Journal of the Eastern Asia Society for Transportation Studies, Vol. 8, 2009

Column Generation-based Heuristics for Vehicle Routing Problem with Soft Time Windows

Ali Gul QURESHI
JSPS Post Doctoral Fellow
Graduate School of Engineering
Kyoto University
C-1, Kyotodaigaku, Katsura Campus,
Nishikyo-ku, Kyoto
615-8540, Japan
Fax: +81-75-950-3800
E-mail: aligul@kiban.kuciv.kyoto-u.ac.jp

Eiichi TANIGUCHI
Professor
Graduate School of Engineering
Kyoto University
C-1, Kyotodaigaku, Katsura Campus,
Nishikyo-ku, Kyoto
615-8540, Japan
Fax: +81-75-950-3800
E-mail: taniguchi@kiban.kuciv.kyoto-u.ac.jp

Tadashi YAMADA
Associate Professor
Graduate School of Engineering
Kyoto University
C-1, Kyotodaigaku, Katsura Campus,
Nishikyo-ku, Kyoto
615-8540, Japan
Fax: +81-75-950-3800
E-mail: t.yamada@kiban.kuciv.kyoto-u.ac.jp

Abstract: This paper presents a column generation-based heuristics for the Vehicle Routing and scheduling Problem with Soft Time Windows (VRPSTW). The subproblem has been solved using a modified stochastic push forward insertion heuristics that incorporates the early and late arrival penalties. The useful dual information (shadow prices) from the column generation master problem guides the heuristic subproblem to provide negative reduced cost columns of sufficient quality. The performance of column generation-based heuristics is evaluated comparing its results with a genetic algorithms heuristics whose initial population is based on the same insertion heuristics as used in the column generation subproblem. The results showed that the column generation-based heuristics produced better quality solutions, both in terms of cost and environment (CO_2), in most cases in only quarter of the computation time, on average.

Key Words: *City logistics, Vehicle routing, Column generation heuristics*

1. INTRODUCTION

A major proportion of the total domestic frieght movement is based on road transport mode in many countries. Whereas, almost all of the intra-urban frieght movement (both pickup and deliveries) is carried out by trucks and/or vans. Door to door service industries such as home applience or utilities repair services, are also considered in the broader defination of the urban logistics. Transportion cost shares a considerable proportion in the prices of such services and many other products. With increase in the competation and innovations in technologies, frieght companies try to introduce many consumer satisfaction driven policies, such as deliveries within a pre-specified time period or *time window*. City logistics deals with the optimization of logistics and transport activities within an urban area by considering costs and benefits for both public and private sectors, not only in terms of delivery cost but in terms of environment, traffic congestion and energy use; while still being within the framework of market economy (Thompson and Taniguchi, 2001).

Route optimization is one of the city logistics-related measures, which not only serves the logistics firms, it also advances the objectives of other stakeholders of city logistics (for example, city administration and residents) as it minimizes the number of delivery vehicles and their traveled distance thus help reducing congestion, emissions and safety problems as well. The Vehicle Routing and scheduling Problem with Time Windows (VRPTW) is a commonly adopted route optimization technique, which consists of finding a set of minimum cost routes (for delivery vehicles) to cover the demands (weights to be picked up or delivered) of all customers within their specified time windows [a_i, b_i]. In exact optimization field, often these time windows are treated *hard*, which means deliveries can not be done after the end of time windows b_i. Furthermore, if a vehicle arrives earlier it has to wait (without associated cost) until the start of time windows a_i. On the contrary, most of the city logistic-related literature is based on a more practical variant, i.e. the Vehicle Routing and scheduling Problem with Soft Time Windows (VRPSTW) that allows the delivery after b_i at some penalty costs and also penalizes the waiting time in the case of early arrival. It has been observed that penalizing the waiting time (the VRPSTW case) results in less waiting time as compared to the hard time windows case (Qureshi *et al.,* 2007).

