorithm GA-PCP for the Vehicles Routing Problem with Time Windows, section three shows the experimentation and comparison of results generated by the GA-PCP algorithm compared with the results obtained by others GA that use constraints satisfaction techniques for the Solomon benchmarks, section four presents conclusions.

2 GA-PCP algorithm for VRPTW

Figure 1 is a general outline of the proposed algorithm called GA-PCP (Genetic Algorithm with Precedence Constraint Posting) for VRPTW, the algorithm consists of the following general steps:

Step 1. Creation of the initial population comprised of individuals with route information and individuals with Time windows information.

Step 2. Apply the tournament selection to the initial population.

Step 3. Apply crossover k

Step 4. Apply flip bit mutation.

Step 5. Construction of the following generation. With migrant quality individuals (with PCP) and individuals of the original population.

Step 6. Evaluate the fitness with objective function that is shown in equation (1), for the case of the VRPTW problem, two primordial objectives are used: the demand and attention time to each client, trying to minimize the cost implied by these two objectives.

$$\min \sum_{k \in k} \sum_{(ij) \in A} c_{ij} X_{ijk} \tag{1}$$

In equation (1), c represents the cost of transporting of an i origin to a j destination, X represents the journey of an i origin to a j destination in a k vehicle. In order to complete the cost objective, the minimum number of vehicles assigned to each journey is searched for, while fulfilling the capacity constraint of the vehicle and the time window. For the journey, the attempt is to find the shortest distance.

Step 7. Verify whether the stop criterion is satisfied. The stop criterion is set based on the execution time, the global optimum and the generation number. If some of these criterions are satisfied, the GA-PCP execution is finished.

In order to create the next generation of the population, a certain percentage x of the population generated by the genetic phase of the GA is taken, and a certain percentage y of another migrant population generated with the PCP is taken [21]. The sum of the (x, y) percentage is 100 of the new generation to be evaluated. The creation procedure of the next generation of the population (x, y) is shown in Fig.2.

There are different types of genetic operators applied in the procedure of generation of x population for the genetic phase. One is the tournament selection method operator [13]. The crossover [13] consists of finding two points randomly in a first individual and looking for the corresponding genes to make the crossover in a second individual. This guarantees the fulfillment of one of the restrictions of the VRPTW which is not passing twice through the same node. For the mutation [13], the Flip-Bit method is used which consists of taking two genes (gen1, gen2) randomly from the same individual, with gen1 being different than gen2, and proceeding to exchange the places of gen1 and gen2.

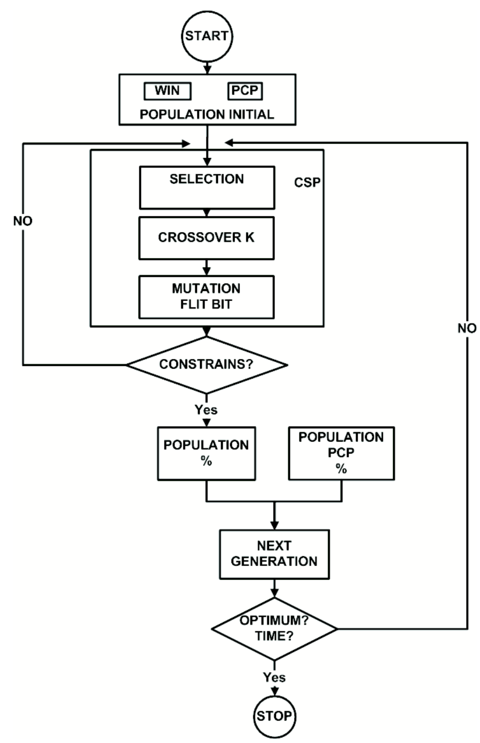

Fig. 1. Flow Diagram of GA-PCP algorithm

Within the search procedure, PCP builds the solution through Depth First using partial assignments of Ω (the set of pair of nodes i, j of the VRPTW disjunctive graph). The PCP algorithm carries out a pruning of the search space early on and provides a heuristic for the assignment of values of the $Ordering_{ij}$ variables.

