

A HEURISTIC COLUMN GENERATION METHOD FOR THE HETEROGENEOUS FLEET VRP (*)

by É.D. TAILLARD (¹)

Communicated by Brian BOFFEY

Abstract. – *This paper presents a heuristic column generation method for solving vehicle routing problems with a heterogeneous fleet of vehicles. The method may also solve the fleet size and composition vehicle routing problem and new best known solutions are reported for a set of classical problems. Numerical results show that the method is robust and efficient, particularly for medium and large size problem instances.*

Keywords: Vehicle routing, iterative searches, parallel algorithms.

Résumé. – *Ce papier présente une méthode heuristique de génération de colonnes pour la résolution de problèmes d'élaboration de tournées de véhicules avec flotte hétérogène. Cette méthode peut aussi s'appliquer au problème du choix de la composition d'une flotte de véhicules et de nouvelles meilleures solutions ont été obtenues sur un ensemble de problèmes de la littérature. Les résultats numériques montrent que la méthode proposée est robuste et efficace, particulièrement pour les problèmes de taille moyenne et élevée.*

Mots clés : Routage de véhicules, recherches itératives, algorithmes parallèles.

1. INTRODUCTION

The vehicle routing problem with a heterogeneous fleet of vehicles (VRPHE) is a major optimization problem. Indeed, most companies that have to deliver or collect goods own a heterogeneous fleet of vehicles. Recently, the solution methods for homogeneous vehicle routing problems (either the classical VRP or the VRP with time windows) have substantially progressed (*see*, e.g. Taillard, 1993, Rochat and Taillard, 1995, Rego and Roucairol, 1996, Potvin and Bengio, 1996, and Badeau *et al.*, 1997). However the VRPHE has attracted much less attention. This is mainly due to the fact that the VRPHE is much harder to solve than the classical VRP.

(*) Received September 1995.

(¹) IDSIA, Corso Elvezia 36, 6900 Lugano, Switzerland,
e-mail: eric@idsia.ch, www. <http://www.idsia.ch/~eric>

The latter may be described as follows: identical vehicles with a fixed carrying capacity Q must deliver order quantities q_i ($i = 1, \dots, n$) of goods to n customers from a single depot ($i = 0$). Knowing the distance d_{ij} between customers i and j ($i, j = 0, \dots, n$) the problem is to find tours for the vehicles in such a way that:

- The total distance travelled by the vehicles is minimized.
- Only one vehicle handles the deliveries for a given customer.
- The total quantity of goods that a single vehicle delivers cannot be larger than Q .

Several variants of VRPs exist, they may include constraints such as:

- The limitation of the length, duration or cost of each individual tour,
- The addition of a service time for each customer,
- The addition of time windows during which the customers have to be visited.

We are not going to illustrate our method on these variants. However, it may be easily adapted to treat such variants, if the iterative search embedded in the column generation procedure is able to take these additional constraints into account.

In the heterogeneous problems, we have a set $\Psi = \{1, \dots, K\}$ of different vehicle types. A vehicle of type $k \in \Psi$ has a carrying capacity Q_k . The number of vehicles of type k available is n_k . The cost of the travel from customer i to j ($i, j = 0, \dots, n$) with a vehicle of type k is d_{ijk} . The use of one vehicle of type k implies a fixed cost f_k .

A special case of VRPHE is the fleet size and mix vehicle routing problem (Golden *et al.*, 1984) also called the fleet size and composition VRP or the vehicle fleet mix (VFM, Salhi *et al.*, 1992). The goal of this problem is to determine a fleet of vehicles such that the sum of fixed costs and travel costs is minimized. This problem is a particular VRPHE for which:

- 1) The travel costs are the same for all vehicle types ($d_{ijk} = d_{ijk'}$, $k, k' \in \Psi$).
- 2) The number n_k of vehicles of each type is not limited ($n_k = \infty$, $k \in \Psi$).

