Journal of Network and Innovative Computing
ISSN 2160-2174 Volume 2 (2014) pp. 149-157
© MIR Labs, www.mirlabs.net/jnic/index.html

Resolving the VRPTW using an improved hybrid genetic algorithm

Sami Mnasri [1] and Fatma Abbes[2]

[1] Université de Toulouse, CNRS-IRIT Laboratory, IRT group,
IUT Blagnac-1 Place Georges Brassens, 31700 Blagnac-Toulouse, France
Sami.Mnasri@fsgf.rnu.tn

[2] Institut Supérieur de Gestion de Tunis, SOIE Laboratory,
41, Rue de la Liberté, Cité Bouchoucha 2000 Le Bardo, Tunis -Tunisie
abbesfatma2009@yahoo.fr

***Abstract*: This paper proposes an approach which is based on a multi objective genetic algorithm to resolve the vehicles routing problem with time windows (VRPTW). The context of this problem is to plan a set of routes to serve heterogeneous demands respecting several constraints (only one depot, vehicles limited capacity, windows of time). We used an approach based on a multi-objective optimization to resolve this problem. The criteria to be optimized are the number of used vehicles and the total required distance. We propose a method of resolution which is based on a hybridization of a genetic algorithm NSGAII (Not dominated Sorting Genetic Algorithm II) and the BCRC algorithm (Best Cost Route Crossover).**

Keywords: Vehicles Routing Problem, VRPTW, Optimization, BCRC, NSGAII.

I. Introduction

Our work concerns the optimization of the logistics of the transport which occupies a strategic importance for the modern companies of transport considering its economic interests. Different research works dealt with these themes using several approaches and optimization methods. We cite as examples: Zidi et al. [1], Raddaoui et al. [2] and Mnasri et al. [3]. We are interested in particular in the resolution of the vehicles routing problem with time windows (VRPTW). Our approach is based on the application of a genetic algorithm (NSGAII). Our contribution will be the hybridization of this algorithm on the crossing level. This approach is articulated around two great parts: The first part is interested in the application of the NSGAII to solve the VRPTW. The second part is related to the hybridization of the NSGAII with the BCRC algorithm. We finish this work by comparing our approach to other already existing approaches using the benchmarks of Solomon [4] on the different classes of the VRPTW.

II. State of the Art

The application of the vehicles routing problems is related to several fields such as the collection and the distribution of goods. More specifically, the vehicles routing problem, according to **Laporte [5]** consist in the search of an optimal set of routes to serve a set of clients which are geographically dispersed and subjected to several constraints. Usually, vehicles have a limited capacity and customers have a known in advance request. Each vehicle starts from a special node called depot, it serves a set of customers and finishes its route in the depot, respecting the constraint indicating that the sum of the requests of the customers does not exceed its own capacity. The vehicles routing problem is a class of the operational research and the combinative optimization problems **Savelsbergh [6]**.

In reality, customers are often characterized not only by their requests, but also, by the periods of service where each customer must be visited during a well defined window of time. Then, the studied problem can be defined as follows: a set of requests must be delivered by one or more vehicles of known capacity respecting a set of constraints. Each client can be served only during a specified time period. The goal is to establish routes in order to optimize one or more criteria.

These criteria can be related to the total traversed distance, the total required time, the number of used vehicles... etc. These problems have a combinatorial aspect and many applications related to the logistics of distribution. To solve the problem described above, we propose to use genetic algorithms.

The vehicle routing problem was introduced for the first time by **Dantzig et al. [7]** under the name of « Truck Dispatching Problem ». Indeed, It deals about the design of the optimal roads using a fleet of vehicles, based in one or more deposits, to serve a set of clients (or cities) that are geographically dispersed and having a limited and known requests.

Because of its widespread use in practice, the VRPTW has been the subject of several research works. Thus, the works of **Desrochers et al. [8], Desrochers et al. [9]** and **Solomon [4]** provides the following classification to solve this problem using different approaches, this classification was also described by **Badeau et al. [10]:**

- Exact algorithms: Works of **Kolen et al. [11]** and those of **Desrochers [12]**.

- Road Construction heuristics: Works of **Solomon [4], Potvin et al. [13]**, and **Russell [14].**
- Road improvement heuristics: Work of **Baker et al. [15], Potvin et al. [13], Solomon [4]**, and **Thompson and Psaraftis [16].**
- Two phases methods which includes the construction method and the improvement method, subject of the works of **Kontoravdis and Bard [17]** and **Russell [14].**
- Metaheuristics: such as the tabu research (works of **Thompson and Psaraftis [16], Taillard and Rochat [18]**), the simulated annealing (works of **Chiang and Russell [19]**), genetic algorithms (works of **Blanton and Wainwright [20], Potvin and Bengio [21],** and **Thangiah [22]**) and hybrid genetic algorithms (**Thangiah et al [23]**).

However, the works of **Bräysy and Gendreau [24]** shows that although this diversity of the resolution methods, genetic algorithms and tabu search remains the methods giving the most competitive results. According to them, the most important research works treating the VRPTW using genetic algorithms are:

Table 1. Research works treating the VRPTW based on genetic algorithms

Author	**Initial Population**	**Crossover**	**Mutation**
Homberger and Gehring [25]	Stochastic savings heuristic	Uniform order-based to create sequence for controlling Or-opt	Or-opt, 2-opt, λ-interchanges, special Or-opt for route elimination
Tan et al. [26]	Solomon's insertion, λ-interchange, random	PMX	Random swap of nodes
Tan et al. [27]	Random ordering	One-point	-
Le Bouthillier and Crainic [28]	Construction heuristics combined with 2-opt, 3-opt, and Or-opt	Order (OX) and edge recombination (ER)	2-opt, 3-opt, and Or-opt
Jung and Moon [29]	Solomon's insertion	Selecting arcs based on 20 image of a solution and nearest neighbor rule	2-opt, Or-opt, relocation and splitting of routes

III. Mathematical Modeling

We formulate the VRPTW according to the model of operational research of **Solomon [4].** This formula translates the natural modeling of the problem by defining a binary variable x_{ij}^{k} equal to 1 if the vehicle K traverses the arc of the graph (V_i, V_j), 0 else; simply noted (i, j).