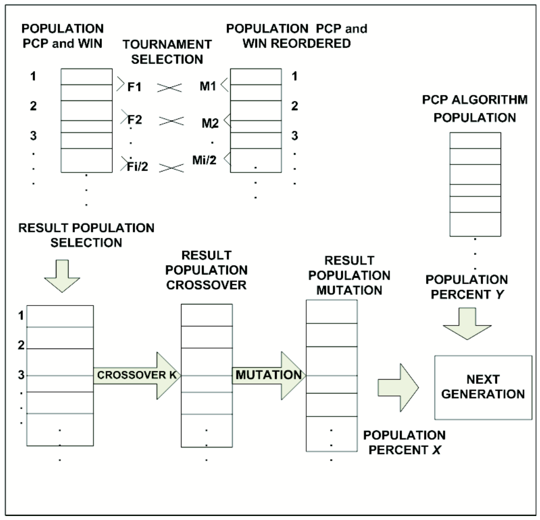

Fig. 2. Realization of the following generation of individuals in GA-PCP

PCP consists of a series of cases in which it should be true that if the shortest path *sp* between a pair of nodes (i, j) that represent the $Ordering_{ij}$ variable then it has a value that fulfills some of the PCP cases. According to the result obtained upon evaluating the shortest path, the value of $Ordering_{ij}$ is designated. The evaluation of *sp* is calculated from i to j (sp_{ij}) and from j to i (sp_{ji}).

The PCP algorithm applies the disjunctive graph model of VRPTW. PCP obtains a digraph as a result, and with the help of a greedy algorithm, the number of optimum routes is obtained that satisfies the capacity restriction of each vehicle used in optimum form that represents a feasible solution to the problem.

The PCP-VRPTW algorithm applied to VRPTW, combined with the search procedure PCP for the CSP consists of the following four steps:

Step 1.- Find the shortest path for each unordered pair of nodes sp_{ij} and sp_{ji}.

Step 2.- Classify the decision of ordination of the pairs not ordered with four cases

Case 1. If $sp_{ij} >= 0$ and $sp_{ji} < 0$ then $O_i \prec O_j$ should be selected.

Case 2. If $sp_{ji} >= 0$ and $sp_{ij} < 0$, then $O_j \prec O_i$ should be selected.

Case 3. If $sp_{ji} < 0$ and $sp_{ij} < 0$, then the partial solution is inconsistent.

Case 4. If $sp_{ji} >= 0$ and $sp_{ij} >= 0$, then no relationship of order is possible

Step 3.- Existence of cases

Does either case 1 or case 2 exist?

If one exists, go to step 4

If neither exists, go to step 1

Step 4.- Fix new precedence for unordered pairs.

The polynomial that defines the complexity in time of the proposed PCP-VRPTW algorithm for the VRPTW as a CSP is$T(n) = c_1n^3 + c_2n^2 + c_3n + c_4$. The complexity of the proposed algorithm is $O(n^3)$, where n is the number of (nodes) clients in the problem.

In order to better understand the algorithm, an example is shown of a small instance of five nodes and a vehicle with a capacity of 200 packages. The disjunctive graph model that is obtained is presented in Fig. 3.

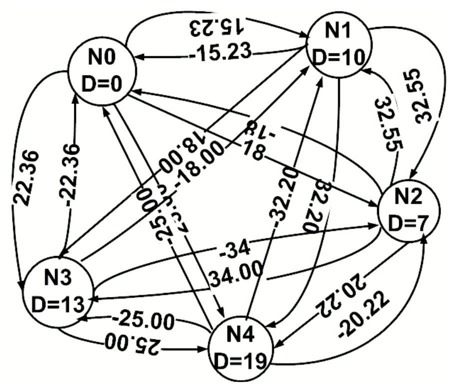

Fig. 3. Disjunctive graph

Applying the shorter path algorithm between pairs of nodes and evaluating the PCP cases, the graph shown in Fig. 4 is obtained. The resulting graph does not generate a feasible solution, this means that a route from the initial node to the final node does not exist through which each node is passed only once.