The column generation-based algorithms have been very popular in the exact optimization field for the Vehicle Routing and scheduling Problem with Hard Time Windows (VRPHTW) (for example, see Desrochers *et al.,* 1992; Kohl *et al.,* 1997; Irnich and Villeneuve, 2003; Feillet *et al.,* 2004; Chabrier, 2006). These algorithms have not only improved the size of the problems solved enormously, they also have significantly reduced the computational efforts required for the hard time windows variant. Complex soft time windows constraints and time dependent costs have been the greatest barriers in the development of exact solution procedures for the VRPSTW. However, there have been few attempts in this regard (for example, see Tagmouti *et al.,* 2007) but the excessive computation time severely limits the size of the problem. Therefore, majority of the literature in city logistics (reviewed in next section) relies on heuristics solutions for the VRPSTW, such as Genetic Algorithms (GA).

This paper presents a column generation-based heuristics for the VRPSTW in order to improve the solution quality and to reduce computational times as compared to the available VRPSTW heuristics. The basic idea is to replace the NP-hard subproblem of column generation scheme i.e. the Elementary Shortest Path Problem with Resource Constraint (ESPPRC) by an insertion heuristics, which can handle complex soft time windows constraints efficiently. The shadow prices generated by the column generation method are utilized to find the reduced cost, and the customer insertion criterion is computed based on this reduced cost. The rest of the paper is structured as follows: section 2 provides a literature review of related research while the section 3 formulates the VRPSTW as a set partitioning formulation commonly adopted for column generation schemes, and its subproblem. The complete column generation-based heuristics is described in section 4. This section also defines the insertion heuristics subproblem. Section 5 outlines a GA heuristics used for the comparison with the column generation-based heuristics. Section 6 describes the implementation, test instances and discusses the results obtained on these instances with the two heuristics. Finally, section 7 draws some conclusions and gives some future research prospects.

2. LITERATURE REVIEW

While bulk of the research targeting the soft time windows is heuristics based, the exact solution research incorporating soft time windows has been focused on the schedule optimization of a given fixed path considering linear (Sexton and Bodin, 1985) and/or generalized convex penalty functions (Dumas *et al.,* 1990). Recently, Tagmouti *et al.* (2007) have presented an arc routing problem with soft time windows, where vehicles are not allowed to wait along their routes. In their column generation scheme, they have used a modified labeling algorithm for the Shortest Path Problem with Time Windows and Time Costs (SPPTWTC) subproblem earlier given by Ioachim *et al.* (1998). The vehicle arrival pattern has been represented by a continuous variable that resulted in very high computation times and limited the maximum size of problem solved to 40 customers.

On the heuristics side, Balakrishnan (1993) described three simple sequential insertion heuristics for the VRPSTW based on the nearest neighbor, Clarke-Wright savings and space-time rules. An early arrived vehicle has options of either start servicing the customer earlier than a_i at some penalty or wait till a_i without any cost. Hashimoto *et al.* (2006) used local search and dynamic programming in a route first schedule second algorithm to solve the VRPTW with soft time windows and variable travel time costs. First, routes are optimized using a local search algorithm and then the optimum service start time for each customer is found as a subproblem. Duin *et al.* (2007) used the VRPSTW model and solved it with a tabu search for each freight carrier in a framework of hybrid freight market, where a fraction of demands is pre-allotted and another fraction is available in a real time auctioning system. The freight carrier providing best bid to carry the extra demands wins the auction.

Genetic algorithms (GA) are often employed in solving complex and close to real life VRPSTW instances in city logistics; for example, Taniguchi and Heijden (2000) used GA solutions of the VRPSTW to evaluate many city logistics measures such as cooperative delivery systems (CDS) and load factor control. Taniguchi *et al.* (2001) used a GA to solve a Probabilistic VRPSTW (VRPSTW-P) that incorporates the uncertainties of travel times on a road network. Yamada *et al.* (2004) used a similar GA approach for VRPSTW-P to study the travel time reliability of a road network. Utilizing the VICS (Vehicle Identification and Communication System) data and the data from 66 days operation of probe pickup/delivery trucks, Ando and Taniguchi (2007) have applied the VRPSTW-P and its GA solution to an actual delivery system in Osaka, Japan. Yamada *et al.* (2001) combined the logistics terminal location, CDS and VRPSTW in a single framework. The combined model was solved using a GA heuristics and the results were compared to a base case that did not use the CDS. Qureshi and Hanaoka (2005) studied a CDS using GA solutions of the associated VRPSTW, where a truck assigning module assigns consolidated routes back to trucks of participating companies.

