Sociedad de Estadística e Investigación Operativa
Top (2002) Vol. 10, No. 2, pp. 211–237

Tabu Search Heuristics for the Vehicle Routing Problem with Time Windows

Olli Bräysy
SINTEF Applied Mathematics, Department of Optimization
P.O. Box 124 Blindern, N-0314 Oslo, Norway
email: Olli.Braysy@sintef.no
Michel Gendreau
Centre de Recherche sur les Transports, Université de Montréal,
Case postale 6128, Succursale "Centre-ville", Montréal, Canada H3C 3J7
email: michelg@crt.umontreal.ca

Abstract

This paper surveys the research on the Tabu Search heuristics for the Vehicle Routing Problem with Time Windows (VRPTW). The VRPTW can be described as the problem of designing least cost routes for a fleet of vehicles from one depot to a set of geographically scattered points. The routes must be designed in such a way that each point is visited only once by exactly one vehicle within a given time interval; all routes start and end at the depot, and the total demands of all points on one particular route must not exceed the capacity of the vehicle. In addition to describing basic features of each method, experimental results for Solomon's benchmark test problems are presented and analyzed.

Key Words: Metaheuristics, tabu search, vehicle routing, time windows.

AMS subject classification: 90B06, 68T20, 90-02.

1 Introduction

Transportation is an important domain of human activity. It supports and makes possible most other social and economic activities. Whenever we use a telephone, shop at our neighborhood foodstore or mall, read our mail or fly for business or pleasure, we are the beneficiaries of some system that has routed messages, goods or people from one place to another. Freight transportation, in particular, is one of today's most important activities, not

This work was partially supported by the Emil Aaltonen Foundation, Liikesivistysrahasto Foundation, the Canadian Natural Science and Engineering Research Council and the TOP program funded by the Research Council of Norway. This support is gratefully acknowledged.

Manuscript received: January 2002. Final version accepted: June 2002.

only measured by the yardstick of its own share of a nation's gross national product (GNP), but also by the increasing influence that the transportation and distribution of goods have on the performance of virtually all other economic sectors. Let us mention that the annual cost of excess travel in the United States has been estimated at some USD 45 billion (King and Mast (1997)) and the turnover of transportation of goods in Europe is some USD 168 billion per year. In the United Kingdom, France and Denmark, for example, transportation represents some 15%, 9% and 15% of national expenditures respectively (Crainic and Laporte (1997), Larsen (1999)). It is estimated that distribution costs account for almost half of the total logistics costs and in some industries, such as in the food and drink business, distribution costs can account for up to 70the value added costs of goods (De Backer et al. (1997), Golden and Wasil (1987)). Halse (1992) reports that in 1989, 76.5% of all the transportation of goods was done by vehicles, which also underlines the importance of routing and scheduling problems.

The Vehicle Routing Problem with Time Windows (VRPTW) is an important problem occurring in many distribution systems. VRPTW can be described as the problem of designing least cost routes for a fleet of identical vehicles from one depot to a set of geographically scattered points. The routes must be designed in such a way that each point is visited only once by exactly one vehicle within a given time interval, all routes start and end at the depot, and the total demands of all points on one particular route must not exceed the capacity of the vehicle. In addition, it is assumed that each vehicle covers just one route. The VRPTW has multiple objectives in that the goal is to minimize not only the number of vehicles required, but also the total travel time and total traveled distance incurred by the fleet of vehicles. Some of the most useful applications of the VRPTW include bank deliveries (Lambert et al. (1993)), postal deliveries (Mechti et al. (2001)), industrial refuse collection (Golden et al. (2001)), national franchise restaurant services (Russell (1995)), school bus routing (Bracca et al. (1994)), industrial gases delivery (Campbell et al. (2001)) and JIT (just in time) manufacturing (Vaidyanathan et al. (1999)).

The VRPTW has been the subject of intensive research efforts for both heuristic and exact optimization approaches. Early surveys of solution techniques for the VRPTW can be found in Golden and Assad (1986), Desrochers et al. (1988), Golden and Assad (1988), and Solomon and Desrosiers (1988). Desrosiers et al. (1995) and Cordeau et al. (2001a) mostly focus on exact techniques. Further details on these exact methods can be found in Larsen (1999) and Cook and Rich (1999). For recent survey on genetic and evolutionary algorithms for the VRPTW, see Bräysy and Gendreau (2001b). In addition to the basic VRPTW, i.e., the capacitated Vehicle Routing Problem (VRP) with time window constraints that is the topic of this paper, there has been a lot of research also on other time constrained vehicle routing problems. Nanry and Barnes (2000) present a reactive tabu search for Pickup and Delivery Problem with Time Windows (PDPTW), where each transportation request specifies a single origin and a single destination, instead of just transporting between depot and peripheral locations. Duhamel et al. (1997) describe a tabu search heuristic for VRP with backhauls and time windows that is a special case of PDPTW where the deliveries precede pickups. Sigurd et al. (2000) extend PDPTW by considering also precedence constraints among customers. Campbell et al. (2001) consider an inventory routing problem with time windows, where inventory management at customer sites is combined with routing planning. Mechti et al. (2001) developed a tabu search for real-life mail collecting application, where the vehicles in the fleet are not identical, i.e., the Fleet Size and Mix VRP with Time Windows. Cordeau et al. (2001b) consider a VRPTW with multiple depots, and a Periodic VRPTW, and Lund et al. (1996) study a dynamic VRPTW, where a subset of customers and demands are unknown beforehand. For more details on these variations, we refer to Bräysy et al. (2002).

Because of the high complexity level of the VRPTW and its wide applicability to real-life situations, solution techniques capable of producing high-quality solutions in limited time, i.e., heuristics are of prime importance. Over the last few years, many authors have proposed new heuristic approaches, mostly metaheuristics, for tackling the VRPTW. So far tabu searches have showed the best performance in tackling the problem. To our knowledge, these have not been comprehensively surveyed and compared. The purpose of this survey is to fill this gap. The remainder of this paper is organized as follows. In Section 2, we recall the formulation of the problem as an integer program. The tabu search heuristics are reviewed in Section 3 and experimental results are presented and analyzed in Section 4. Section 5 concludes the paper.

2 Problem formulation

The VRPTW is defined on a graph (N, A). The node set N consist of the set of customers, denoted by C, and the nodes 0 and $n+1$, which represent the depot. The number of customers $|C|$ will be denoted n and the customers will be denoted by $1, 2, \ldots, n$. The arc set A corresponds to possible connections between the nodes. No arc terminates at node 0 and no arc originates at node $n+1$. All routes start at 0 and end at $n+1$. A cost c_{ij} and travel time t_{ij} are associated with each arc $(i, j) \in A$ of the network. The travel time t_{ij} includes a service time at customer i. The set of identical vehicles is denoted by V. Each vehicle has a given capacity q and each customer a demand d_i, $i \in C$. At each customer, the start of the service must be within a given time interval, called a time window, $[a_i, b_i]$, $i \in C$. Vehicles must also leave the depot within the time window $[a_0, b_0]$ and return during the time window $[a_{n+1}, b_{n+1}]$. A vehicle is permitted to arrive before the opening of the time window, and wait at no cost until service becomes possible, but it is not permitted to arrive after the latest time window. Since waiting time is permitted at no cost, we may assume without loss of generality that $a_0 = b_0 = 0$; that is, all routes start at time 0.