Another special case of the VRPHE is the VFM with variable unit running costs (VFMVRC, Salhi *et al.*, 1992). The VFMVRC is a particular VRPHE for which $n_k = \infty$, $k \in \Psi$. Several papers on the VFM have been published. Golden *et al.* (1984) were among the first to address this problem. They proposed a set of test problems we have used in this paper. Among the

heuristic methods proposed for the VFM, let us mention those of Golden *et al.* (1984), Geysens *et al.* (1984, 1986), Desrochers and Verhoog (1991), Salhi and Rand (1993) and Osman and Salhi (1996). The reader is referred to Salhi and Rand (1993) for a literature review. Adaptations of heuristic methods for the VFM for solving the VFMVRC may be found in Salhi *et al.* (1992). Much less work has been done for the VRPHE. Let us quote the taboo searches of Semet and Taillard (1993) and Rochat and Semet (1994) for real-life problems including many other constraints.

For homogeneous VRPs, many heuristic methods have been proposed. Among the most efficient ones, are the adaptive memory procedure (AMP) of Rochat and Taillard (1995) and the taboo search of Taillard (1993). This last method uses a local search mechanism based on the move of one customer from one tour to another or the exchange of two customers that belong to different tours. Since the vehicles are identical, it is easy to check the feasibility of a move and to evaluate its cost. For the VRPHE, the feasibility check or the evaluation of a move requires finding a new assignment of the vehicles to the new solution's tours. In Semet and Taillard (1993), several techniques have been proposed to simplify and accelerate the re-assignment of vehicles to tours. However, the re-assignment problem is very simple in the case of the VFM: each tour is performed with the cheapest vehicle type that is able to carry all the orders of the tours. This is certainly a reason that the VFM has been more studied than the VRPHE.

In this paper, we present a heuristic column generation method for solving the VRPHE. The column generation is based on the AMP of Taillard (1994) which uses an embedded taboo search due to Taillard (1993). Section 2 describes the heuristic column generation proposed to solve the VRPHE. Numerical results on a set of test problems are given in Section 3.

2. HEURISTIC COLUMN GENERATION PROCEDURE

2.1. General description of the method

Since the solution methods for homogeneous VRPs are becoming more and more efficient, we propose to treat the VRPHE by solving a succession of homogeneous VRPs. For each type of vehicle, we solve a homogeneous VRP (without limitation on the number of vehicles available) with an AMP. The tours of the homogeneous VRP solutions are then combined to produce a solution to the VRPHE. The method used to solve the homogeneous VRPs was originally proposed by Taillard (1994) and published in Rochat and Taillard (1995).

The AMP first generates a set of good solutions using the taboo search of Taillard (1993). It then extracts single vehicle tours from this set of solutions, and combines some of these tours to define a partial starting solution for another application of taboo search. This process is repeated a number of times and the tours are memorized as candidates for the final VRPHE solution. Once the homogeneous VRPs are solved for each vehicle type, one has a set T of tours that have been memorized. The useless tours of T are removed: only one copy of each tour is kept in T ; the dominated tours are eliminated (a tour is dominated if it is more expensive than another tour of T servicing the same customers).

Let $m = |T|$ and c_{jk} ($j = 1, \dots, m$, $k = 1, \dots, K$) be the cost of the j^{th} tour of T performed by a vehicle of type k ($c_{jk} = \infty$ if the j^{th} tour of T cannot be performed by a vehicle of type k). Let us define the matrix

$$A = (a_{ij}) \text{ with : } a_{ij} = \begin{cases} 1 & \text{if customer } i \in j^{\text{th}} \text{ tour of } T \\ 0 & \text{otherwise} \end{cases}$$

The j^{th} tour of T defines a column of A . Consequently, the AMP is a heuristic column generation procedure in the sense that only a very limited fraction of all the tours that can be constructed for the homogeneous VRP are generated. The best solution to the VRPHE that can be constructed from the tours of T can be found by solving the following boolean linear programme (A):

$$\begin{aligned} \min & \sum_{k=1}^K \sum_{j=1}^m c_{jk} x_{jk} \\ \text{s. t. } & \sum_{k=1}^K \sum_{j=1}^m a_{ij} x_{jk} = 1 \quad i = 1, \dots, n \\ & \sum_{j=1}^m x_{jk} \leq n_k \quad k = 1, \dots, K \\ & x_{jk} \in \{0, 1\} \quad j = 1, \dots, m, \quad k = 1, \dots, K. \end{aligned} \tag{A}$$