Many researchers in the field of the VRPSTW research have tried to use set partitioning linear optimization. Rochat and Taillard (1995) used a heuristic approach by first generating many candidate routes using intensified and diversified tabu search and then used set partitioning linear programming. Calvete *et al.* (2007) exploited goal programming to enumerate all the feasible routes and then used a set partitioning problem to solve the VRP with soft time windows with heterogeneous fleet and multi objectives. In a similar approach, Fagerholt (2001) solved a ship-scheduling problem with soft time windows. He used the Traveling Salesman Problem with Capacity, Hard Time Windows and Precedence Constraint (TSP-CHTWPC) to enumerate all feasible routes and then optimizes their schedule using soft time windows, before using a set partitioning problem.

All above-cited references used set partitioning linear programming after enumerating some or all possible candidate routes for the VRPSTW, while this study utilizes the useful dual information, i.e. shadow prices, obtained every time a set partitioning linear program is solved. The set partitioning master problem and the heuristic subproblem are solved in cycles. Shadow prices are generated in master problem that guide the optimization in the heuristic subproblem, which in return provides the routes with negative reduced costs to augment the set partitioning linear program. A similar approach has been adopted for a dynamic VRPHTW by Chen and Xu (2006).

Alvarenga *et al.* (2007) used a specialized genetic algorithm for VRPHTW to generate routes to be optimized with set partitioning formulation at the end of the algorithm. Dana and Le Pape (2005) presented a somewhat different approach in their branch and price heuristic for the VRPHTW, using a cooperation scheme between the classic column generation (using ESSPRC as subproblem), a MIP solver and some local search schemes. The main feature is to find a better integer solution (Global Upper Bound) earlier by MIP and local search to accelerate the classic branch and price method. An excellent review of the heuristic methods applied to the VRPHTW and the VRPSTW can be found in Braysy and Gendreau (2005a; 2005b).

3. SET PARTITIONING MODEL FORMULATION OF VRPSTW

The VRPSTW is defined on a directed graph $G = (V, A)$. The vertex set V includes the depot vertex 0 and set of customers $C = \{1, 2, . . ., n\}$. The arc set A consists of all feasible arcs (i, j), $i, j \in V$. Both cost c_{ij} as well as time t_{ij} are associated with each arc $(i, j) \in A$. Time t_{ij} includes the travel time on arc (i, j) and the service time at vertex i, and a fixed vehicle utilization cost is added to all outgoing arcs from the depot, i.e. in c_{0j}, $j \in C$. With every vertex of V there is an associated demand d_i, with $d_0 = 0$, and a time window $[a_i, b_i]$ representing the earliest and the latest possible service start times.

The Dantzig-Wolfe decomposition or commonly known column generation scheme formulates the VRPSTW as a set partitioning problem (Eq. (1)-Eq. (3)) and into an Elementary Shortest Path Problem with Resource Constraint (ESPPRC) as a subproblem.

$$\min \sum_{p \in P} c_p y_p \tag{1}$$

subject to

$$\sum_{p \in P} a_{ip} y_p = 1 \qquad \forall i \in C \tag{2}$$

$$y_p \in \{0,1\} \qquad \forall p \in P \tag{3}$$

The role of the subproblem is to provide feasible single vehicle routes (also called paths or columns) to the linear program (LP) of the set partitioning master problem. Set P is the set of all such feasible routes. The variable y_p takes value 1 if the route $p \in P$, is selected and 0 otherwise. The cost of route p is denoted by c_p, and a_{ip} represents the number of times route p serves customer i. The objective (Eq. (1)) selects a minimum cost set of routes from P such that each customer $i \in C$ is serviced exactly once as per the constraint represented by Eq. (2). The integrality constraint is represented by Eq. (3); usually a linear programming relaxation of this constraint is solved where y_p can vary anywhere between 0 and 1. In an exact solution algorithm, if the problem at hand satisfies the cost triangular inequality $c_{ij} \leq c_{ih} + c_{hj}$, the solution possesses an integrality property which means that any optimal solution of the linear relaxation of the given set partitioning problem will have all y_p values either 1 or 0. The cost structure of the VRPSTW not always follows the cost triangular inequality; therefore, a heuristics approach has been developed to get an integer solution (as described in §4.2).