The model contains two types of decision variables. The decision variable X_{ij}^k (defined $\forall (i, j) \in A$, $\forall k \in V$) is at equal to 1 if vehicle k drives from node i to node j, and 0 otherwise. The decision variable S_i^k (defined $\forall i \in N$, $\forall k \in V$ denotes the time vehicle k, $k \in V$ starts service at customer i, $i \in C$. If vehicle k does not service customer i, S_i^k has no meaning. We may assume that $S_0^k = 0$, $\forall k$ and S_{n+1}^k denotes the arrival time of vehicle k at the depot. The objective is to design a set of minimal cost routes, one for each vehicle, such that all customers are serviced exactly once. Hence, split deliveries are not allowed. The routes must be feasible with respect to the capacity of the vehicles and the time windows of the customers serviced. The VRPTW can be stated mathematically as:

$$\min \sum_{k \in V} \sum_{(i,j) \in A} c_{ij} X_{ij}^k \tag{2.1}$$

subject to

$$\sum_{k \in V} \sum_{j \in N} X_{ij}^k = 1, \qquad \forall i \in C \tag{2.2}$$

$$\sum_{i \in C} d_i \sum_{j \in N} X_{ij}^k \leq q, \qquad \forall k \in V \tag{2.3}$$

$$\sum_{j \in N} X_{oj}^k = 1, \qquad \forall k \in V \tag{2.4}$$

$$\sum_{i \in N} X_{ih}^k - \sum_{j \in N} X_{hj}^k = 0, \qquad \forall h \in C, \forall k \in V \tag{2.5}$$

$$\sum_{i \in N} X_{i,n+1}^k = 1, \qquad \forall k \in V \tag{2.6}$$

$$X_{ij}^k (S_i^k + t_{ij} - S_j^k) \leq 0, \qquad \forall (i,j) \in A, \forall k \in V \tag{2.7}$$

$$a_i \leq S_i^k \leq b_i, \qquad \forall i \in N, \forall k \in V \tag{2.8}$$

$$X_{ij}^k \in \{0, 1\}, \qquad \forall (i,j) \in A, k \in V \tag{2.9}$$

The objective function (2.1) states that costs should be minimized. Constraint set (2.2) states that each customer must be assigned to exactly one vehicle, and constraint set (2.3) states that no vehicle can service more customers than its capacity permits. Constraint sets (2.4), (2.5) and (2.6) are the flow constraints requiring that each vehicle k leaves node 0 once, leaves node h, $h \in C$ if and only if it enters that node, and returns to node $n+1$. Note that constraint set (2.6) is redundant, but is maintained in the model to underline the network structure. The arc $(0, n+1)$ is included in the network, to allow empty tours. More precisely, we permit an unrestricted number of vehicles, but a cost c_v is put on each vehicle performing an empty tour, i.e., to each vehicle not used. This is done by setting $c_{0,n+1} = -c_v$ That is, the more there are empty tours, the lower is the total cost. The value of c_v is sufficiently large to primarily minimize the number of vehicles and secondarily minimize travel costs. Nonlinear (easily linearized, see for example Desrosiers et al. (1995)) constraint set (2.7) states that vehicle k cannot arrive at j before $S_i^k + t_{ij}$ if it travels from i to j. Constraint set (2.8) ensures that all time windows are respected and (2.9) is the set of integrality constraints.

3 Tabu search algorithms

Tabu Search (TS) is a local search metaheuristic introduced by Glover (1986). Details about tabu search can also be found in Glover (1989), Glover (1990), Hertz et al. (1997) and Glover and Laguna (1997). TS explores the solution space by moving at each iteration from a solution s to the best solution in a subset of its neighborhood $N(s)$. Contrary to classical descent methods, the current solution may deteriorate from one iteration to the next. Thus, to avoid cycling, solutions possessing some attributes of recently explored solutions are temporarily declared tabu or forbidden. The duration that an attribute remains tabu is called tabu-tenure and it can vary over different intervals of time and the tabu status can be overridden if certain conditions are met. This is called the aspiration criterion and it happens for example when a tabu solution is better than any previously seen solution. Also, various techniques are often employed to diversify or to intensify the search process.

Garcia et al. (1994) describe a tabu search heuristic where the neighborhood is restricted to the exchange of arcs that are close in distance. The initial solution is created using Solomon's (1987) I1 insertion heuristic, and the algorithm oscillates between 2-opt* (Potvin and Rousseau (1995)) and Or-opt (Or (1976)) exchanges. When one has not made any improvement for a certain number of iterations, the other improvement operator is used and vice versa. In order to minimize the number of routes, the algorithm tries to move customers from routes with a few customers into other routes using Or-opt exchanges. The parallel implementation is performed by partitioning the neighborhood among slave processors. The master processor is then used to guide the tabu search. After exploration of the neighborhood, the best move from each processor is sent to the master. Potvin et al. (1996) present a similar tabu search approach for one processor, using more computational power to improve solutions.

Bachem et al. (1996) describe an improvement heuristic based on the mechanisms of trading. The partition of customers into the tours is determined by finding matches in a leveled bipartite graph that the authors call a "trading graph". The nodes correspond to either an insertion (buy) of a customer into a tour or a deletion (sell). The edges represent possible exchanges and the weight of each edge is the gain that is obtained by the corresponding action. Thus, every matching of the trading graph corresponds to a number of interchanges of customers. In each iteration, tours are shuffled by choosing some permutation at random. Then for each tour either a sell or buy action is selected and finally possible trading matches are evaluated and the best one selected. The approach allows infeasibilities against certain penalty factors, as well as trading matchings with negative weights causing deterioration. Because of this deterioration a tabu list is also added to prevent cycling. The approach was implemented using two different kinds of parallelizations. In the first approach, each tour was mapped into one processor, which makes sell or buy decision. To reduce the idle time of the processors, the second approach partitions the current tour plan such that each processor gets about the same number of different tours.

Rochat and Taillard (1995) propose a tabu search approach based on a so-called "adaptive memory" and 2-opt local search (for details, see Bräysy and Gendreau (2001a)). The adaptive memory is a pool of routes taken from the best solutions visited during the search. Its purpose is to provide new starting solutions for the tabu search through selection and combination of routes extracted from the memory. In the first phase, tabu search is used to create a number of different solutions and these solutions are then stored into the adaptive memory. The selection of routes from the memory is done probabilistically and the probability of selecting a particular route depends on the value of the solution the route belongs to. The selected tours are improved using tabu search and inserted subsequently back into adaptive memory. At the end, a set partitioning problem is solved exactly using the routes in the pool to create the best possible solution.

Taillard et al. (1997) propose a tabu search heuristic for the vehicle routing problem with soft time windows. In this problem, lateness at customer locations is allowed although a penalty is incurred and added to the objective value. By adding large penalty values, the vehicle routing problem with hard time windows can be addressed as well. The authors propose a new exchange heuristic called CROSS- exchange. First, the two edges $(i-1, i)$, and $(k, k+1)$ are removed from the first route while the edges $(j-1, j)$ and $(l, l+1)$ are removed from the second route. Then the segments $i-k$ and $j-l$, which may contain an arbitrary number of customers, are swapped by introducing the new edges $(i-1, j)$, $(l, k+1)$, $(j-1, i)$ and $(k, l+1)$. 2-opt* and Or-opt are special cases of this operator and CROSS-exchange is also a special case of λ-interchange (Osman, (1993)). CROSS-exchange is illustrated in Figure 1.