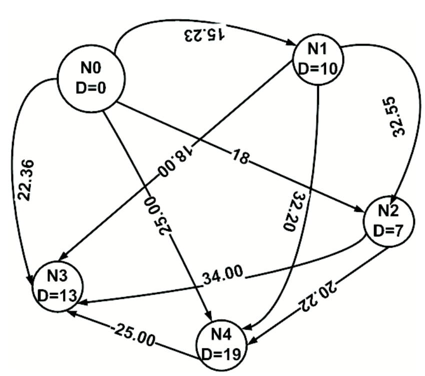

Fig. 4. Conjunctive graph

Because the resulting graph does not generate a solution, backtracking is applied in the nodes with an enter zero and exit zero, leaving fixed the nodes that have at least one entrance and one exit. The PCP algorithm is applied in order to find a route, if a feasible solution is generated, it is taken as a solution. A solution of the problem is shown in Fig. 5. Lastly, a greedy algorithm is used which divides the shortest path presented in Fig. 4 into a set of routes in order to satisfy the demand constraints of the client and capacity of the vehicles.

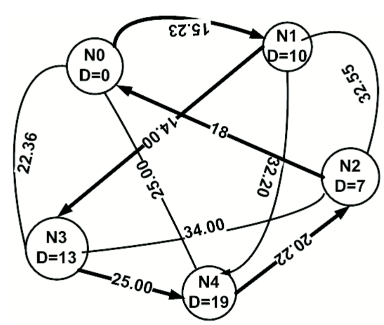

Fig. 5. Solution graph

There are four criteria for stopping the algorithm, it is carried out: (1) for an established amount of time, (2) until the global optimum is found, (3) until within a certain range of an optimum is reached, or (4) for a certain number of generations.

3 Experimental results

The VRPTW problems used in the experiment are taken from the Solomon benchmarks [9]. The instances for VRPTW are classified by type and by class. Two types of instances exist; type 1 manages narrow windows of time and small vehicle capacity, type 2 manages large windows of time and large vehicle capacity. Three classifications exist, C, R and RC. The C classification includes the instances that have a territorial distribution for clients bunched together. The R classification has the clients evenly distributed in a territorial area. The RC classification is the combination of territorial bunched together and distributed distribution is. The Solomon benchmarks for VRPTW used in this experimentation are types C1, R1, RC1, C2, R2 and RC2.

The proposed algorithm GA-CSP is compared with others GA that use CSP. The results that are reported were obtained in a computer with the following characteristic: Pentium processor (R) M to 1.60 GHz, 1GB RAM, operating system XP Windows, and compiler visual C+ 6.0.

The instances used in the experiment were C104, R104, RC108, C204, R208, RC208, for 25 nodes.

Table 1 presents the results obtained with the GA-PCP algorithm. Ten executions were carried out for each benchmark; the reported results include the results of the executions, the best, and medium values as well as the standard deviation. The time of each execution was one hour. Table 1 shows that for the problems of 25 nodes, the best result is near the global optimum, for C104-25 and RC108-25 it was reached with regard to the distance, but for C204-25 the relative error was large. The results for 25 nodes show that GA-PCP works acceptably if the problem is R and/or C classification, when the clients are evenly distributed or bunched together respectively in a territorial area, and when the time windows are narrow and vehicles with a small capacity (type 1) are used, see the results for C104-25, R104-25 and RC108-25. When the problem is of R and C classification but has a big time window and the vehicles have a large capacity (type 2), the results begin decrease in quality, see the result for R208-25 and RC208-25. When the problem only has the C property but has a large time window and the vehicles have a large capacity (type 2), the results are very poor, see the result for C204-25.