As the number of all feasible single vehicle routes in a VRPSTW instance may be very large, in each column generation iteration the set P is dynamically augmented using routes generated by the subproblem. The Elementary Shortest Path Problem with Resource Constraint (ESPPRC) subproblem is solved on the same graph as the original VRPSTW problem, but with reduced costs $\overline{c'_{ij}}$ on the arcs given by Eq. (4). The vector $\boldsymbol{\pi}$ represents the dual variable values (prices or shadow costs) for each customer $i \in C$ obtained from the master problem. In the VRPSTW, the arc costs c'_{ij} depend on the service start time at customer s'_j (Figure 1); these costs are calculated as per Eq. (5), where c_e and c_l are unit penalty costs for early and late arrival, respectively.

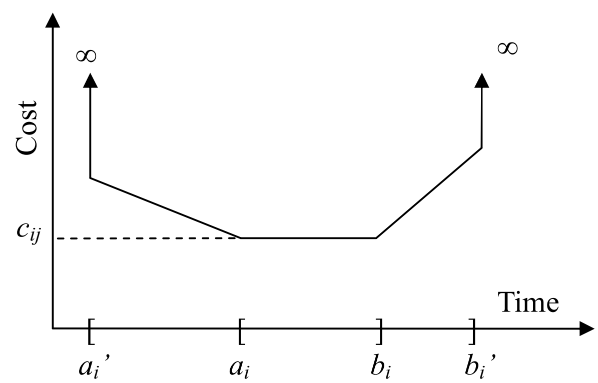

Figure 1 Cost function in the VRPSTW

$$\overline{c'_{ij}} = c'_{ij} - \pi_i \qquad \forall i \in C \tag{4}$$

$$c'_{ij} = \begin{cases} c_{ij}, \text{if } \mathrm{a_j} \leq s'_j \leq b_j \\ c_{ij} + c_l(s'_j - b_j), \text{if } b_j < s'_j \leq b'_j \\ c_{ij} + c_e(a_j - s'_j), \text{if } a'_j \leq s'_j < a_j \end{cases} \tag{5}$$

The ESPPRC with early and late arrival penalties can be formulated as Eq. (6)-Eq. (14). It contains two decision variables: s'_j determines the service start time at customer j as well as the travel cost of arc (i, j), and x_{ij} represents whether arc (i, j) is used in the solution ($x_{ij} = 1$) or not ($x_{ij} = 0$). The objective function (Eq. (6)) minimizes the reduced cost of a single vehicle route including the fixed cost for that vehicle and the travel cost on the arcs with early and late arrival penalty costs. The capacity constraint (Eq. (7)) keeps the cumulative demand within vehicle capacity. Flow conservation constraints (Eq. (8)-Eq. (10)) specify that the route shall start and end at depot; while on the route, if the vehicle travels to a customer location h it must also travel from it. The soft time windows constraints (Eq. (11) and Eq. (12)) allow the arrival of the vehicle within the soft time window $[a'_i, b'_i]$, but ensures that the service must start after a_i even if the vehicle arrives earlier than the service start time i.e. within [a_i, b'_i]. Another time window constraint (Eq. (13)) ensures that if the vehicle travels from i to j, service at j can not start earlier than that at i. Here, M is a large constant. Finally, the integrality constraint for the flow variables x_{ij} is represented by Eq. (14).

$$\min \sum_{(i,j)\in A} \overline{c'_{ij}} x_{ij} \tag{6}$$

subject to

$$\sum_{i\in C} d_i \sum_{j\in V} x_{ij} \le q \tag{7}$$

$$\sum_{j\in V} x_{0j} = 1 \tag{8}$$

$$\sum_{i\in V} x_{ih} - \sum_{j\in V} x_{hj} = 0 \qquad \forall\, h \in C \tag{9}$$

$$\sum_{i\in V} x_{i0} = 1 \tag{10}$$

$$a'_i \le s'_i \le b'_i \qquad \forall\, i \in V \tag{11}$$

$$a_i \le s_i \le b'_i \qquad \forall\, i \in V \tag{12}$$

$$s_i + t_{ij} - s_j \le (1 - x_{ij}) M_{ij} \qquad \forall\, (i,j) \in A \tag{13}$$