The transport problems are modeled by a graph G = (V, E) where V = $\{V_i\ V_n\}$ is a set of nodes modeling the cities (or customers) and E = $\{(V_i, V_j)$: $i \leq j$: $V_i, V_j \in V\}$ is a set of arcs connecting the cities. The decision variable of the problem is: x_{ij}^{k} = 1 if (i, j) is traversed by the vehicle K, 0 else.

Thus, as a problem of optimization, we propose the following formulation of the VRPTW; this formulation is inspired from the formulation of **Dridi [30].**

A. Objective Functions

Objectives to optimize are the minimization of the used vehicles number and the minimization of the total distance of the routes. Generally, these two goals are conflictual, but we will try to have an optimal compromise between the number of vehicles and the total distance of the routes.

$$\text{F1 : Cost(X)} = Minimize \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m} C_{ij} x_{ij}^{k} \qquad (1)$$

$$\text{F2: Number of vehicles(X)} = Minimize \sum_{v \to V}\sum_{j \to C} x_{0,j}^{v} \qquad (2)$$

Other objectives can be considered as perspectives:
- Minimizing the total duration of the routes, the total cost of the routes (taking into account the vehicles costs, drivers' costs ...).
- Minimizing the penalties related to the violation of some constraints, particularly the time windows ones.
- Considering the quality of service as a criterion to satisfy.

B. Constraints

We use the following notations:

- **n** = number of nodes,
- **m** = number of vehicles,
- **D** = capacity of a vehicle,
- **T_k** = maximum allowed time for the kth vehicle route,
- **d_i** = demand of the node i,
- **t^k_i** = necessary time taken by the vehicle k to load or unload demands,
- **t^k_{ij}** = necessary time taken by the vehicle k to travel from the summit i to j,
- **c_{ij}** = cost or distance to travel from the summit i to j,
- **x^k_{ij}**: $\begin{cases} 1 \text{ if the vehicle k travels from node i to j.} \\ 0 \text{ otherwise.} \end{cases}$
- **[ai, bi]**: defines the time window.
- **S_i^k**: instant when the vehicle k begins serving the customer.

The following constraints are related to the function F1:

$$\sum_{i=1}^{n}\sum_{k=1}^{m} x_{ij}{}^{k} = 1 \forall 1 \le j \le n \quad (3)$$

$$\sum_{j=1}^{n}\sum_{k=1}^{m} x_{ij}{}^{k} = 1 \forall 1 \le i \le n \quad (4)$$

$$\sum_{i=1}^{n} x_{ip}^{k} - \sum_{j=1}^{n} x_{pj}^{k} = 0, k = 1,\dots m; p = 1,\dots n \quad (5)$$

$$\sum_{i=1}^{n} t_{i}^{k} \sum_{j=1}^{n} x_{ij}^{k} + \sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij}^{k} x_{ij}^{k} \le T_{k}, k = 1,\dots,m \quad (6)$$

$$\sum_{i=1}^{n} d_{i} \left(\sum_{j=1}^{n} x_{ij}^{k} \right) \le D, k = 1,\dots,m \quad (7)$$

$$\sum_{j=2}^{n} x_{1J}^{k} \le 1, k = 1,\dots,m \quad (8)$$

$$\sum_{i=2}^{n} x_{i1}^{k} \le 1, k = 1,\dots,m \quad (9)$$

$$x_{ijk}\left(s_{ik} + t_{ij} - s_{jk} \right) \le 0 \quad (10)$$

$$\text{and :} \quad a_{i} \le s_{ik} \le b_{i} \quad (11)$$

The following constraint is related to the function F2:

$$\sum_{v \to V}\sum_{j \to C} x_{0,j}^{v} \le 25 \quad (12)$$

The equations (3) and (4) ensure that each node is served only once by only one vehicle. The equation (5) ensures the continuity of a route traversed by a vehicle: the visited node must imperatively be left. The equation (6) ensures the respect of the total duration of a route. The equation (7) ensures the respect of the vehicles capacity. The equations (8) and (9) ensure that a vehicle leaves the depot and returns to it only once. The equations (10) and (11) ensure the respect of the temporal constraints. The equation (12) ensures the respect of a maximum number of vehicles.

IV. Basic Algorithms

A. NSGA II

According to **Deb et al. [31]**, the NSGA-II is a new version of NSGA which is based on a classification of the individuals into different levels. It comprises the following phases:

1) Phase 0: Initialization:

It consists in generating the initial population in a random way.

2) Phase 1: Classification of the individuals according to the principle of the Non Dominance of Pareto:

According to **Deb et al. [31]**, it consists initially, before passing at the stage of the selection, to "assign to each individual population a rank" (using the Pareto ranking). All the non-dominated individuals of the same rank are classified in a category; we affect to it an efficiency which is inversely proportional to the Pareto rank of the considered category. The pseudo code of this classification procedure is represented in the algorithm below (figure 2). Figure 1 shows an example of a Pareto fronts classification. Front1 contains a set of solutions which are dominated by the Front2 points. The algorithm in Figure 2, inspired from **Ghali [32]** works, explains this phase.