The value of x_{jk} is 1 if, and only if, the j^{th} tour of T is selected and performed by a vehicle of type k . In the case of the VFM, let $c_j = \min_{k \in \Psi} (c_{jk})$, ($j = 1, \dots, m$). The best solution that can be constructed with the tours of

T may be determined by solving the set partitioning problem (B), where x_j is a variable that indicates whether the j^{th} tour of T has to be selected or not:

$$\begin{aligned} \min \quad & \sum_{j=1}^m c_j x_j \\ \text{s. t.} \quad & \sum_{j=1}^m a_{ij} x_j = 1 \quad i = 1, \dots, n \\ & x_j \in \{0, 1\} \quad j = 1, \dots, m. \end{aligned} \quad (\text{B})$$

Both problems (A) and (B) are NP-hard but relatively easy to solve optimally. A general software, like CPLEX, is able to find optimum solutions to set partitioning problems with several hundred rows and columns (*see* Chu and Beasley, 1995).

2.2. Algorithm for the VRPHE

More formally, we propose the following algorithm:

- 1) For each type $k \in \Psi$ of vehicle, repeat: (AMP)
 - 1.1) With the taboo search algorithm of Taillard (1993), generate I initial solutions to the homogeneous VRP with vehicles of type k .
 - 1.2) Memorize the tours of these solutions in the set M_k .
 - 1.3) Repeat R times:
 - 1.3.1) $M'_k = M_k$.
 - 1.3.2) Repeat, until $M'_k = \emptyset$ (construction of a new initial solution)
 - 1.3.2.1) Randomly select a tour from M'_k , according to a criterion that gives a larger weight to tours that are often generated or that belong to better VRP solutions.
 - 1.3.2.2) Remove from M'_k all tours that have customers in common with already selected tours.
 - 1.3.3) Using the selected tours as starting point, apply taboo search to generate a new VRP solution.
 - 1.3.4) Append the individual tours of this solution to M_k .
- 2) $T = \bigcup_{k \in \Psi} M_k$
- 3) For each tour t_j of T , compute c_{jk} the cost of t_j when performed with a vehicle of type k .
- 4) Remove the useless tours of T .

5) With CPLEX, solve the integer linear programme (A) associated with T .

The algorithm we propose for the VFM is essentially the same, but the set of useless tours at step 4) is extended with the tours not performed by the most appropriate vehicle type and step 5) solves the integer linear programme (B).

Step 1) of this algorithm corresponds to the adaptive memory procedure of Taillard (1994) applied once for each vehicle type. This procedure can be easily and efficiently parallelized with I processes (*see e.g. Badeau et al., 1997*). Therefore, step 1) may be parallelized with KI processes. If the homogeneous VRPs solved at steps 1.1) and 1.3.3) involve more than 6 to 8 vehicles, the taboo search of Taillard (1993) may also be parallelized. This means that the overall parallelization potential of step 1) is very high. In this algorithm, it is clear that the taboo search of Taillard (1993) may be replaced by another improving search, for example simulated annealing. Rochat and Taillard (1995) successfully used another taboo search that was able to treat vehicle routing problems with time windows. The reader is referred to this last reference for more details on the AMP.

In the case of the VFM, the algorithm always produces a feasible solution if the iterative search used to solve the homogeneous VRP succeeds in finding feasible solutions. This is always the case with the taboo search of Taillard (1993). For the VRPHE, there is no guarantee that the algorithm produces a feasible solution. However, during the elaboration of the present paper, all runs of the algorithm have produced a feasible solution for the problem instances including at least 50 customers. It was found that for problem instances involving very few vehicles, there was a higher probability that a run would not produce a good or even a feasible solution. For such instances, our method might be inappropriate and the use of a more direct algorithm, such as an adaptation of the taboo search of Semet and Taillard (1993), might be preferable.

Pre- and post-processing

For some problem instances, the smallest vehicles may not have sufficient capacity to carry the order of customers with large demand. Before performing step 1.1) of the algorithm, it is necessary to eliminate these customers.