$$x_{ij} \in \{0,1\} \qquad \forall\, (i,j) \in A \tag{14}$$

4. THE COLUMN GENERATION-BASED HEURISTICS

The nature of the algorithm used for solving the subproblem determines the exact or heuristics characteristics of any column generation scheme even if the master problem is solved exactly. The ESPPRC subproblem is a NP-hard problem in the strong sense (Dror, 1994) even without early and late arrival penalties. Therefore, the basic idea of the column generation-based heuristics is to solve the subproblem using insertion heuristics, which can efficiently incorporate the arrival time dependent arc costs of the VRPSTW formulation. Figure 2 shows the complete flow chart for the column generation-based heuristics for the VRPSTW. The linear relaxation of the set partitioning master problem LP must remain feasible; therefore, the LP was initialized with n artificial columns equivalent to a solution where every customer is individually serviced by a dedicated vehicle. The costs of these routes provide the first set of values as well as the upper bounds of the dual variables $\boldsymbol{\pi}$. Reduced costs based on these dual variable values is used in the insertion heuristics to generate routes of negative reduced costs, which are then used to augment the LP of the master problem. The set covering master problem is solved by replacing constraint (2) with (15), as linear programming relaxation of set covering type master problem is more stable than the set partitioning type (Desrochers *et al.,* 1992).

$$\sum_{p\in P} a_{ip} y_p \ge 1 \qquad \forall i \in C \tag{15}$$

To stabilize the column generation scheme convex combinations of current and previous dual variables are tried first, if the subproblem fails to provide a negative reduced cost column the weight of current dual variables is increased in steps. The subproblem and the master problem are solved iteratively in cycles until a stopping criterion is achieved, such as maximum number of iterations. If the solution at hand is not integer a problem reduction step (as described in §4.2) is taken.

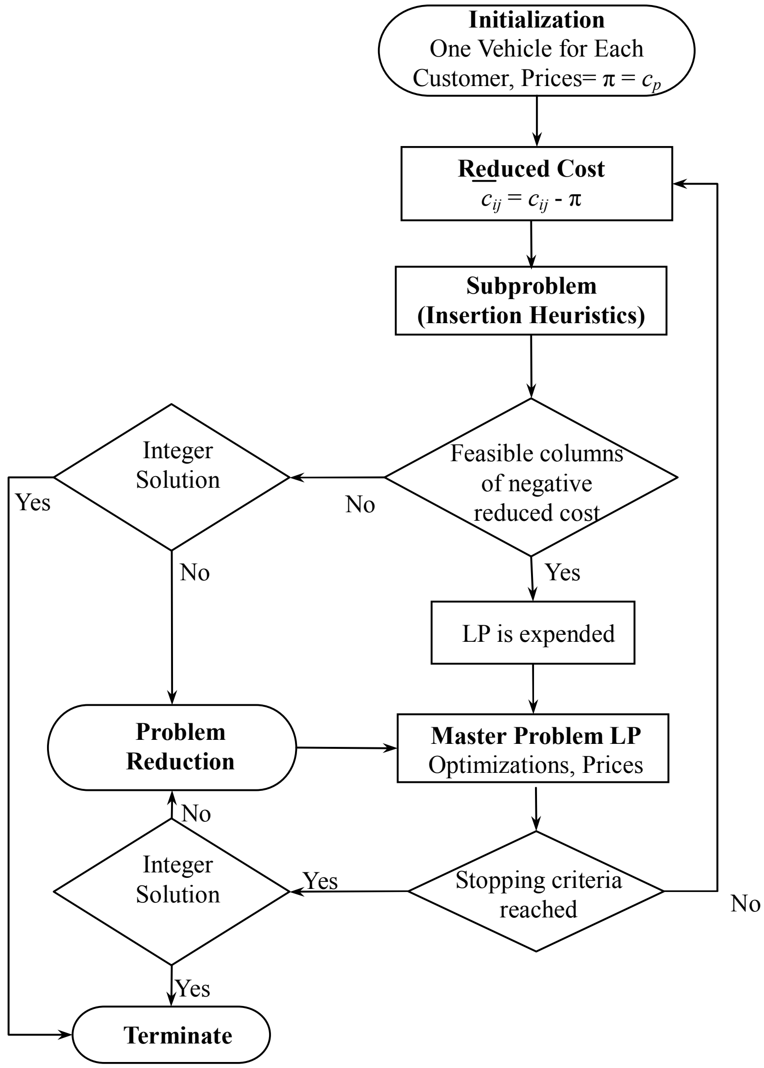

Figure 2 Flow chart of column generation-based heuristics

4.1 Insertion Heuristics Subproblem

The Push Forward Insertion Heuristics (PFIH) (Solomon, 1987) is one of the earliest sequential route-building algorithms for the VRPTW. It has been used in the initialization of many other route-improving heuristics and metaheuristics such as in a GA for VRPHTW, Alvarenga *et al.* (2007) used its stochastic version (SPFIH), in which the first customer of a route is chosen randomly and then remaining unrouted customers are inserted in this route until the capacity or time windows constraints forbid any further insertion. This study utilizes a modified version of SPFIH that incorporates the early and late arrival penalties as well.