The proposed local search-operator is used both to exchange customers between routes and for intra- route optimization. Solomon's insertion heuristic I1 with random parameters is first used to fill the adaptive memory (Rochat and Taillard (1995)) with different types of routes. The solution is then decomposed into a disjoint subset of routes by using the polar angle

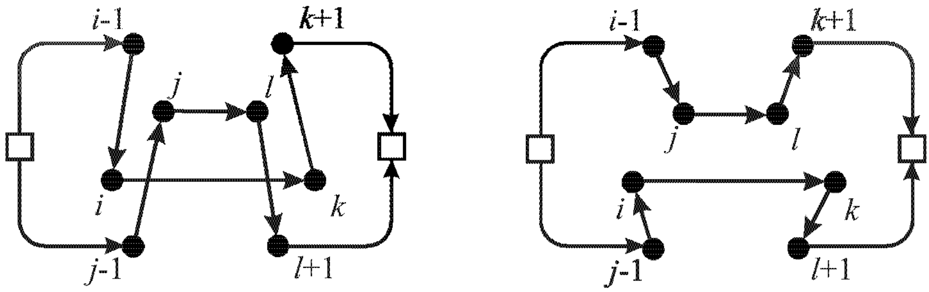

Figure 1: *CROSS-exchange. Segments (i,k) on the upper route and (j,l) on the lower route are simultaneously reinserted into lower and upper routes respectively. This is performed by replacing edges $(i-1,i)$, $(k,k+1)$, $(j-1,j)$ and $(l,l+1)$ by edges $(i-1,j)$, $(l,k+1)$, $(j-1,i)$ and $(k,l+1)$. Note that the orientation of both routes is preserved.*

associated with the center of gravity of each route, and tabu search is used to solve each subset separately. A complete solution is reconstructed by merging the new routes found by tabu search. Decomposition, tabu search and reconstruction are repeated for certain number of iterations. The algorithm penalizes frequently performed exchanges to diversify the search and reorders the customers within the best routes using Solomon's I1 insertion heuristic. Moreover, an adaptation of the GENIUS heuristic (Gendreau et al. (1992)) for time windows is applied to each individual route of final solution. GENIUS is an extension of the relocate neighborhood in which a customer can also be inserted between the two customers nodes on the destination route that are nearest to it, even if these customer nodes are not consecutive. The operator is illustrated in Figure 2.

Badeau et al. (1997) study the same problem solving framework as Taillard et al. (1997) using a 2- level parallel implementation that combines the so-called master-slave scheme with an allocation of each subproblem to a different processor. In this master-slave scheme, the master process manages the adaptive memory and generates solutions from it; these solutions are then transmitted to slave processes that improve them by performing tabu search and return the best solutions found to the master. The authors conclude that parallelization of the original sequential approach does not degrade solution quality, for the same amount of computations, while

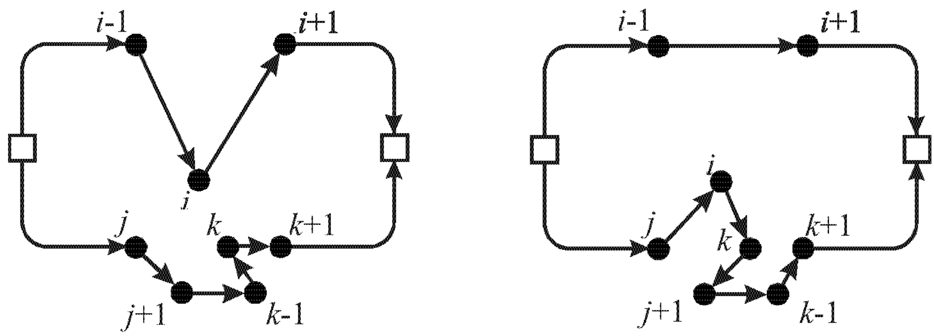

Figure 2: *The GENIUS-exchange operator. Customer i on the upper route is inserted into the lower route between the customers j and k closest to it by adding the edges (j, i) and (i, k). Since j and k are not consecutive, one has to reorder the lower route. Here the feasible tour is obtained by deleting edges $(j, j+1)$ and $(k-1, k)$ and by relocating the path $j+1, \ldots, k-1$.*

providing substantial speed-ups. No detailed results are presented in the paper.

Carlton (1995) describes a reactive tabu search that dynamically adjusts its parameter values based on the current search status. More precisely, the size of the tabu list is managed by increasing the tabu list size if identical solutions occur too often and reducing it if no feasible solution can be found. This approach is applied to several types of problems with time windows. Its robustness comes from a simple neighborhood structure, which can be easily adapted to different problems. Namely, each customer is removed and reinserted at some other location in the current solution.

Chiang and Russell (1997) develop a reactive tabu search that dynamically varies the size of the list of forbidden moves to avoid cycles as well as an overly constrained search path. More precisely, this is done by increasing the tabu list size, if identical solutions occur too often and reducing the size if no feasible solution can be found. The underlying local search is based on the λ-interchange mechanism of Osman (1993). The λ-exchange generation mechanism can be described as follows. Given a solution for the problem represented by a set of routes $S = \{r_1, \ldots, r_p, \ldots, r_q, \ldots, r_k\}$, a λ-interchange between a pair of routes (r_p, r_q) is a replacement of a subset of customers $S_1 \subseteq r_p$ of size $|S_1| \leq \lambda$ by another subset $S_2 \subseteq r_q$ of size

$|S_2| \leq \lambda$ to get two new routes $r_p^* = (r_p - S_1) \cup S_2,$, $r_q^* = (r_q - S_2) \cup S_1$ and a new neighboring solution $S' = \{r_, \ldots, r_p^*, \ldots, r_q^*, \ldots, r_k\}$. The neighborhood $N_\lambda(S)$ of a given solution S is the set of all neighbors $S\prime$ generated in this way for a given value of λ.

Here $(\lambda = 1)$and the tabu search is applied to the parallel construction approach of Russell (1995) that incorporates improvement procedures during the construction process and builds several routes simultaneously. The frequency-based diversification strategy penalizes the customer nodes that switch too frequently. The intensification strategy is designed to reduce the waiting time at each customer. This is specifically achieved by forbidding certain customers from moving into another route.

De Backer and Furnon (1997) describe a classical two-phase mechanism to solve TSP, VRP and VRPTW. The initial solution is first generated using the savings heuristic of Clarke and Wright (1964). This solution is subsequently optimized using two intra-route local searches (2-opt and Or-opt) and three inter-route operators (cross, exchange, relocate - for details, see Bräysy and Gendreau (2001a)) guided with tabu search. Here recently removed or inserted arcs are unauthorized, i.e., tabu for a given amount of time. The approach is coupled to a Constraint Programming framework, where feasibility of the new solution is checked through constraint propagation.

Brandão (1999) describes a tabu search algorithm allowing infeasible solutions during the search process. The initial solution is created with a simple sequential cheapest insertion algorithm. Here only unrouted customers that are close to customers already inserted into the route are considered for insertion. The created solution is improved by inserting randomly selected customers into other routes or into another place within the same route, or by swapping randomly selected customers between two routes. After each successful insertion or swap, the routes with a new customer are reordered using a modified version of the GENI algorithm of Gendreau et al. (1992). A table of frequencies is maintained to count the number of times that each customer has been moved from one route to another. In the later part of the search, only those customers whose frequency is below the average are considered.

Schulze and Fahle (1999) propose a tabu search performing several search threads in parallel. Each thread is started with a different initial solution and a neighboring solution is generated by performing a sequence of simple customer shifts (ejection chain: Glover (1991) and Glover (1992)). All routes generated by the tabu search heuristic are collected in a pool. At the termination of local optimization steps, the worst solution is replaced by a new one created by solving the set covering problem on the routes in the pool with Lagrangian relaxation based heuristic of Beasley (1990). With this new set of solutions, the whole process is restarted until a certain stopping criterion is fulfilled. In addition, the proposed method tries to eliminate routes having at most three customers by trying to move these customers into other routes. The routing of customers supplied by the same vehicle is improved by performing Or-opt exchanges within the route and the search is diversified by penalizing frequently performed customer shifts. To generate an appropriate number of initial solutions, three different heuristics are used, namely Solomon's (1987) I1 insertion heuristic, the parallel route building heuristic of Potvin and Rousseau (1993) and a modified version of the Savings heuristic of Clarke and Wright (1964). In the parallel implementation, each processor handles a set of solutions instead of just one and solves also the set covering problem separately on these solutions to avoid idle times. Each time a processor terminates its local optimization process, the routes of the optimized solutions are sent to all other processors to enable sharing of knowledge.