Since step 5) of the algorithm solves a NP-hard problem exactly, the computation time may become prohibitive if the number of tours in T is large. Consequently, it might be necessary to reduce the size of T . This may be done in step 2) by storing in T only the tours of the best

homogeneous VRP solutions contained in M_k ($k = 1, \dots, K$) and not all the tours of M_k . For example, only the tours belonging to the 10 to 30 best solutions produced by the AMP may be introduced in the set T . A similar approach was successfully applied for the vehicles routing problem with multiple use of the vehicles (Taillard *et al.*, 1996). However, we did not reduce the size of T for the numerical results that follows.

Parameter settings

The parameter settings of our method consist of fixing the parameters of the AMP for solving the homogeneous VRPs. We have chosen $I = 10$ initial solutions and $R = 30$ constructions of solutions using the adaptive memory. Each call of the taboo search of Taillard (1993) embedded in the AMP performs $10n$ iterations.

3. NUMERICAL RESULTS

Since we are not aware of VRPHE instances in the literature, we have first to generate a set of VRPHE instances, and then to show that our method for the VRPHE is efficient. To convince the reader that the method is actually efficient, we proceed in three steps:

1) We show that our method produces solutions that are, on average, better than the best solutions published for the VFM instances of Golden *et al.* (1984), with $n \geq 50$ and fixed costs only. This will show that our method is very efficient for the VFM.

2) We find solutions to VFM instances with variable costs. Since this problem type is very similar to the problem with fixed costs, it is reasonable to assume that the solutions found with variable costs are very good.

3) Finally, we solve the VRPHE instances with variable costs and imposed fleet and show that the values found for these instances are very close to the values found for VFM instances with variables costs. Therefore, it is also reasonable to assume that our method produces good solutions for the VRPHE.

3.1. Problem instances

We have chosen to generate VRPHE instances by replacing the fixed costs f_k ($k = 1, \dots, K$) by variable costs α_k in the largest problem instances of Golden *et al.* More precisely, the travel cost d_{ij} between customer i and j is replaced by $d_{ijk} = \alpha_k d_{ij}$ when the travel is performed by a vehicle

of type k . The variable costs α_k have been chosen in such a way that no single vehicle type is much better or worse than the any of others (i.e. the good solutions of these problems, if considered as VFMs, are composed of vehicles of each type). The numbers n_k of vehicles of type k available have been chosen in such a way that: 1) The imposed fleet composition is very different from the fleet composition of good solutions to the VFM problems with variable costs only. 2) The total capacity of the imposed fleet exceeds the total order quantities by less than twice the capacity of the smallest vehicle types. Table 1 gives the characteristics of the problems we have generated. In this table, we give: the problem number, as given by Golden *et al.*, the number n of customers and, for each vehicle type k : the capacity Q_k , the fixed cost f_k , the variable cost α_k and the number n_k of vehicles available. For the VRPHE, the fixed costs f_k have to be set to 0 and for the VFM, the variable costs α_k have to be set to 1 and the number n_k of vehicles available to infinity for all k .

We justify the way we have generated VRPHE instances as follows: by replacing the fixed costs by variable costs, it is assumed that the main goal for these problem instances is to find a good set of tours that can be performed by a given fleet, and not to determine a good fleet (let us mention that, for the VFM instances, the fixed costs of good solutions are generally higher than the variable costs). Our choice of the variable costs α_k is justified by the fact that we wanted to compare VFM solution values (free fleet) to VRPHE solution values (imposed fleet); therefore no vehicle must be much better or worse than the others. Finally, we have chosen the imposed fleet composition so that the problem instances are relatively constrained; this will show that our method is able to find good solutions even with almost no alternative in the fleet composition.

As mentioned above, our method might be inappropriate for instances involving very few vehicles. For example, the homogeneous VRP induced by the problem instances 3 to 6 of Golden *et al.* (1984) with $n = 20$ are optimally solved by almost each call of the taboo search procedure of Taillard (1993) embedded in the algorithm for VRPHE (steps 1.1 and 1.3.3). Consequently, the set of tours remaining after step 4) is very limited and a good solution is not always found at step 5). Moreover, only the problems given with the coordinates of the customers can be solved with the taboo search algorithm used. Problem instances 1, 2 and 7 to 12 of Golden *et al.* are not of this type. Consequently, we have not considered these instances in our numerical results. Finally, the taboo search version embedded in the AMP does not take the number of vehicles used into account. This can

TABLE 1
Data for VFM instances and for VRPHE instances.