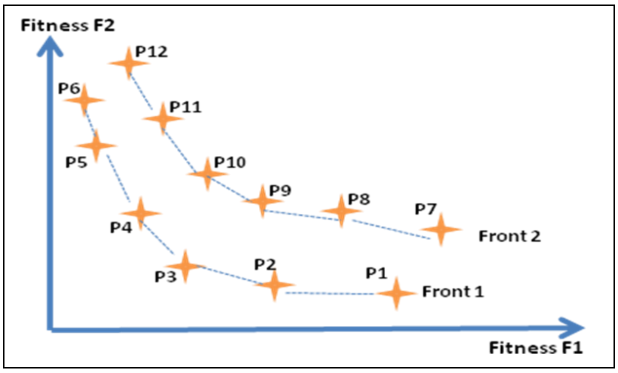

Figure 1. Classification of the individuals into several categories based on the Pareto rank

```
Beginning
  P = population ()   /* the whole set of the population */
  i = 1   /* i is the Pareto front counter, initialized to 1 */

  Repeat until P = an empty set
    Fi = seek-indiv-non-dominated(P) /* Finding F, the set of the non-dominated individuals corresponding to the Pareto front i */
    P = P \ Fi   /* Remove the non dominated individuals from the global population P */
    i = i + 1   / * Incrementing the front counter * /
  End Repeat
End
```

Figure 2. Algorithm of classification of the individuals

3) Phase 2: Diversity preservation using the Crowding Distance technique:

According to **Deb et al. [31]**, the NSGAII algorithm uses the technique of crowding distance to maintain the diversity of solutions on the Pareto front. This procedure is applied to the last Pareto front to complete the size of the parent population for the next generation.

4) Phase 3: Individuals selection according to the crowded comparison operator:

The crowded comparison operator guides the selection process with the uniform distribution of Pareto solutions. Each individual in the population has two attributes:

- The non-domination rank (i_{rank}).
- The crowding distance ($i_{distance}$).

Having two individuals i and j: $(i <_{n} j$ ***if*** $i_{rank} < j_{rank})$ ***or*** $(i_{rank} = j_{rank})$ ***and*** $(i_{distance} > j_{distance})$: when comparing two non-dominated solutions from two Pareto fronts, it is preferred to select the solution belonging to the Pareto front of the lowest order. If the two solutions belong to the same Pareto front, we choose the solution having the greater crowding distance.

5) Phase 4: OX Crossing:

According to **Haj-Rashid et al. [33]**, we chooses two random crossing points. The child inherits elements located between the two crossover points of the first parent. These elements occupy the same positions, and appear in the same order in the child.

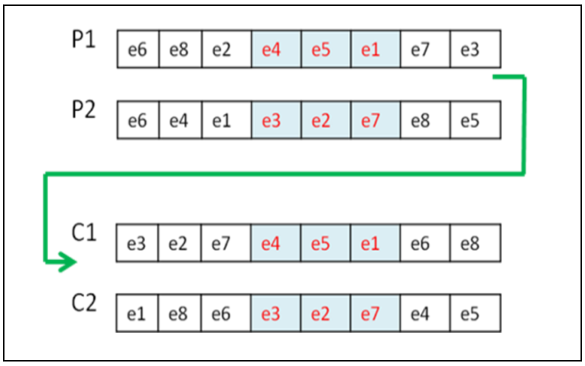

Figure 3. The OX crossing Illustration

6) Phase 5: Mutation:

Usual mutation by inversion of the positions. The following example illustrates the mutation by inversion:

$$00110\underline{1} \longrightarrow 00110\underline{0}$$

7) Phase 6: determining a stopping criterion:

According to **Ghali [32]**, this criterion ensures the convergence and can be a time limit (not to exceed), a maximum number of generations, or a combination of the two previous criteria.

B. BCRC

To carry out the crossover, we apply the Best Cost Route Crossover operator "BCRC" proposed by **Ombuki and al. [34]**. The BCRC is the most efficient operator for a fast convergence towards the optimum. Thus, we propose an hybridization of the NSGAII algorithm with the BCRC. The BCRC algorithm is detailed in Figure 4.

```
Size ← constant ;
P1 ← choose_parent(Size) ;
P2 ← choose_parent(Size) ;

RP1 ← decompose_into_routes(P1) ;
RP2 ← decompose_into_routes(P2) ;

t_P1 ← choose_route(P1) ;
t_P2 ← choose_route(P2) ;

remove_cities(RP1,t_P2) ;
remove_cities(RP2,t_P1) ;

insert (t_P2, RP1) ;
insert (t_P1, RP2) ;
```

Figure 4. Algorithm of the BCRC operator

V. A BCRC-NSGAII Algorithm to resolve the VRPTW

The general principle of the used genetic algorithm is represented in the Figure 5 below.