Gehring and Homberger (1999) and Gehring and Homberger (2001) study a two-phase approach, where the tabu search is combined with evolutionary algorithm ES1 of Homberger and Gehring (1999). In this evolutionary algorithm the search is mainly driven by mutation based on Or-opt (Or, 1976), 2-opt* (Potvin and Rousseau, 1995) and λ-interchange moves (Osman, 1993) with $\lambda = 1$. In addition, a special Or-opt based operator is used to reduce the number of routes. The individuals of a starting population are generated by means of a stochastic approach that is based on the savings algorithm of Clarke and Wright (1964). The evolutionary algorithm is used in the first phase to minimize the number of routes. In the second phase, the total distance is minimized using a tabu search algorithm utilizing the same local search operators. The approach is parallelized using the concept of cooperative autonomy, i.e., several autonomous sequential solution procedures cooperate through the exchange of solutions. The cooperating slave processes are configured in different ways using different seeds for random number generators to create diversity in the search.

Tan et al. (2000) describe a tabu search algorithm based on λ-interchanges with the best-accept strategy. The initial solution is created with cheapest insertion heuristics of Thangiah et al. (1994). Each time a local minimum is found, the search is diversified by performing a series of random λ-interchange hops combined with 2-opt* operator. A candidate list is maintained to record elite solutions discovered during the search process. These elite solutions are then used as a starting point for intensification.

Cordeau et al. (2001b) introduce a simple tabu search procedure for VRPTW and two of its extensions, namely Periodic VRPTW and Multi-depot VRPTW. An important feature of the approach is the possibility of exploring infeasible solutions during the search. The violations of constraints are penalized in the cost function and the parameter values regarding each type of violation (load, duration and time windows constraints) are adjusted dynamically. The local search operator used is a simple insertion, which inserts one customer at a time in another location. To diversify the search, frequently appearing customer-route combinations are penalized. The initial solution is constructed by sorting customers in increasing order of the angle they make with the depot and an arbitrary radius. Then customers are inserted in this order into the first route that can incorporate the customer so as to minimize the increase in the total travel time of the route. The best solution identified after n iterations is post-optimized by applying to each individual route a specialized heuristic for the Traveling Salesman Problem with Time Windows (Gendreau et al. (1998)).

Lau et al. (2000) present a generic constraint-based diversification technique to enhance a tabu search algorithm for VRPTW. An initial solution is first created with a simple greedy insertion algorithm. Then a local optimal solution is generated by tabu search using exchange and relocate operators with the best-accept strategy. Using a strategy similar to Large Neighborhood Search by Shaw (1998), a part of this solution is extracted and passed to the constraint-based local search. Here VRPTW is modeled as a linear constraint satisfaction problem that is solved by a simple local search algorithm performing local flips. In the first stage, the time window and capacity constraints are relaxed hence allowing the search to explore infeasible regions. In the second stage, the relaxed constraints are resumed and the search continues until a new feasible solution is obtained. This feasible solution is then passed back to tabu search and the whole process is repeated up to a preset number of times.

In addition to above described approaches, the basic idea of tabu search, i.e., using short-term memory to prevent cycling is used is many other meta-

Authors	**Year**	**Initial solution**	**Neighborhood Operators**	Notes **Notes**
Garcia et al.	1994	Solomon's I1 heuristic	2-opt*, Or-opt	Neighborhood restricted to arcs close in distance
Rochat and Taillard	1995	Tabu search	2-opt	Adaptive memory
Carlton	1995	Insertion heuristic	Relocate	Reactive tabu search
Potvin et al.	1996	Solomon's I1 heuristic	2-opt*, Or-opt	Neighborhood restricted to arcs close in distance
Bachem et al.	1996	Not described	Insertion, deletion of a customer from tour	Simulates mechanisms of trading, infeasibilities allowed
Taillard et al.	1997	Solomon's I1 heuristic	CROSS	Soft time windows, adaptive memory
Badeau et al.	1997	Solomon's I1 heuristic	CROSS	Soft time windows, adaptive memory
Chiang and Russell	1997	Modification of Russell (1995)	λ-interchange	Reactive tabu search
De Backer and Furnon	1997	Savings heuristic	Exchange, relocate, 2-opt*, 2-opt, Or-opt	Constraint programming used to check feasibility of moves
Brandão	1999	Insertion heuristic	Relocate, exchange, GENI	Neighborhoods restricted to arcs close in distance
Schulze and Fahle	1999	Solomon's I1, parallel I1 and savings heuristic	Ejection chains, Or-opt	Generated routes stored in a pool
Tan et al.	2000	Insertion heuristic of Thangiah (1994)	λ-interchange, 2-opt*	–
Lau et al.	2000	Insertion heuristic	Exchange, relocate	Constraint based diversification
Cordeau et al.	2001	Modification of Sweep heuristic	Relocate, GENI	Infeasibilities allowed during the search
Gehring and Homberger	1999, 2001	Savings heuristic	Or-opt, 2-opt*, λ-interchange	Tabu Search hybridized with an evolutionary algorithm

Table 1: *The main features of tabu search heuristics for VRPTW.*

heuristic approaches as well. For instance, Thangiah et al. (1994) combine simulated annealing (Metropolis et al. (1953)) with tabu search. Also Li et al. (2001) use tabu search within simulated annealing, and De Backer et al. (2000) test different combinations of guided local search (Voudouris (1997)) and tabu search. Wee Kit et al. (2001) propose a hybrid genetic algorithm (Holland (1975)), where a simple tabu search based on cross, exchange, relocate and 2-opt neighborhoods is applied on individual solutions in the later generations to intensify the search. The main features of the described tabu search heuristics are summarized in Table 1, where we present the authors, year, initial solution heuristics, neighborhood operators used, as well as some notes.

4 Analysis of results

The described tabu searches are compared and analyzed here using the results obtained for Solomon's (1987) well-known 56 benchmark problems. These problems have a hundred customers, a central depot, capacity constraints, time windows on the time of delivery, and a total route time constraint. The C1 and C2 classes have customers located in clusters and in the R1 and R2 classes the customers are at random positions. The RC1 and RC2 classes contain a mix of both random and clustered customers. Each class contains between 8 and 12 individual problem instances and all problems in any one class have the same customer locations, and the same vehicle capacities; only the time windows differ. In terms of time window density (the percentage of customers with time windows), the problems have 25%, 50%, 75%, and 100% time windows. The C1, R1 and RC1 problems have a short scheduling horizon, and require 9 to 19 vehicles. In short horizon problems vehicles have small capacities and short route times, and cannot service many customers at one time. Classes C2, R2 and RC2 are more representative of "long-haul" delivery with longer scheduling horizons and fewer (2-4) vehicles. Both travel time and distance are given by the Euclidean distance between points.

The results are usually ranked according to a hierarchical objective function, where the number of vehicles is considered as the primary objective, and for the same number of vehicles, the secondary objective is often either total traveled distance or total duration of routes. Therefore, a solution requiring fewer routes is always considered better than solution with more routes, regardless of the total traveled distance. The results obtained with the described tabu search heuristics are presented in Tables 2 and 3. The upper part of Table 2 depicts the best results attained with the tabu searches without paying attention to the computational effort. The results are compared against the results of the best recent metaheuristics by Gambardella et al. (1999), Homberger and Gehring (1999), Bräysy (2001b) and Berger et al. (2001), described in the lower part of Table 2.