Table 1. Results of the GA-PCP algorithm

Benchmark	V	Best	Average	σ	Op*	V*	RE
C104-25	2	186.9	189.8	4.05	1869	3	0
R104-25	3	436.3	467.8	21.48	416.9	4	4.60
RC108-25	5	294.7	300.4	15.03	294.5	3	0.08
C204-25	2	286.0	290.9	4.51	213.1	1	34.25
R208-25	2	329.1	332.0	3.01	328.2	1	0.30
RC208-25	2	271.8	285.9	10.17	269.1	2	1.01

The global optimum in RC108-25 is obtained with $y = 60\%$ for the population PCP of individuals in the GA-PCP. The experiments showing with when y increases from 0 to 60%, GA-PCP tends to improve the solution of RC108-25, also when y increases from 60 to 100%, GA-PCP tends to worsen the solution of RC108-25. For C204-25, with $y = 95\%$ is obtained a value near the global optimum. The experiments showing with when y increases from 0 to 100%, GA-PCP tends to improve the solution of RC204-25.

According to these the results for each instance proven in this paper, the appropriate percentage of the PCP population required in order to improve the efficiency of GA-PCP will be different and will need tuning according to the properties and type of the problem.

The following is a comparison of the results of the GA-PCP algorithm with other genetic algorithms that uses the constraints satisfaction techniques. The heuristics laboratory [23] implemented the GA used for the comparison. These comparison algorithms are the GGA (Generic Genetic Algorithm) [24], SSGA (Steady-State Genetic Algorithm) [25] and SXGA (Sexual Genetic Algorithm) [26]. The GGA algorithm uses the constraints satisfaction technique of systematic search, the SSGA algorithm uses the constraints satisfaction technique of. The SXGA algorithm uses the constraints satisfaction technique of systematic search. These genetic algorithms of the heuristics laboratory report their best results using the following tuning of their entry variables: overload penalty = 50.00, Tardiness penalty = 20.00, Route time penalty = 0.05, Travel time excess penalty = 50.00, Distance penalty = 1.00. The selection operator was tournament. The generations number and population size is the same as used for GA-PCP, 1000 and 100 respectively. With this tuning the GGA algorithms, SSGA and SXGA were executed, giving the results presented in Table 2 and Table 3.

Table 2. Comparative results of efficiency of GA-CSP vs. other algorithms that apply constraints satisfaction technique.

	GA-PCP		GGA		SSGA		SXGA		
Benchmark	UB	t, sec	UB	t, sec	UB	t, sec	UB	t, sec	Op
C104-25	186.9	39.2	190.6	73.8	188.8	53.4	190.6	75.0	186.9
R104-25	436.3	45.8	417.9	75.0	417.6	0.7	418.0	75.6	416.9
RC108-25	294.7	44.9	295.4	78.0	294.9	0.6	295.4	78.0	294.5
C204-25	286.0	44.9	223.3	76.8	223.3	0.4	223.3	76.8	213.1
R208-25	329.1	43.1	329.3	75.6	329.3	0.5	329.3	76.2	328.2
RC208-25	271.8	49.9	271.6	82.8	269.5	0.6	272.0	81.6	269.1

The tuning percentage of PCP per population depends on the problem. For the instance C104, $x = 0.1$ and $y = 0.9$. For instances R104 and C204, $x = 0.05$ and $y = 0.95$. For instances RC108, R208 and RC208, $x = 0.2$ and $y = 0.8$.

Table 3 presents the times that correspond to the time of the best solution obtained in 10 tests executed by each algorithm in each instance of VRPTW. Table 3 shows that the efficiency of GA-PCP is better than GGA and SXGA because it obtains better results with regard to the tuning of the entry parameters for each algorithm. It is observed that SSGA is better in efficacy because the times of execution are the shortest.

Table 3 presents results of 10 tests executed by each algorithm in each problem. It shows the best and worst results, the average value, the standard deviation, and the relative error. These results demonstrate that GA-PCP is competitive with these three algorithms that also use the constraints satisfaction technique. One could observe that the proposed algorithm obtains the best results in three of the six problems, that is, for 50% of the revised benchmarks.