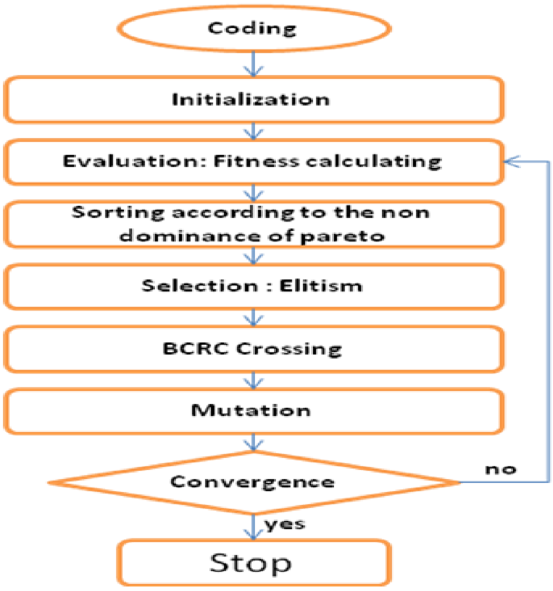

Figure 5. The BCRC-NSGAII algorithm

A. Representing a chromosome

A coding (or a representation) of an individual must include the basic characteristics of the problem, it must also be easy to handle by the operators of variation to allow an easy transformation on the search space and to generate, if possible, acceptable solutions. Thus, a good encoding must:

- Facilitate the definition and the application of the variation operators (genetic transformations: mutation, crossover ...) to properly cover the individuals space.
- Be in conformity with the addressed problem and must be simple in its construction and provide a simple and an effective transition to the search space (and vice versa).

The following chromosome (Figure 6) represents a possible solution for the CVRPTW with 9 cities. It presents the number of visited cities. Each sequence Ti is a route. Routes are separated by the depot which has the index 0.

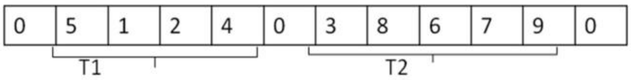

Figure 6. Representation of a chromosome

B. Generation of the initial population

In the NSGAII algorithm, an initial population P is generated in a random way and has a size N.

C. Objective functions

We have two objective functions to be minimized:

- The function of the Euclidean cost of the visited cities:

$$F1 = Minimize \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{m} C_{ij} x_{ij}^{k}$$

- And the function of minimizing the number of vehicles:

$$F2 = Minimize \sum_{v \to V} \sum_{j \to C} x_{0,j}^{v}$$

D. The crossing operator BCRC

To carry out the crossover, we apply the Best Cost Route Crossover operator "BCRC" proposed by **Ombuki et al. [34]**. According to them, this type of crossing is efficient in resolving the VRPTW. It is the most efficient operator for a fast convergence towards the optimum. Thus, we propose a hybridization of the NSGAII algorithm with the BCRC. The BCRC algorithm is detailed in Figure 8. Figure 7 shows two examples of vehicles routing. Figure 8 shows the application of the BCRC operator on the two previous examples.

E. Mutation operator

While executing the evolution process, mutation performs a larger exploration of the search space, to avoid any premature convergence or disappearance of diversity while bringing innovation to the population. The usual mutation is carried out by reversing the position of two genes.

F. Stopping criterion

The stopping criterion is the maximum number of generations. We take the value 350 as a stopping criterion to be able to compare our works with those of **Ombuki et al. [34]**.

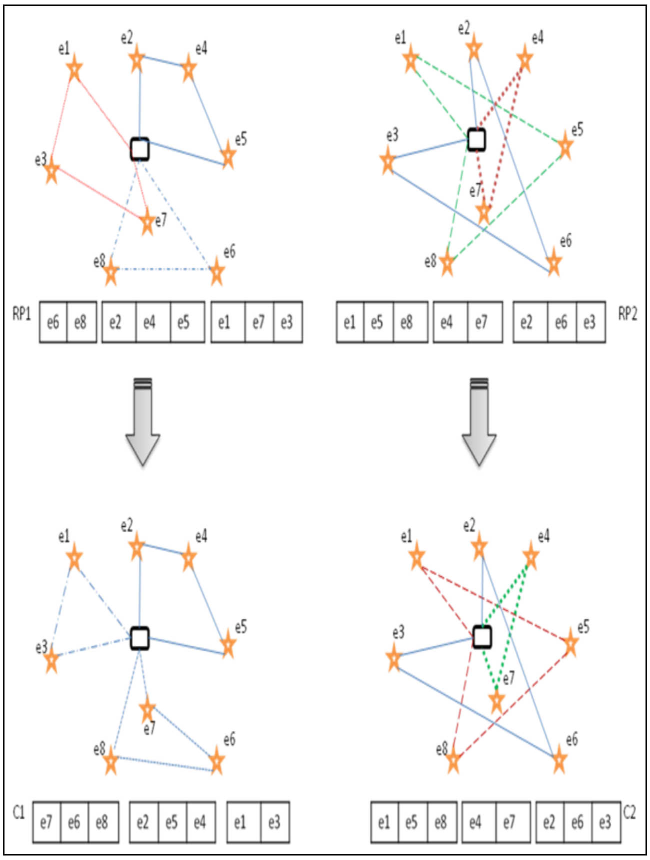

Figure 7. Examples of vehicles routing

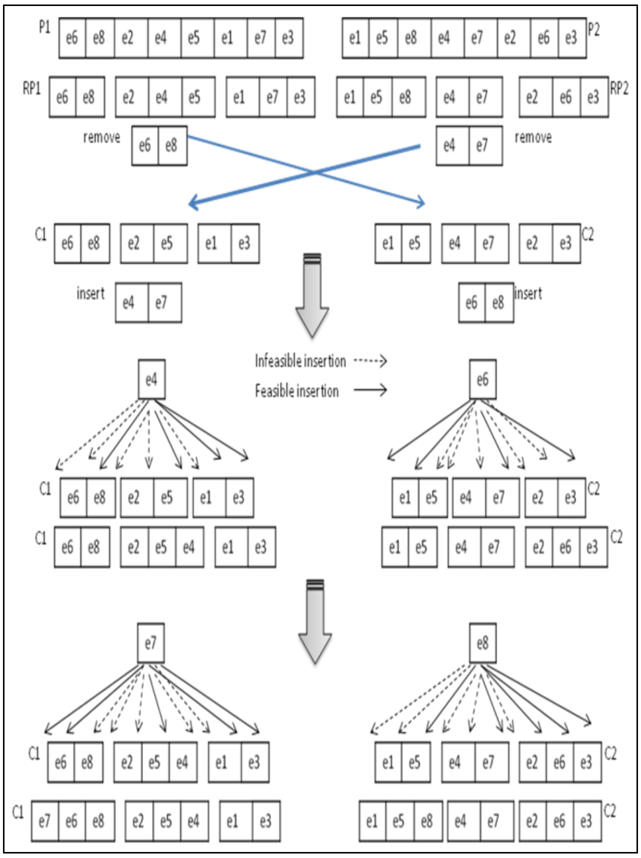

Figure 8. Crossing BCRC (inspired from Ombuki et al. [34])