Gambardella et al. (1999) developed an ant colony system consisting of two separate ant colonies with different objectives, guiding well-known nearest neighbor heuristic for solution construction, and CROSS-exchanges of Taillard et al. (1997) for improvement. Homberger and Gehring (1999) introduced two evolutionary metaheuristics, where the search is mainly driven by mutation, based on random selection between well-known standard neighborhoods (see page 13). Also in the parallel genetic algorithm of Berger et al. (2001) the search is mainly performed by a set of six different mutation techniques based on standard construction and improvement heuristics, such as Solomon (1987), λ-exchanges of Osman (1993), relocate (Savelsbergh, 1992), Large Neighborhood Search by Shaw (1998), and the above described ant colony system of Gambardella et al. (1999). Bräysy (2001b) presented a new four-phase deterministic metaheuristic based on Variable Neighborhood Search of Mladenovic and Hansen (1997), and several new improvement heuristics based on CROSS-exchanges of Taillard et al. (1997), Large Neighborhood Search by Shaw (1998) and Ejection Chains of Glover (1991 and 1992).

In Table 2, the first column to the left describes the authors, and columns R1, R2, C1, C2, RC1 and RC2 present the average number of routes and average total distance with respect to six problem groups of Solomon (1987). Finally, the rightmost column indicates the cumulative number of vehicles and cumulative total distance over all 56 test problems. Due to lack of exact information, we cannot consider all algorithms here.

Even though Brandão (1999) uses rounded distances during the execution of the algorithm, we believe that the differences in final solutions remain small and the results are therefore comparable. To our knowledge, only De Backer and Furnon (1997) and Bräysy (2001b) proposed deterministic method. All other procedures in Table 2 are stochastic, i.e., with each run one gets typically different results. Here one must note that the tabu search framework itself is deterministic. The stochastic nature of the

Author	R1	R2	C1	C2	RC1	RC2	CNV/ CTD
Garcia et al. (1994)	**12.92** 1317.7	**3.09** 1222.6	**10.00** 877.1	**3.00** 602.3	**12.88** 1473.5	**3.75** 1527.0	**436** 65977
Rochat et al. (1995)	**12.25** 1208.50	**2.91** 961.72	**10.00** 828.38	**3.00** 589.86	**11.88** 1377.39	**3.38** 1119.59	**415** 57231
Potvin et al. (1996)	**12.50** 1294.50	**3.09** 1154.4	**10.00** 850.2	**3.00** 594.6	**12.63** 1456.3	**3.38** 1404.8	**426** 63530
Bachem et al. (1996)	**12.58** 1392.0	**3.00** 1199.6	–	–	**12.13** 1501.6	**3.38** 1500.1	–
Taillard et al. (1997)	**12.17** 1209.35	**2.82** 980.27	**10.00** 828.38	**3.00** 589.86	**11.50** 1389.22	**3.38** 1117.44	**410** 57523
Chiang et al. (1997)	**12.17** 1204.19	**2.73** 986.32	**10.00** 828.38	**3.00** 591.42	**11.88** 1397.44	**3.25** 1229.54	**411** 58502
De Backer et al. (1997)	**14.17** 1214.86	**5.27** 930.18	**10.00** 829.77	**3.25** 604.84	**14.25** 1385.12	**6.25** 1099.96	**508** 56998
Brandão (1999)	**12.58** 1205	**3.18** 995	**10.00** 829	**3.00** 591	**12.13** 1371	**3.50** 1250	**425** 58562
Schulze et al. (1999)	**12.25** 1239.15	**2.82** 1066.68	**10.00** 828.94	**3.00** 589.93	**11.75** 1409.26	**3.38** 1286.05	**414** 60346
Gehring et al. (1999)	**12.42** 1198	**2.82** 947	**10.00** 829	**3.00** 590	**11.88** 1356	**3.25** 1140	**415** 56942
Tan et al. (2000)	**13.83** 1266.37	**3.82** 1080.24	**10.00** 870.87	**3.25** 634.85	**13.63** 1458.16	**4.25** 1293.38	**467** 62008
Lau et al. (2000)	**14.00** 1211.54	**3.55** 960.43	**10.00** 832.13	**3.00** 612.25	**13.63** 1385.05	**4.25** 1232.65	**464** 58432
Cordeau et al. (2001b)	**12.08** 1210.14	**2.73** 969.57	**10.00** 828.38	**3.00** 589.86	**11.50** 1389.78	**3.25** 1134.52	**407** 57556
Gehring et al. (2001)	**12.00** 1217.57	**2.73** 961.29	**10.00** 828.63	**3.00** 590.33	**11.50** 1395.13	**3.25** 1139.37	**406** 57641
Gambardella et al. (1999)	**12.00** 1217.73	**2.73** 967.75	**10.00** 828.38	**3.00** 589.86	**11.63** 1382.42	**3.25** 1129.19	**407** 57525
Homberger et al. (1999)	**11.92** 1228.06	**2.73** 969.95	**10.00** 828.38	**3.00** 589.86	**11.63** 1392.57	**3.25** 1144.43	**406** 57876
Bräysy (2001b)	**11.92** 1222.12	**2.73** 975.12	**10.00** 828.38	**3.00** 589.86	**11.50** 1389.58	**3.25** 1128.38	**405** 57710
Berger et al. (2001)	**11.92** 1221.10	**2.73** 975.43	**10.00** 828.48	**3.00** 589.93	**11.50** 1389.89	**3.25** 1159.37	**405** 57952

Table 2: *Comparison of tabu search algorithms. For each algorithm the average results with respect to Solomon's benchmarks are depicted. The notations CNV and CTD in the rightmost column indicate the cumulative number of vehicles and cumulative total distance over all 56 test problems.*

tabu search heuristics described here is due to different random components included by authors such as randomly chosen parameter values or neighborhoods, randomly inserted customers or random moves to neighboring solutions. All methods consider the number of vehicles as the primary optimization criterion. The only exceptions are approaches by De Backer and Furnon (1997) and Tan et al. (2001) that concentrate only on minimization of distance. The second objective is total duration in Garcia et al. (1994), Potvin et al. (1996) and Schulze et al. (1999). The other procedures use the total distance of routes as the second objective. The cumulative total distance values in Tables 2 and 3 are rounded to integers due to usage of rounded distance measures reported by other authors for calculation.

If one considers only tabu search heuristics, according to Table 2 the approaches of Gehring and Homberger (2001) and Cordeau et al. (2001b) seem to produce the best results in terms of solution quality. However, the difference with regard to other well-performing tabu searches by Taillard et al. (1997) and Chiang and Russell (1997) is only about 1% in number of vehicles. Also regarding the total traveled distance the differences between the three methods are small: The differences in CTD remain within 2%. The greatest differences can be found in problem group RC2, in which hybrid of evolutionary algorithm and tabu search by Gehring and Homberger (2001) is about 8% better than the method of Chiang and Russell (1997).

If one analyzes these four methods showing the best performance, it seems that in addition to short- term tabu list to prevent cycling, they all memorize also the long term search history, and use the gathered information to diversify and intensify the search. In addition, all methods include different random choices to further diversify the search. The initial solution is constructed with well-known construction heuristics, and the neighborhood considered in the improvement phase is small in all cases. Two of the methods create and maintain a population of several solutions, and also enable interaction between these solutions during the search. Finally, two of the best four methods use a special post-optimization technique in the end.