Let $(i_0, i_1, i_2, \ldots, i_m)$ be the current partial route that starts and end at the depot (i.e. $i_0 = i_m = 0$). The service start time s_{i_r}, waiting time w_{i_r} and late arrival time l_{i_r} are known for $0 \le r \le m$. Insertion of a customer vertex u between i_{p-1} and i_p, causes a *push forward* (PF_{i_p}) in the schedule at the customer i_p that may change the values of s_{i_r}, w_{i_r} and l_{i_r}, $p \le r \le m$. As shown in the Figure 3, the effects of insertion of a customer need to be evaluated from its point of insertion till the end. For the VRPSTW, the conditions $s_u \le b'_u$ and $s_{i_r} + PF_{i_r} \le b'_{i_r}$ provide the feasibility criteria for a feasible insertion position of the customer u.

Similar to Solomon (1987), the best feasible insertion place is determined using Eq. (16) for each unrouted customer u; however, an additional term is added to consider the changes in early and late arrival penalties for the customers: i_r, $p+1 \le r \le m-1$ in order to find the insertion cost (Eq. (17)) of each unrouted customer u. As in this study, the insertion heuristics is used as the subproblem, reduced costs are used to find the insertion cost of the customer u between i_{p-1} and i_p. Finally, the best customer u^* to be inserted in the route, is obtained using Eq. (18).

$$c_1(i(u), u, j(u)) = \min[c_1(i_{p-1}, u, i_p)],\ p = 1, \ldots, m \tag{16}$$

$$c_1(i_{p-1}, u, i_p) = \overline{c'_{i_{p-1},u}} + \overline{c'_{u,i_p}} - \overline{c'_{i_{p-1},i_p}} + \sum_{r=p+1}^{m-1} (c_e(w_{i_r}^{new} - w_{i_r}) + c_l(l_{i_r}^{new} - l_{i_r})) \tag{17}$$