Table 3. Comparative results of the efficacy of GA-CSP with other algorithms that apply the costraints satisfaction technique part 1

	Algorithm					Algorithm			
Results	GA-PCP	GGA	SSGA	SXGA	Results	GA-PCP	GGA	SSGA	SXGA
Problem	C104, OPTIMUM=186.9				Problem	C204, OPTIMUM=213.1			
Best*	186.9	190.6	188.8	190.6	Best*	286.0	223.3	223.3	223.3
Worst	228.2	224.4	201.0	195.6	Worst	317.6	223.4	224.7	223.4
Average	189.8	197.8	193.9	192.2	Average	290.9	223.3	223.7	223.3
σ	4.05	14.03	3.93	2.05	σ	4.51	0.05	0.48	0.04
RE*	0.00	1.98	1.02	1.98	RE*	34.25	4.78	4.79	4.78
Problem	R104, OPTIMUM=416.9				Problem	R208, OPTIMUM=328.2			
Best*	436.3	417.9	417.6	418.0	Best*	329.1	329.3	329.3	329.3
Worst	521.0	423.5	436.2	423.5	Worst	490.2	329.3	333.8	331.3
Average	467.8	419.1	422.9	418.5	Average	332.0	329.3	332.9	329.5
σ	21.48	2.32	5.72	1.74	σ	3.01	0.00	1.56	0.6198
RE*	4.60	0.24	0.17	0.25	RE*	0.30	0.34	0.34	0.34
Problem	RC108, OPTIMUM=294.5				Problem	RC208, OPTIMUM=269.1			
Best*	294.7	295.4	294.9	295.4	Best*	271.8	271.6	269.6	272.0
Worst	452.6	295.0	295.7	295.8	Worst	494.4	277.1	288.4	277.1
Average	300.4	295.4	295.7	295.5	Average	285.9	274.8	277.7	274.4
σ	15.03	0.00	0.74	0.11	σ	10.17	2.73	6.26	2.25
RE*	0.08	0.31	0.14	.31	RE*	34.25	4.78	4.79	4.78

4 Conclusions

The results reported in this research indicate that using the PCP local search algorithm in GA improves the results in VRPTW only for problems of type 1 (small window and small vehicle capacity) with C and RC classification.

The initial population is formed of feasible individuals; a randomly selected population is not used. Instead, the initial population is selected in such a way that it consists of feasible individuals that contain route information and time information. For the next generations, a certain percentage of population of PCP is worked with in order to form the total population of the following generation. It was proven that when the population is formed in great part by PCP individuals, the generated results are near the global optimum for problems of type 1. When problems of type 2 are used, it is better not to use PCP in GA.

It is demonstrated that for the revised benchmarks, the GA-PCP proposed algorithm is competitive in efficiency and efficacy to comparison algorithms used in this investigation that also apply the constraints satisfaction technique. The GA-PCP obtains the best results in 50% of the problems with competitive times of execution.

It can be seen through the results of this experiment that applying a greater percentage of PCP population improves the result of the solution of the GA.

References

[1] Mitchell, M.: An Introduction to Genetic Algorithms. Massachusetts Institute of Technology Press. London (1999)

[2] Holland, J.: Adaptation in Natural and Artificial Systems. The University of Michigan. (1975)

[3] Alvarenga, G.B., Mateus, G.R., De Tomi, G.:A genetic and set partition two-phase approach for the vehicle routing problem with time Windows. Computers & Operations Research.Vol.34. Num.6. Elsevier, 1561-1584. (2007)

[4] Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning . Addison Wesley Professional (1989)

[5] Krasnogor, N. Smith, J.:MAFRA a Java Memetic Algorithm Framework. Intelligent Computer System Centre University of the west of England Bristol, United Kingdom. (2000)

[6] Tavakkoli-Moghaddam, R., Saremi A.R., Ziaee M.S.: A memetic algorithm for a vehicle routing problem with backhauls. Applied Mathematics and Computation Vol. 181 Elsevier (2006)1049-1060

[7] Moscato, P.: On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms. Technical Report Caltech Concurrent Computation Program,Report. 826, California Institute of technology,Pasadena, California, USA(1989)

[8] Cheng-Chung,C., Smith,S.F.: A Constraint Satisfaction Approach to Makespan Scheduling. Proceedings of the Third International Conference on Artificial Intelligence PlanningSystems. Edinburgh, Scotland. ISBN 0-929280-97-0.(1996) 45-52