In conclusion, it seems that one can obtain very good results by hybridizing tabu search with standard construction and improvement heuristics described in the literature (see Bräysy and Gendreau (2001a) for details). In this regard it seems to be important to use a long term memory based strategies to guide the search. As Table 2 presents the best results obtained during all computational experiments, one cannot conclude anything regarding the efficiency of the different approaches here. Further discussion is provided later based on the results presented in Table 3.

The algorithms De Backer and Furnon (1997) and Tan et al. (2000) seem to give the worst results with respect to the number of vehicles. The reason for this can be found in the optimization criteria used. De Backer and Furnon (1997) and Tan et al. (2000) consider only the total traveled distance, while the other procedures minimize the number of vehicles first. However, in spite of this difference in objective function, the method by De Backer and Furnon (1997) yields better outcomes than the other approaches in terms of total distance only for problem groups R2 and RC2.

The difference in cumulative number of vehicles between the best and worst approaches considering the number of vehicles as the primary objective by Bräysy (2001b) and Lau et al. (2000), respectively, is even 14%. In our opinion this difference is significant and in terms of total distance the differences are even greater. For example the difference between approaches of Garcia et al. (1994) and Gehring and Homberger (1999) in CTD is about 16% and the difference between Garcia et al. (1994) and Gehring and Homberger (2001) in problem group RC2 is about 36%. By comparing the results produced by different tabu search heuristics with the results of the best recent metaheuristics, described in the lower part of Table 2, it can be seen that the performance of recent tabu searches is excellent. The CNV reported by Cordeau et al. (2001b) and Gehring and Homberger (2001) is within 0.5% from the lowest known value, 405, and in terms of total distance, these tabu searches are even better than the other best approaches. As far as the computational burden is concerned, conclusions are very difficult to draw, since most of the authors do not report the CPU time consumption or the number of runs used to obtain the results in Table 2.

To be able to analyze the computational effort, we consider in Table 3 only results for which the time consumption and the number of runs are described by the authors. To facilitate the comparison, the effect of different hardware is normalized to equal Sun Sparc 10 using Dongarra's (1998) factors. In addition, if the reported results are the best ones over multiple experiments, we multiplied the computation times by this number to see the real computational effort. The computer, number of independent runs, and the CPU time used to obtain the reported results are described in the lower part of Table 3. The number of runs is greater than one only if the reported result is best over multiple executions of the given algorithm and the CPU time is reported only for a single run. Two CPU time values are described: the one reported by the authors and in the parenthesis the modified CPU time. Details for calculating these modified CPU times can be found in Bräysy (2001a).

According to Table 3, the algorithms by Homberger and Gehring (1999), Gehring and Homberger (1999 and 2001) and Bräysy (2001b) show the best overall performance. Regarding individual problem groups, it seems that practically all approaches yield excellent results to problem groups C1 and C2, having customers located in geographical clusters. Bräysy (2001b) gives the best output to R1, RC1 and RC2 on average, though the difference to the best competing approaches is very small, less than 0.5%. Gehring and Homberger (2001) report the best results to R2 on average, though again the differences to the best competing approaches by Homberger and Gehring (1999) and Bräysy (2001b) are very small. Also in general, the differences between recent results, reported in year 2001, are small, varying within 2%. For detailed best-known results to Solomon's benchmarks, we refer to http://www.top.sintef.no/. As for the efficiency of the described approaches, it seems that substantial computational effort is required to obtain the good results reported in Gehring and Homberger (2001). On the other hand, Gehring and Homberger (1999) report rather good results, obtained with small CPU time. In general, apart from Gehring and Homberger (2001) and Potvin et al. (1996) that is much faster than the other approaches, there are not significant differences in time consumption between the approaches described in Table 3. In conclusion, it seems that the evolutionary algorithm of Homberger and Gehring (1999) and the reactive variable neighborhood search of Bräysy (2001b) show slightly better performance than the recent tabu searches, when one considers both solution quality and time consumption.

It seems that there are four common features among the algorithms showing best performance in Table 3 that one should consider when designing a solution technique for the VRPTW. First, the neighborhood size is small in all methods. The reason for this is of course that it would be much more time consuming to explore larger neighborhoods that can be prohibitive in real-life context. However, as a consequence, the output is often heavily dependent on the initial solution, even in case of tabu searches as it is very time consuming to do significant modifications to the structure of the solution, when the neighborhood size is small. Therefore, to over-

Author	R1	R2	C1	C2	RC1	RC2	CNV/ CTD
(1) Rochat et al. (1995)	**12.58** 1197.4 2	**3.09** 954.36	**10.00** 828.45	**3.00** 590.32	**12.38** 1369.48	**3.62** 1139.79	**427** 57120
(2) Potvin et al. (1996)	**12.58** 1294.7	**3.09** 1185.9	**10.00** 861.0	**3.00** 602.5	**12.63** 1465.0	**3.38** 1476.1	**427** 64679
(3) Taillard et al. (1997)	**12.33** 1220.35	**3.00** 1013.35	**10.00** 828.4 5	**3.00** 590.91	**11.90** 1381.31	**3.38** 1198.63	**417** 58614
(4) Brandão (1999)	**12.58** 1205	**3.18** 995	**10.00** 829	**3.00** 591	**12.13** 1371	**3.50** 1250	**425** 58562
(5) Schulze et al. (1999)	**12.50** 1268.42	**3.09** 1055.90	**10.00** 828.94	**3.00** 589.93	**12.25** 1396.07	**3.38** 1308.31	**423** 60651
(6) Gehring et al. (1999)	**12.42** 1198	**2.82** 947	**10.00** 829	**3.00** 590	**11.88** 1356	**3.25** 1140	**415** 56942
(7) Gehring et al. (2001)	**12.00** 1217.57	**2.73** 961.29	**10.00** 828.6 3	**3.00** 590.33	**11.50** 1395.13	**3.25** 1139.37	**406** 57641
(8) Homberger et al. (1999)	**11.92** 1228.06	**2.73** 969.95	**10.00** 828.3 8	**3.00** 589.86	**11.63** 1392.57	**3.25** 1144.43	**406** 57876
(9) Gambardella et al. (1999)	**12.38** 1210.83	**3.00** 960.31	**10.00** 828.38	**3.00** 591.85	**11.92** 1388.13	**3.33** 1149.28	**418** 57583
(10) Bräysy (2001b)	**11.92** 1222.12	**2.73** 975.12	**10.00** 828.38	**3.00** 589.86	**11.50** 1389.58	**3.25** 1128.38	**405** 57710
(11) Berger et al. (2001)	**12.17** 1251.40	**2.73** 1056.59	**10.00** 828.5 0	**3.00** 590.06	**11.88** 1414.86	**3.25** 1258.15	**411** 60200

(1) Silicon Graphics 100 MHz, 1 run, 92.2 (138) min., (2) Sun Sparc 10, 1 run, 9.8 (9.8) min., (3) Sun Sparc 10, 1 run, 248 (248) min., (4) Pentium 200, 4 runs, 38.9 (373) min., (5) Motorola PowerPc 604, 5 runs, 8.3 (270) min., (6) 4-Pentium 200 MHz, 1 run, 5 (48) min., (7) 4-Pentium 400 MHz, 5 runs, 15.1 (1631) min., (8) Pentium 200 MHz, 10 runs, 13 (312) min., (9) Sun Ultrasparc 1, 1 run 30 (210) min., (10) Pentium 200 MHz, 1 run, 82.5 (198) min., (11) Pentium 400 MHz, 1 run, 30 (162) min.