VI. Implementation and obtained results

A. Working environment

The application that we have implemented is performed on a laptop Intel Core 2 Duo, processor of 2.00 GHz. We use the Java language and the XML.

B. Parameters of the algorithm

We use the same parameters taken by **Ombuki et al. [34]** which are:

- Number of cities = 100.
- Maximum number of generation = 350.
- Number of individuals per generation = 300.
- Probability of mutation = 0.1.
- Probability of crossover = 0.8.
- Number of constraints = 03 (only one depot, the time window and the limited capacity).

C. Used benchmarks

The instances of **Solomon [4]** are the benchmarks of reference for the VRPTW, which constitutes, according to **Bräysy and Gendreau [24]**, an efficient mean to compare the different methods solving this problem (see table 2).

Table 2. Characteristics of the used instances.

Instances type	Instances number	Vehicles capacity		Clients spatial repartition		Service duration		Time Window	
		Small	Large	clustered	Random	long	Short	Short	Large
R1	12	×			×		×	×	
R2	11		×		×	×			×
C1	9	×		×			×	×	
C2	8		×	×		×			×
RC1	8	×		×	×		×	×	
RC2	8		×	×	×	×			×

*D. Comparing our results with those of **Ombuki et al.*** **[34]**

In the table 3, we compare our results with the AG-BCRC approach of **Ombuki et al. [34]**.

Table 3. Comparing our results with the AG-BCRC results.

	AG and BCRC		**BCRC-NSGAII**	
	Vehicles number	**Distance**	**Vehicles number**	**Distance**
R101	19	1690.2	16	1532.1
R102	18	1513.7	15	1519,8
R103	14	1237.0	14	1551.8
R104	10	1020.8	10	1550.3
R105	15	1415.1	13	1391.2
R106	13	1254.22	12	1507 .0
R107	11	1100.5	9	1552.3
R108	10	975.34	10	1520.93
R109	12	1169.8	10	1541,10
R110	11	1112.2	10	1445.83
R111	11	1084.7	10	1041,1
R112	10	976.99	9	1066.62

Figure 9 compares our results with those obtained by the AG-BCRC according to the vehicles number.

Figure 10 compares our results with those obtained by the AG-BCRC according to the distance.

To compare our approach with the approach of **Ombuki et al. [34]**, we use the same parameters of the algorithm. We obtain the following results: Our approach gives better results for the first objective with a rate of 75%. Whereas results are equivalent for 25% of the instances. These results show that our approach is competitive for the optimization of the number of vehicles. Our approach aims at finding a compromise between the two objectives.

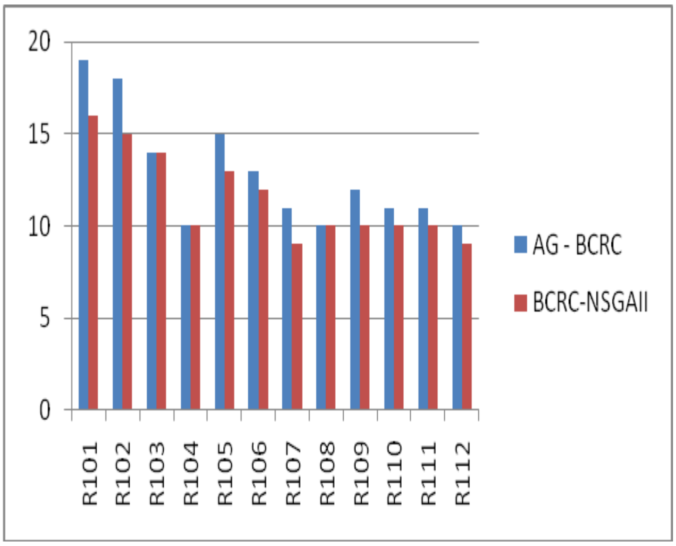

Figure 9. Comparison of our results with the AG-BCRC method according to the number of vehicles.

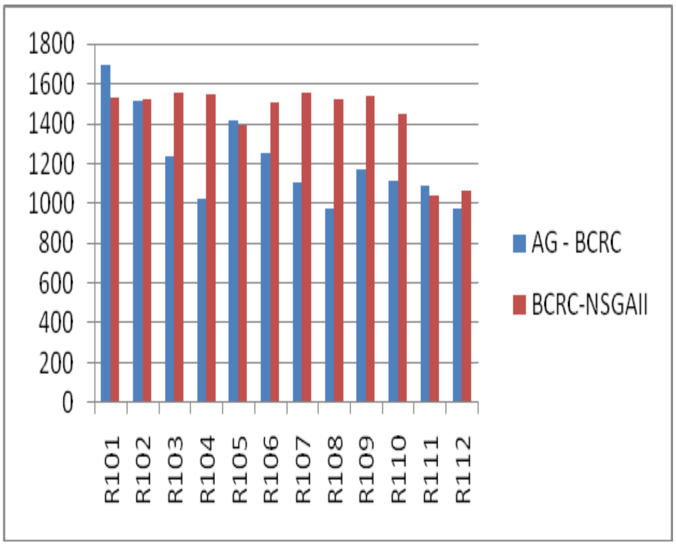

Figure 10. Comparison of our results with the AG-BCRC method according to the distance.