[9] Solomon, M M.: Algorithms for vehicle routing and scheduling problems with time window constraints. Operations Research, 35(2), (1987)

[10] Garey, M.R. and Johnson, D.S.: Computers and intractability, A Guide to the theory of NP-Completeness, W.H. Freeman and Company, New York, USA. (2003)

[11] Toth, P., Vigo, D.: The Vehicle Routing Problem. Monographs on Discrete Mathematics and Applications. SIAM. Philadelphia. (2001)

[12] Thangiah, S.R.: Vehicle Routing with Time Windows using Genetic Algorithms. In L. Chambers, Editor. Application Handbook of Genetic Algorithms: New Frontiers. Vol. 2. CRC Press. (1995) 253-277

[13] Tan, K.C., Lee, L.H., Zhu, Q.L., Ou, K.: Heuristics methods for vehicle routing problem with time windows. Artificial Intelligence in Engineering. Elsevier (2001) 281-295

[14] Zhu, K.Q.: A new Algorithm for VRPTW. Proceedings of the International Conference on Artificial Intelligence IC-AI 2000. Las Vegas. USA (2000)

[15] Prins, C.: A simple and effective evolutionary algorithm for the vehicle routing problem. Computers & Operations Research. Vol.31.Num.12. Elsevier (2004) 1985-2004

[16] Tan, K.C., Lee, L.H., Ou, K.: Artificial intelligence heuristics in solving vehicle routing problems with time windows constraints. Engineering Applications of Artificial Intelligence. Vol. 14. Num. 6. Elsevier (2001) 825-837

[17] Rhalibi, E.A., Kelleher, G.: An approach to dynamic vehicle routing, rescheduling and disruption metrics, IEEE International Conference on Systems, Man and Cybernetics, Vol.4 (2003) 3613-3618

[18] Chin, A., Kit, H., Lim, A.: A new GA approach for the vehicle routing problem, Proceedings 11th IEEE International Conference on Tools with Artificial Intelligence.(1999) 307-310

[19] Tan, K.C., Lee, T.H., Chew, Y.H., Lee, L.H.: IEEE International Conference on Systems, Man and Cybernetics, Vol. 1(2003)361-366
[20] Castillo, L., Borrajo, D., Salido, M.A.: Planning, Scheduling and Constraint Satisfaction: From Theory to Practice (Frontiers in Artificial Intelligence and Applications), IOS Press, ISBN-10: 1586034847,ISBN-13: 978-1586034849. Spain (2005)
[21] Cruz-Chávez M. A., Díaz-Parra O., Hernández J. A., Zavala-Díaz J. C., Martínez-Rangel M. G.: Search Algorithm for the Constraint Satisfaction Problem of VRPTW, Proceeding of CERMA2007, IEEE-Computer Society, 0-7695-2974-7. 25-28 September, México (2007) 336-341
[22] Aho A.V., Hopcroft J.E., Ulllman J.D.: Structure of data and algorithms, Adisson-Wesley Iberoamericana, Nueva Jersey, Nueva York, California,U.S.A. (1988) (Spanish).
[23] Wagner, S., Affenzeller, M.: The HeuristicLab Optimization Environment, Technical Report. Institute of Formal Models and Verification, Johannes Kepler University Linz, Austria (2004)
[24] Affenzeller M.: A Generic Evolutionary Computation Approach Based Upon Genetic Algorithms and Evolution Strategies, Journal of Systems Science, vol. 28, no. 2 (2002) 59-72
[25] Chafekar, D., Xuan, J., Rasheed, K.: Constrained Multi-objective Optimization Using Steady State Genetic Algorithms, Computer Science Departament University of Georgia. Athens, Genetic and Evolutionary Computation Conference, GA 30602 (USA, 2003)
[26] Wagner, S., Affenzeller, M.: SexualGA: Gender-Specifc Selection for Genetic Algorithms, Proceedings of the 9th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI) 2005, vol. 4 (2005)76-81