Table 3: *Comparison of results obtained with limited computational effort for Solomon's benchmark problems.*

come the problem, all the best performing methods in Table 3 create a set of several initial solutions in the beginning, and improve them either independently or by interacting between the solutions. For creating these initial solutions, the authors use slight modifications of standard route construction heuristics (for details, see Bräysy and Gendreau (2001a)). Finally, all the best performing methods use special operators or strategies for reducing the number of routes that is considered as the primary objective.

5 Conclusions

NP-hardness of the vehicle routing problem requires heuristic solution strategies for most real-life instances. In the previous sections, we have comprehensively surveyed tabu search approaches for the VRPTW. Currently algorithms by Gehring and Homberger (2001), Cordeau et al. (2001b), Taillard et al. (1997) and Chiang and Russell (1997) show the best performance. In the comparison with the current best approaches, recent tabu searches were found to be very competitive. Thus one can conclude that tabu search heuristics are clearly among the best techniques to tackle the vehicle routing problem with time windows. It is our hope that this paper has provided valuable insights for the pursuit of solutions to many current and future challenging problems.

References

Bachem A., Hochstättler W. and Malich M. (1996). The Simulated Trading Heuristic for Solving Vehicle Routing Problems. *Discrete Applied Mathematics* 65, 47-72.

Badeau P., Gendreau M., Guertin F., Potvin J.-Y. and Taillard E. (1997). A Parallel Tabu Search Heuristic for the Vehicle Routing Problem with Time Windows. *Transportation Research C* 5, 109-122.

Beasley J.E. (1990). A Lagrangian Heuristic for Set Covering Problems. *Naval Research Logistics* 37, 151-164.

Berger J., Barkaoui M. and Bräysy O. (2001). A Route-directed Hybrid Genetic Approach for the Vehicle Routing Problem with Time Windows. Working Paper, Defence Research Establishment Valcartier, Canada.

Bracca J., Bramel J. and Simchi-Levi D. (1994). A Computerized Approach to the New York City School Bus Routing Problem. *IEE Transactions* 29, 693-702.

Brandão J. (1999). Metaheuristic for the Vehicle Routing Problem with Time Windows. In: Voss S., Martello S., Osman I.H. and Roucairol C. (eds.), *Metaheuristics Advances and Trends in Local Search Paradigms for Optimization.* Kluwer Academic Publishers, 19-36.

Bräysy O. (2001a). Local Search and Variable Neighborhood Search Algorithms for the Vehicle Routing Problem with Time Windows. Ph.D. Thesis, University of Vaasa, Finland.

Bräysy O. (2001b). A Reactive Variable Neighbourhood Search for the Vehicle Routing Problem with Time Windows. *INFORMS Journal on Computing* (to appear).

Bräysy O. and Gendreau M. (2001a). Route Construction and Local Search Algorithms for the Vehicle Routing Problem with Time Windows. Internal Report STF42 A01024, SINTEF Applied Mathematics, Department of Optimization, Norway.

Bräysy O. and Gendreau M. (2001b). Genetic Algorithms for the Vehicle Routing Problem with Time Windows. Internal Report STF42 A01021, SINTEF Applied Mathematics, Department of Optimization, Norway.

Bräysy O., Gendreau M., Hasle G. and Løkketangen A. (2002). A Survey of Rich Vehicle Routing Models and Heuristic Solution Techniques. Working Paper, SINTEF Applied Mathematics, Department of Optimization, Norway.

Campbell A., Clarke L. and Savelsbergh M. (2001). Inventory Routing in Practice. In: Toth P. and Vigo D. (eds.), *The Vehicle Routing Problem.* SIAM, 309-330.

Carlton W.B. (1995). A Tabu Search Approach to the General Vehicle Routing Problem. Ph.D. Thesis, University of Texas, Austin, U.S.A.

Chiang W.C. and Russell R.A. (1997). A Reactive Tabu Search Metaheuristic for the Vehicle Routing Problem with Time Windows. *INFORMS Journal on Computing* 9, 417-430.

Clarke G. and Wright J.W. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Operations Research* 12, 568-581.

Cook W. and Rich J.L. (1999). A Parallel Cutting-Plane Algorithm for the Vehicle Routing Problems with Time Windows. Working Paper, Department of Computational and Applied Mathematics, Rice University, Houston.

Cordeau J.F., Desaulniers G., Desrosiers J., Solomon M.M. and Soumis F. (2001a). The VRP with Time Windows. In: Toth P. and Vigo D. (eds.), *The Vehicle Routing Problem.* SIAM, 157-193.

Cordeau J.-F., Laporte G. and Mercier A. (2001b). A Unified Tabu Search Heuristic for Vehicle Routing Problems with Time Windows. *Journal of the Operational Research Society* 52, 928-936.

Crainic T.G. and Laporte G. (1997). Planning Models for Freight Transportation. *European Journal of Operational Research* 97, 409-438.

De Backer B. and Furnon V. (1997). Meta-heuristics in Constraint Programming Experiments with Tabu Search on the Vehicle Routing Problem. Second International Conference on Metaheuristics (MIC'97), Sophia Antipolis, France.

De Backer B., Furnon V., Kilby P., Prosser P. and Shaw P. (2000). Solving Vehicle Routing Problems Using Constraint Programming and Metaheuristics. *Journal of Heuristics* 6, 501-523.

Desrochers M., Lenstra J.K., Savelsbergh M.W.P. and Soumis F. (1988). Vehicle Routing with Time Windows: Optimization and Approximation. In: Golden B. and Assad A. (eds.), *Vehicle Routing: Methods and Studies.* Elsevier Science Publishers, 65-84.

Desrosiers J., Dumas Y., Solomon M.M. and Soumis F. (1995). Time Constrained Routing and Scheduling. In: Ball M.O., Magnanti T.L., Monma C.L. and Nemhauser G.L. (eds.), *Handbooks in Operations Research and Management Science 8: Network Routing.* Elsevier Science Publishers, 35-139.

Dongarra J. (1998). Performance of Various Computers Using Standard Linear Equations Software. Report CS-89-85, Department of Computer Science, University of Tennessee, U.S.A.

Duhamel C., Potvin J.-Y. and Rousseau J.-M. (1997). A Tabu Search Heuristic for the Vehicle Routing Problem with Backhauls and Time Windows. *Transportation Science* 31, 49-59.

Gambardella L.M., Taillard E. and Agazzi G. (1999). MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows. In: Corne D., Dorigo M. and Glover F. (eds.), *New Ideas in Optimization.* McGraw-Hill, 63-76.

García B.-L., Potvin J.-Y. and Rousseau J.-M. (1994). A Parallel Implementation of the Tabu Search Heuristic for Vehicle Routing Problems with Time Window Constraints. *Computers and Operations Research* 21, 1025-1033.

Gehring H. and Homberger J. (1999). A Parallel Hybrid Evolutionary Metaheuristic for the Vehicle Routing Problem with Time Windows. In: Miettinen K., Mäkelä M. and Toivanen J. (eds.), *Proceedings of EUROGEN99 - Short Course on Evolutionary Algorithms in Engineering and Computer Science, Reports of the Department of Mathematical Information Technology*, Series A. Collections, No. A 2/1999. University of Jyväskylä 57-64.

Gehring H. and Homberger J. (2001). Parallelization of a Two-Phase Metaheuristic for Routing Problems with Time Windows. *Asia-Pacific Journal of Operational Research* 18, 35-47.

Gendreau M., Hertz A. and Laporte G. (1992). A New Insertion and Postoptimization Procedure for the Traveling Salesman Problem. *Operations Research* 40, 1086-1093.

Gendreau M., Hertz A., Laporte G. and Stan M. (1998). A Generalized Insertion Heuristic for the Traveling Salesman Problem with Time Windows. *Operations Research* 46, 330-335.