E. Comparing our results with the best results

Table 4 shows a comparison of our results with the best known ones for the Solomon instances. Figure 11 and Figure 12 compare our results with the best obtained results, respectively according to the distance and the number of vehicles.

Table 4. Comparison of our results with the best known results of the Solomon instances.

		R101	R102	R103	R104	R105	R106	R107	R108	R109	R110	R111	R112
Vehicles Number	**BCRC-NSGA II**	18	17	13	9	14	12	10	9	11	10	10	9
	Best known results	16	15	14	10	13	12	9	10	10	10	10	9
Distance	**BCRC-NSGA II**	1532,1	1519,88	1551,81	1550,38	1391,21	1507,07	1552,38	1520,93	1541,1	1445,83	1041,1	1066,62
	Best known results	1611	1486,12	1292,68	1007,24	1377,11	1251,98	1104,66	960,88	1194,73	1118,59	1096,7	982,14
Used algorithm to obtain the best known results		Ant colony, **Ding et al. [35]**	Diversification, intensification and local search,**Taillard and Rochat [18]**	Simulated annealing and local search, **Li et al.[36]**	multi- parametric evolution strategy, **Mester et al. [37]**	Programming by constraints and local search, **Shaw[38]**	multi- parametric evolution strategy, **Mester et al. [37]**	Programming by constraints and local search, **Shaw [38]**	parallel and hybrid genetic algorithms, **Berger et al. [39]**	Hybridization of the evolution strategy and the tabu search, **Hao et al. [40]**	multi- parametric evolution strategy, **Mester et al. [37]**	Ant colony, **Rousseau et al. [41]**	Ant Colony, **Gambardella et al. [42]**

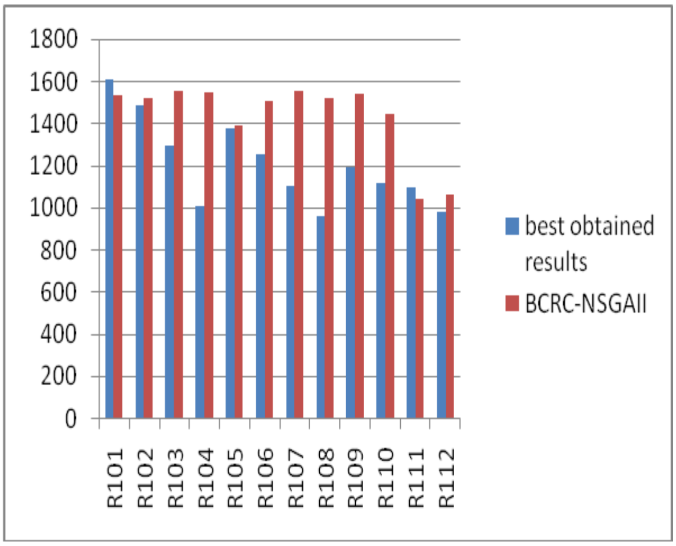

Figure 11. Comparison of our results with the best known results according to the distance.

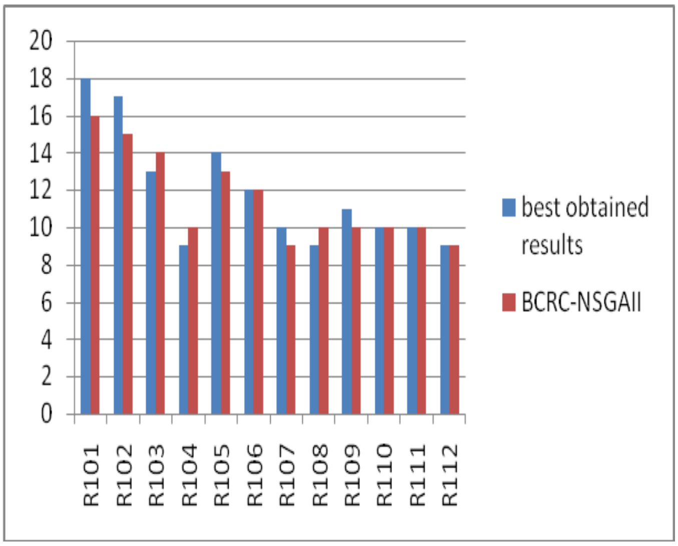

Figure 12. Comparison of our results with the best known results according to the number of vehicles

Results shows that our approach provides competitive results compared to the ant colony algorithm for the instances R101 and R111 in terms of minimizing the number of vehicles and the total traveled distance.

Similarly, the hybridization (BCRC-NSGAII) gives competitive results and provides better results than the method of **Taillard and Rochat [18]** for the instances R102. Our approach is also better than the hybridization of the programming by constraint and the local search (for the instances R105 and R107).

Also the strategy of multi-parametric evolution proposed by **Mester et al. [37]**,, gives an optimal number of vehicles. Concerning the total traveled distance, our approach did not reach the optimal value (for the instances R104, R106 and R110). We can also note that the hybridization of the simulated annealing with the local search (R103) and the parallel hybrid genetic algorithms (R108) give better results than the hybridization of the NSGAII with the BCRC.