Problem number		Vehicle type																							
		A				B				C				D				E				F			
		Q_A	f_A	α_A	n_A	A_B	f_B	α_B	n_B	Q_C	f_C	α_C	n_C	Q_D	f_D	α_D	n_D	Q_E	f_E	α_E	n_E	Q_F	f_F	α_F	n_F
13	50	20	20	1.0	4	30	35	1.1	2	40	50	1.2	4	70	120	1.7	4	120	225	2.5	2	200	400	3.2	1
14	50	120	100	1.0	4	160	1500	1.1	2	300	3500	1.4	1												
15	50	50	100	1.0	4	100	250	1.6	3	160	450	2.0	2												
16	50	40	100	1.0	2	80	200	1.6	4	140	400	2.1	3												
17	75	50	25	1.0	4	120	80	1.2	4	200	150	1.5	2	350	320	1.8	1								
18	75	20	10	1.0	4	50	35	1.3	4	100	100	1.9	2	150	180	2.4	2	250	400	2.9	1	400	800	3.2	1
19	100	100	500	1.0	4	200	1200	1.4	3	300	2100	1.7	3												
20	100	60	100	1.0	6	140	300	1.7	4	200	500	2.0	3												

alter the quality of the solutions produced, especially for VFM instances for which the best fleet composition is homogeneous.

3.2. Numerical results for the VFM

The best solutions published for the VFM instances of Golden *et al.* may be found either in this reference, in Salhi and Rand (1993) or in Osman and Salhi (1994). We are going to compare the average performances of our heuristic column generation algorithm to the best solutions published in these references. All tests have been performed on Sun Sparc work stations (with processor TMS390Z55, 50 MHz) and the AMP was programmed in Pascal. We have solved each problem instances five times and all the results we give are averaged on these five runs. Table 2 gives numerical results concerning the VFM instances 13 to 20 of Golden *et al.* The following information is provided in the table:

- The problem number (as in Golden *et al.*).
- n , the number of customers.
- Best published: Best solution value published.
- New best known: Best solution value we have found during the elaboration of this paper.

TABLE 2
Numerical results for VFM instances of Golden *et al.*

Problem number	n	Best published	New best known	Average solution value	CPU AMS [s]	CPU CPLEX [s]	Total time [s]	m	Fleet
13	50	2437 ^a	2413.78	2436.78	469	0.47	470	366.2	1A 3B 2C 4F
14	50	9125.65 ^b	9119.03	9123.60	562	7.3	570	219.4	7A 1B
15	50	2600.31 ^b	2586.37	2593.61	331	2.2	334	230.0	10A 3B
16	50	2745.01 ^b	2741.50	2744.25	348	0.35	349	170.6	10B
17	75	1762.05 ^b	1747.24	1753.74	2055	17	2072	611.0	3A 2B 5C
18	75	2412.56 ^b	2373.63	2382.80	2678	65	2744	871.0	6B 8C 2D
19	100	8685.71 ^b	8661.81	8665.40	4692	7836	12528	932.6	15A
20	100	4166.03 ^b	4047.55	4063.18	1727	390	2117	946.6	13A 5B

^a Golden *et al.* (1984).

^b Osman and Salhi (1994).

- Average value: Average solution value found considering five different runs of the programme.
- CPU AMP: Average computation time (seconds) for the generation of the tours, corresponding to step 1) to 4) of the algorithm.
- CPU CPLEX: Average computation time of CPLEX runs, corresponding to step 5) of the algorithm.
- Total time: Average total computation time.
- m : Average number of tours generated by step 4) of the algorithm.
- Best fleet: fleet composition of the new best known solution.

In this table it is seen that the average solution values produced by the algorithm are always lower than the best solution values published and never more than one per cent above the best known solution. The worst individual solution found by a run of our method was less than 2% above the best known solution. In 40 runs, only six produced solutions worse than the best published solutions. For problems with $n \geq 75$, all runs have found solutions less than 1% above the best known solutions and all were better than the best solutions published. Therefore, it is concluded that our algorithm is very efficient and robust, at least for problems with a large number of tours. Naturally, the price for such a high quality is a computing time that can be significantly higher than that of heuristics producing solutions of lower quality.