Glover F. (1986). Future Paths for Integer Programming and Links to Artificial Intelligence. *Computers and Operations Research* 13, 533-549.

Glover F. (1989). Tabu Search Ű Part I. *Journal on Computing* 1, 190-206.

Glover F. (1990). Tabu Search Ű Part II. *Journal on Computing* 2, 4-32.

Glover F. (1991). Multilevel Tabu Search and Embedded Search Neighborhoods for the Traveling Salesman Problem. Working Paper, College of Business and Administration, University of Colorado, U.S.A.

Glover F. (1992). New Ejection Chain and Alternating Path Methods for Traveling Salesman Problems. In: Balci O., Sharda R. and Zenios S. (eds.), *Computer Science and Operations Research: New Developments in Their Interfaces.* Pergamon Press, 449-509.

Glover F. and Laguna M. (1997). *Tabu Search.* Kluwer Academic Publishers.

Golden B.L. and Assad A.A. (1986). Perspectives on Vehicle Routing: Exciting New Developments. *Operations Research* 34, 803-809.

Golden B.L. and Wasil E.A. (1987). Computerized Vehicle Routing in the Soft Drink Industry. *Operations Research* 35, 6-17.

Golden B.L. and Assad A.A. (1988). *Vehicle Routing: Methods and Studies.* Elsevier Science Publishers.

Golden B.L., Assad A.A. and Wasil E.A. (2001). Routing Vehicles in the Real World: Applications in the Solid Waste, Beverage, Food, Dairy and Newspaper Industries. In: Toth P. and Vigo D. (eds.), *The Vehicle Routing Problem.* SIAM, 245-286.

Halse K. (1992). Modeling and Solving Complex Vehicle Routing Problems. Ph.D. Thesis, Institute of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark.

Hertz A., Taillard E. and De Werra D. (1997). Tabu Search. In: Aarts E. and Lenstra J.K. (eds.), *Local Search in Combinatorial Optimization.* John Wiley, 121-136.

Holland J.H. (1975). *Adaptation in Natural and Artificial Systems.* University of Michigan Press.

Homberger J. and Gehring H. (1999). Two Evolutionary Meta-heuristics for the Vehicle Routing Problem with Time Windows. *INFOR* 37, 297-318.

King G.F. and Mast C.F. (1997). Excess Travel: Causes, Extent and Consequences. *Transportation Research Record* 1111, 126-134.

Lambert V., Laporte G. and Louveaux F. (1993). Designing Collection Routes through Bank Branches. *Computers and Operations Research* 20, 783-791.

Larsen J. (1999). Parallelization of the Vehicle Routing Problem with Time Windows. Ph.D. Thesis, Institute of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark.

Lau H.C., Lim Y.F. and Liu Q. (2000). Diversification of Neighborhood via Constraint-Based Local Search and its Application to VRPTW. Working paper, School of Computing, National University of Singapore.

Li H., Lim A. and Huang J. (2001). Local Search with Annealing-like Restarts to Solve the VRPTW. Working paper, Department of Computer Science, National University of Singapore.

Lund K., Madsen O.B.G. and Rygaard J.M. (1996). Vehicle Routing Problems with Varying Degree of Dynamism. Technical Report, IMM, The department of Mathematical Modelling, Technical University of Denmark.

Mechti R., Poujade S., Roucairol C. and Lemarié B. (2001). Global and Local Moves in Tabu Search: A Real-Life Mail Collecting Application. Manuscript, PriSM laboratory, University of Versailles, France.

Metropolis W., Rosenbluth A., Rosenbluth M., Teller A. and Teller E. (1953). Equation of the State Calculations by Fast Computing Machines. *Journal of Chemical Physics* 21, 1087-1092.

Mladenovic N. and Hansen P. (1997). Variable Neighborhood Search. *Computers and Operations Research* 24, 1097-1100.

Nanry W.P. and Barnes J.W. (2000). Solving the Pickup and Delivery Problem with Time Windows Using Reactive Tabu Search. *Transportation Research B* 34, 107-121.

Or I. (1976). Traveling Salesman-Type Combinatorial Problems and Their Relation to the Logistics of Regional Blood Banking. Ph.D. Thesis, Northwestern University, Evanston, Illinois.

Osman I.H. (1993). Metastrategy Simulated Annealing and Tabu Search Algorithms for the Vehicle Routing Problems. *Annals of Operations Research* 41, 421-452.

Potvin J.-Y. and Rousseau J.-M. (1993). A Parallel Route Building Algorithm for the Vehicle Routing and Scheduling Problem with Time Windows. *European*

Journal of Operational Research 66, 331-340.

Potvin J.-Y. and Rousseau J.-M. (1995). An Exchange Heuristic for Routing Problems with Time Windows. *Journal of the Operational Research Society* 46, 1433-1446.

Potvin J.-Y., Kervahut T., Garcia B.L. and Rousseau J.-M. (1996). The Vehicle Routing Problem with Time Windows Part I: Tabu Search. *INFORMS Journal on Computing* 8, 157-164.

Rochat Y. and Taillard E. (1995). Probabilistic Diversification and Intensification in Local Search for Vehicle Routing. *Journal of Heuristics* 1, 147-167.

Russell R.A. (1995). Hybrid Heuristics for the Vehicle Routing Problem with Time Windows. *Transportation Science* 29, 156-166.

Savelsbergh, M.W.P. (1992). The Vehicle Routing Problem with Time Windows: Minimizing Route Duration. *Journal on Computing* 4, 146-154.

Schulze J. and Fahle T. (1999). A Parallel Algorithm for the Vehicle Routing Problem with Time Window Constraints. *Annals of Operations Research* 86, 585-607.

Shaw P. (1998). Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems. In: Maher M. and Puget J.-F. (eds.), *Principles and Practice of Constraint Programming - CP98, Lecture Notes in Computer Science.* Springer-Verlag, 417-431.

Sigurd M., Pisinger D. and Sig M. (2000). The Pickup and Delivery Problem with Time Windows and Precedences. Technical Report TR-00/08, Department of Computer Science, University of Copenhagen, Denmark.

Solomon M.M. (1987). Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Operations Research* 35, 254-265.

Solomon M.M. and Desrosiers J. (1988). Time Window Constrained Routing and Scheduling Problems. *Transportation Science* 22, 1-13.

Taillard E., Badeau P., Gendreau M., Guertin F. and Potvin J.-Y. (1997). A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows. *Transportation Science* 31, 170-186.

Tan K.C., Lee L.H. and Zhu K.Q. (2000). Heuristic Methods for Vehicle Routing Problem with Time Windows. *Proceedings of the 6th International Symposium on Artificial Intelligence and Mathematics.* Ft. Lauderdale, Florida.

Thangiah S., Osman I.H. and Sun T. (1994). Hybrid Genetic Algorithm, Simulated Annealing and Tabu Search Methods for Vehicle Routing Problems with Time Windows. Working Paper UKC/IMS/OR94/4, Institute of Mathematics and Statistics, University of Kent, Canterbury.

Vaidyanathan B.S, Matson J.O., Miller D.M. and Matson J.E. (1999). A Capacitated Vehicle Routing Problem for Just-In-Time Delivery. *IEE Transactions* 31, 1083-1092.

Voudouris C. (1997). Guided Local Search for Combinatorial Problems. Ph.D. Thesis, Department of Computer Science, University of Essex, Colchester, UK.

Wee Kit H., Chin J. and Lim A. (2001). A Hybrid Search Algorithm for the Vehicle Routing Problem with Time Windows. *International Journal on Artificial Intelligence Tools* (to appear).