Our approach gives an optimal value for the first objective function, with a rate of 75%, which shows that our approach is competitive for minimizing the number of vehicles. Whereas it succeeds in minimizing the total traversed distance only for two instances (R101 and R111).

F. Comparison of the average for the class R1

The following table (table 5) presents a comparison of the average for the class RI.

Table 5. Average of the vehicles number and the total travelled distance.

	Best results	**AG and BCRC Hybridization**	**BCRC-NSGAII**
Vehicles Number	11,83	12,83	11,50
Total Distance	1206,98	1212,58	1435,03

Indeed, the hybridization BCRC-NSGAII minimizes the first objective (the number of used vehicles) while it does not gives the best results for the second objective (the total traversed distance). We conclude that our approach is more adapted to the problems aiming to optimize the number of vehicles.

VII. Conclusion

In this paper we presented our approach which consists in applying the BCRC-NSGAII algorithm to resolve the VRPTW. Obtained results show that our method was able to give best results for the first objective (the vehicles number) but not for the second objective (the total traversed distance).

References

[1] I. Zidi, K. Zidi, K. Ghedira and K. MESGHOUNI, "A Multi-Objective Simulated Annealing for the Multi-Criteria Dial a Ride Problem," The special issue of the International Journal of Intelligent Real-Time Automation.Elsevier Publishers,2011.

[2] A. Raddaoui and K. Zidi, "Distributed Genetic Algorithm NSGA II for solving the DARP," 4th International Conference on Metaheuristics and Nature Inspired Computing, Sousse - Tunisia, 27- 31 october 2012.

[3] S. Mnasri, K. Zidi and K. Ghedira," A heuristic multi-agents model to solve the P||Cmax: application to the DDBAP," 13th International Conference on Hybrid Intelligent Systems, Hammamet-Tunisie, 04- 06 decembre 2013.

[4] M.M. Solomon,"Algorithms for the vehicle routing and scheduling problems with time window constraints," Operations Research 35, 254–265, 1987.

[5] G. Laporte,"The Vehicle Routing Problem: An overview of exact algorithms," European journal of Operational Research, 1992.

[6] M. Savelsbergh,"Local search in routing problems with time windows," Annals of Operations Research, 4, 1985, pp. 285–305.

[7] B. Dantzig, D.R. Fulkerson, and S.M., "Johnson. Solution of a large-sale traveling-salesman problem," Operations Researh, 2 : 393-410, 1954.

[8] M. Desrochers, J. K. Lenstra, M. W. P. Savelsbergh, and F. Soumis," Vehicle Routing: Methods and Studies," chapter "Vehicle routing with time windows: Optimization and approximation", In B. L. Golden and A. A. Assad, editors, pages 65–84. North-Holland, Amsterdam, 1988.

[9] M. Desrochers, J. Desrosiers and M. Solomon, "A new Optimization Algorithm for the Vehicle Routing Problem with Time Windows," Operations Research 40, 342 – 354. 1992

[10] P. Badeau ,M. Gendreau, F. Guertin ,J-Y Potvin ,E. Taillard, "A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows," p29, Transportation Science 31(2): 170-186. 1997.

[11] a. w. j., a. h. g Kolen, R. Kan and H. w. j. m. Trienekens, "Vehicle Routing with Time Windows," Opns. Res. 35, 266-273. 1987

[12] M. Desrochers, "Thése : La fabrication d'horaires de travail pour les conducteurs d'autobus par une méthode de génération de colonnes," Département d'informatique et de Recherche Opérationnelle, Université de Montréal,1986.

[13] J-Y Potvin, and S. Bengio, "A Genetic Approach to the Vehicle Routing Problem with Time Windows," Technical Report, CRT-953, Centre de Recherche sur les Transports, Universite de Montreal, C.P. 6128, Succ. Montreal, Canada H3C 3J7. 1993.

[14] R.A. Russell, "Hybrid heuristics for the vehicle routing problem with time windows," In Transportation Science, 29:156–166, 1995.

[15] E. Baker, and J. Schaffer M.M. Solomon, "Vehicle routing and scheduling problems with time window constraints: Efficient implementations of solution improvement procedures." In B.L. Golden and A.A. Assad, editors, Vehicle routing: Methods and studies, pages 85–106. North-Holland, Amsterdam, 1988.

[16] P.M. Thompson and H.N. Psaraftis. "Cyclic transfer algorithms for multi-vehicle routing and scheduling problems," Operations Research, 41:935–946, 1993.

[17] G. Kontoravdis and J.F. Bard, "A GRASP for the vehicle routing problem with time windows," ORSA Journal on Computing, 7:10–23, 1995.

[18] E. Taillard and Y. Rochat,"Probabilistic diversification and intensification in local search for vehicle routing," Journal of Heuristics, 1:147–167, 1995.

[19] C. Chiang and R. Russell, "Simulated Annealing Metaheuristics for the Vehicle Routing Problem with Time Windows," Department of Quantitative Methods, University of Tulsa, Tulsa, OK 74104,USA. 1993.

[20] J.L.Blanton and R.L. Wainwright, "Multiple vehicle routing with time and capacity constraints using genetic algorithms," In Proceedings of the 5th International Conference on Genetic Algorithms, pages 452–459, San Mateo, CA, 1993.

[21] J.-Y. Potvin and S. Bengio, "The vehicle routing problem with time windows – Part II: Genetic search," INFORMS Journal on Computing, 8:165–172, 1996.

[22] S.R. Thangiah, "Vehicle Routing with Time Windows using Genetic Algorithms," Technical Report SRU-CpSc-TR-93-23, Computer Science Department, Slippery Rock University, Slippery Rock, PA, 1993.