We also see from this table that the computing times needed by CPLEX are generally very low as compared to the computing time of the AMP. However, as the problem size and number of tours generated grow, the times needed by CPLEX may become prohibitive, as is the case for problem number 19. To overcome this difficulty, we suggest limiting the number of tours included in the set T at step 2) of the algorithm. But, in our numerical experiments, we included in T all the tours produced by the AMP. When the number of tours kept was large enough, we observed that limiting the size of T did not significantly alter the quality of the solutions produced by the method. For example, for problem number 19, the quality of the solutions produced is not altered as soon as the best homogeneous VRP solution with vehicles of types A produced by the AMP is included in T . In fact, a good solution with vehicles of type A was found very early in the process: none of the five runs of the method took more than 34 seconds to find a solution better than the best previously published for problem number 19. This means that a different tuning of the parameters of the method may vastly reduce the computing time while producing acceptable solutions.

Finally, let us report few results concerning the problems instances 3 to 6 of Golden *et al.* with $n = 20$. Our method, with the same parameter settings as used above sometimes provides very bad results (15% above the best known solution), either due to a lack of diversity in the tour generated (the taboo searches return always the same solutions) or due to the fact that the first goal of the taboo searches is not to use as few vehicles as possible. By reducing the number of iterations performed by the taboo searches to $4n$ and by constraining the search to use a limited number of vehicles, we succeed in improving all the best published solutions of the problem instances 3 to 6 in 20 to 40 seconds of computing time. Namely, the new best known solution values (respectively the best previously published solution values) are: 961.03 (965), 6437.33 (6445.10), 1008.59 (1009.15) and 6516.47 (6516.56). The corresponding fleet compositions are 1A 2B 1C 2E, 6A, 3E and 6A. Since these results were not obtained with the “standard” parameter settings, they have not been included in Table 2.

3.3. Numerical results for the VRPHE

For the heterogeneous VRP with a fixed fleet, we give in Table 3 the same statistics as those given in Table 2 with the following modifications:

- The column of the best solution values published has been replaced by the best solution values we have found when the fleet composition is free (i.e. VFM with variable costs: $n_k = \infty$, $f_k = 0$, $k = 1, \dots, K$).
- The fleet composition is given in Table 1 and is not recalled here.
- m has the same value as in Table 2 since the homogeneous VRP solutions generated are identical and we have replaced this column in Table 3 by the average number of variables considered in CPLEX. As a tour is not always performed by the cheapest vehicle due to fleet restrictions, the number of variables in CPLEX is higher than m .
- The computation times for the AMP are the same as in Table 2 and are not recalled.

It is seen from this table that the average solution value are very close (max. 4%) above the “lower bound”, that is, the best solution found when the fleet composition is free. Again, the gaps between the best and worst runs are very low. This means that the method is robust. The computation times for CPLEX are acceptable even if the number of variables has been multiplied by a factor of 2 or 3 with respect to the VFM. For one instance, the computation time is even reduced. This may be explained by the fact that the problem is more constrained, and so the number of feasible solutions is

reduced. It is thus becoming easier to find an optimum assignment of tours of T for the fleet considered.

TABLE 3
Numerical results for VRPHE instances.

Problem number	n	Free fleet	Best known	Average value	Number of variables	CPU CPLEX [s]	Total time [s]
13	50	1494.58	1518.05	1536.55	1316.4	3	473
14	50	603.21	615.64	623.05	573.4	13	575
15	50	1007.35	1016.86	1022.05	501.0	3	335
16	50	1144.39	1154.05	1159.14	344.6	12	350
17	75	1044.93	1071.79	1095.01	1669.0	190	2245
18	75	1831.24	1870.16	1894.73	2969.0	198	2876
19	100	1110.96	1117.51	1156.93	2186.6	1141	5833
20	100	1550.36	1559.77	1592.16	2020.4	1775	3402

4. CONCLUSIONS

We have presented an efficient and robust heuristic method for the vehicle routing problem with a heterogeneous fleet of vehicles. This method is able to optimize either the fleet composition or the total distribution costs in the case of a fixed heterogeneous fleet. This latter problem has attracted much less attention in the OR literature. However it is a more realistic problem than the homogeneous VRP. We believe that our method could easily be adapted to solve real-life vehicle routing problems. The method may be efficiently parallelized with a relatively high number of processor.

ACKNOWLEDGEMENTS

This work was partly supported by an NSERC International Post-Doctoral Fellowship and by an NSERC Strategic Grant on *Parallel Software for Intelligent Vehicle-Highway System* during the stag of the author at the centre for research on transportation, University of Montreal, Canada.

REFERENCES

1. P. BADEAU, M. GENDREAU, F. GUERTIN, J.-Y. POTVIN, and É. D. TAILLARD, *A parallel tabu search heuristic for the vehicle routing problem with time windows*, Transportation Research-C5, 1997, pp. 109–122.

2. P. C. CHU and J. E. BEASLEY, *A Genetic Algorithm for the Set Partitioning Problem*, working paper, The Management School, Imperial College, London, 1995.
3. M. DESROCHERS and J. W. VERHOOG, *A new heuristic for the fleet size and mix vehicle routing problem*, *Computers and Operations Research*, 1991, 18, pp. 263-274.
4. F. G. GHEYSSENS, B. L. GOLDEN and A. ASSAD, *A comparison of techniques for solving the fleet size and mix vehicle routing problem*, *Operations Research Spektrum*, 1984, 6, pp. 207-216.
5. F. G. GHEYSSENS, B. L. GOLDEN and A. ASSAD, *A new heuristic for determining fleet size and composition*, *Mathematical Programming Studies*, 1986, 26, pp. 233-236.
6. B. GOLDEN, A. ASSAD, L. LEVY and F. G. GHEYSSENS, *The fleet size and mix vehicle routing problem*, *Computers and Operations Research*, 1984, 11, pp. 49-66.
7. I. H. OSMAN and S. SALHI, *Local search strategies for the Vehicle Fleet Mix Problem*, Rayward-Smith *et al.* (Editors), *Modern Heuristic Search Methods*, Wiley, 1996, pp. 131-153.
8. J.-Y. POTVIN and S. BENGIO, *The Vehicle Routing Problem with Time Windows – Part II: Genetic Search*, *Inform Journal on Computing* 8, 1996, pp. 165-172.
9. C. REGO and C. ROUCAIROL, *Parallel Tabu Search Algorithm Using Ejection Chains for the Vehicle Routing Problem*, in *Meta-heuristics*, Kelly and Osman, (Editors), Kluwer Academic Publishers, 1996, pp. 661-675.
10. Y. ROCHAT and F. SEMET, *A tabu search approach for delivering pet food and flour in Switzerland*, *Journal of the Operational Research Society*, 1994, 45, pp. 1233-1246.
11. Y. ROCHAT and É. D. TAILLARD, *Probabilistic diversification and intensification in local search for vehicle routing*, *Journal of Heuristics* 1, 1995, pp. 147-167.
12. S. SALHI, M. SARI and N. TOUATI, *Adaptation of some vehicle fleet mix heuristics*, *Omega*, 1992, 20, pp. 653-660.
13. S. SALHI and G. K. RAND, *Incorporating vehicle routing into the vehicle fleet composition problem*, *European Journal of Operational Research*, 1993, 66, pp. 313-330.
14. F. SEMET and É. D. TAILLARD, *Solving real-life vehicle routing problems efficiently using taboo search*, *Annals of Operations Research*, 1993, 41, pp. 469-488.
15. É. D. TAILLARD, *Parallel Iterative Search Methods for Vehicle Routing Problems*, *Networks* 23, 1993, pp. 661-676.
16. É. D. TAILLARD, *A diversification/intensification technique for local searches applied to vehicle routing problem*, internal report, 1994.
17. É. D. TAILLARD, G. LAPORTE and M. GENDREAU, *Vehicle routing with multiple use of vehicles*, *Journal of the Operations Research Society* 47, 1996, pp. 1065-1070.