[23] S.R. Thangiah, I.H. Osman, and T. Sun, "Hybrid genetic algorithm, simulated annealing and tabu search methods for vehicle routing problems with time windows," Technical Report UKC/OR94/4, Institute of Mathematics and Statistics, University of Kent, Canterbury, UK, 1994.

[24] O. Bräysy and M. Gendreau,” Vehicle Routing Problem with Time Windows, Part I: Route construction and local search algorithms” Transportation Science,39(1): 101-118, 2005.

[25] J. Homberger, H. Gehring, “Two evolutionary metaheuristics for the vehicle routing problem with time windows,” Inform. Systems Operational Research, 37 297–318. 1999.

[26] K. C. Tan, L. H. Lee, K. Ou, “Hybrid genetic algorithms in solving vehicle routing problems with time window constraints,” Asia-Pacific J. Oper. Res. 18 121–130, 2001.

[27] K. C. Tan, T. H. Lee, K. Ou, L. H. Lee, “A messy genetic algorithm for the vehicle routing problem with time window constraints,” Proc. 2001 Congress Evolutionary Comput., IEEE Service Center, Pistacaway, NJ, 679–686.

[28] A. Le Bouthillier, T. G. Crainic, “A cooperative parallel metaheuristic for vehicle routing with time windows,” *Comput. Oper. Res.* 32 1685–1708, 2005.

[29] S. Jung, B.-R. Moon, “A hybrid genetic algorithm for the vehicle routing problem with time windows,” Proc. Genetic Evolutionary Comput. Conf., Morgan Kaufmann, San Francisco, 1309–1316. 2002.

[30] L. Dridi, ”thesis : Développement et validation d’une approche de remplacement des conduites pour les réseaux d’eau potable,” Institut National de la Recherche Scientifique - Eau Terre et Environnement, Québec, 317 p, Canada, 2005.

[31] K. Deb, S. Agrawal, A. Pratap and T. Meyarivan, “A Fast Elitist Non–dominated Sorting Genetic Algorithm for Multi–Objective Optimization: NSGA–II,” Parallel Problem Solving from Nature PPSN VI, 2000.

[32] K. Ghali, “Methodologie de conception systeme a base de plateformes reconfigurables et programmables,“ Universite Paris xi UFR scientifique d’Orsay, 2005.

[33] M. Haj-Rachid, C. Bloch ,W. Ramdane-Cherif , P. Chatonnay**,** “Différentes opérateurs évolutionnaires de permutation: sélections, croisements et mutations,“ Rapport de Recherche noRR2010–07, p 60, Juillet 2010.

[34] B. Ombuki, B. J. Ross and F. Hanshar,“Multi-Objective Genetic Algorithms for Vehicle Routing Problem with Time Windows,” Springer Science + Business Media, Inc. Manufactured in The Netherlands, 2006.

[35] Q. Ding, X. Hu, L. Sun,Y. Wang, “An improved ant colony optimization and its application to vehicle routing problem with time windows”, Neurocomputing,2012, http://dx.doi.org/10.1016/j.neucom.2011.09.040

[36] H. Li, A. Lim, and J. Huang. “Local Search with Annealing-like Restarts to Solve the VRPTW,”. Working Paper, Department of Computer Science, National University of Singapore, 2001.

[37] D. Mester, O. Bräysy and W. Dullaert,”A Multi-parametric Evolution Strategies Algorithm for Vehicle Routing Problems,”Institute of Evolution, University of Haifa, 2005.

[38] P. Shaw.”Using constraint programming and local search methods to solve vehicle routing problems,”. In Proceedings of the Fourth International Conference on Principles and Practice of Constraint Programming (CP’98), edited by M. Maher and J.-F. Puget, Springer-Verlag, pp. 417–431, 1998.

[39] J. Berger, M. Barkaoui and O. Bräysy.”A Parallel Hybrid Genetic Algorithm for the Vehicle Routing Problem with Time Windows,”, Working paper, Defense Research Establishment Valcartier, Canada, 2001.

[40] J. K. Hao, P.Galinier and M.Habib.”Métaheurstiques pour l’optimisation combinatoire et l’affectation sous contraintes,” Revue d’Intelligence Artificielle,13(2) :283 -324, 1999.

[41] L.M. Rousseau, M. Gendreau, and G. Peasant,”Using constraint-based operators to solve the vehicle routing problem with time windows,” Journal of Heuristics. 2002.

[42] L.M. Gambardella, E. Taillard, and G. Agazzi.”MACS-VRPTW: A multiple ant colony system for vehicle routing problems with time windows,” Edited by David Corne, Marco Dorigo, and Fred Glover, New Ideas in Optimization, , McGraw-Hill: London, pp. 63–76, 1999.

Author Biographies

Sami Mnasri received his Engineering degree in Computer Science (speciality: Software Engineering.) from ESSTT University, Tunis, Tunisia in 2008, Master of Research in Computer Science (Speciality: Information Systems and New Technologies) from FSEGS University, Sfax, Tunisia in 2012 and he is now a PhD student in Computer Science in IRIT, Toulouse, France. He is currently an assistant professor in the Department of Computer Science at the FSG University, Gafsa, Tunisia. His research interests include optimization using the metaheuristics, deployment strategies, coverage, connectivity and energy optimization issues in wireless sensor networks.

Fatma Abbes received his Engineering degree in Computer Science from FSEGS University, Sfax, Tunisia in 2007, Master of Research in Computer Science from ISIMG University, Gabes, Tunisia in 2014. He is currently an assistant professor in the Department of Computer Science at the FSG University, Gafsa, Tunisia. His research interests include optimization using the metaheuristics and the transport problems.