$$c_2(i(u^*), u^*, j(u^*)) = \min_u [c_1(i(u), u, j(u))] \tag{18}$$

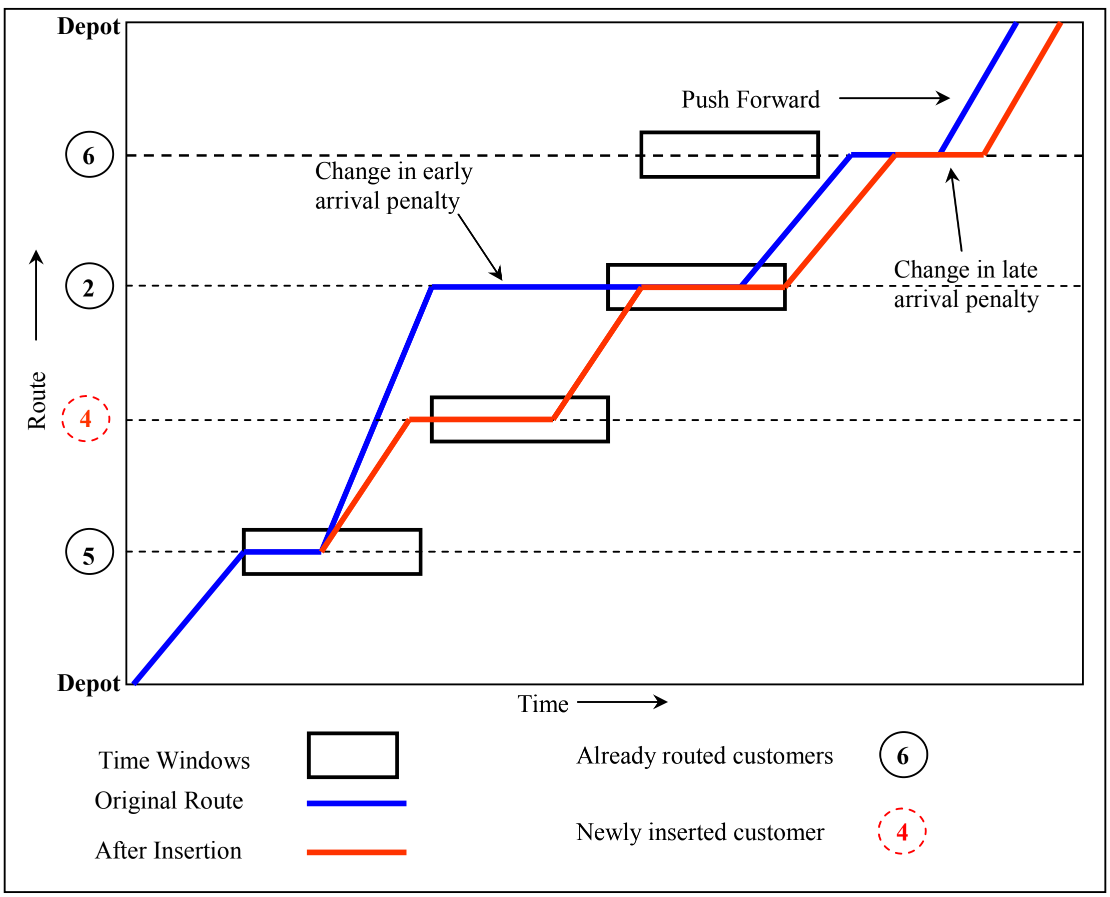

Figure 3 Effect of customer insertion on early and late arrival penalties

The column generation subproblem (Eq. (6)-Eq. (14)) searches for the best negative reduced cost route for only one vehicle and returns the most negative reduced cost column; whereas, the insertion heuristics subproblem actually solves the whole VRPSTW instance considering reduced costs. This helps in ensuring that columns containing all customers are available in the master problem LP. Also to accelerate the column generation scheme the insertion heuristics subproblem is forced to stop earlier if a fixed number (set to 100) of negative reduced costs columns found during its execution.

4.2 Problem Reduction

Three stopping criteria were used in the column generation-based heuristics viz. maximum number of iterations, same values of dual variables in two consecutive iterations and no negative reduced cost column returned by the subproblem. At the end of column generation procedure, if the solution is fractional or contains more than one single-customer routes, a problem reduction step was taken thereby extracting routes depending on the values of y_p and formulating a smaller instance. The following procedure is used in this stage.

Step 1: Optimize the set covering master LP problem containing all present columns and select a column with the maximum y_p value; add all its customers to temporary solution *S*'

Step 2: Remove all columns from master LP problem containing any of the customers in temporary solution *S*'

Step 3: Check
if $|C| > 25$, $|S'| \geq (|C|) \times 0.5$ go to step 4 else go to step 1
if $|C| < 25$, $|S'| = (|C|)$ go to step 4 else go to step 1

Step 4: If number of single customer routes in *S'* > 1, remove all single customer routes from *S'*

Step 5: Reformulate the reduced VRPSTW instance from customer set $C' = C / S'$ and repeat the column generation-based heuristics of Figure 2.

No proper branch and bound scheme can be developed due to the use of heuristics. This approach can be seen as branching on various variables at a time. The cycle of column generation scheme and problem reduction continues until the integer solution of the original instance with at most one single-customer route is obtained.

5. GENETIC ALGORITHM HEURISTICS (GA)

As discussed earlier (in §2), typically, Genetic Algorithms (GA) have been used to obtain good solutions for the VRPSTW. Therefore, this study also adopts a GA for the VRPSTW to compare the results of the column generation-based heuristics in order to identify its practicality. In GA, initialization and size of population, genetic operators (crossover and mutation), elitism and maximum number of generations (iterations) play important roles in the heuristics quality. The GA, used for comparison, has been evaluated for its satisfactory performance by comparing it with exact VRPTW solutions under slightly different soft time windows settings (Qureshi *et al.*, 2009).

5.1 Chromosome Representation and Population

The same modified Stochastic Push Forward Insertion Heuristics (SPFIH) was used to get the initial population for the GA as was used in the subproblem of column generation-based heuristics (as described in §4.1) with only exception that the true arc costs (not the reduced ones) were used. The population was consisted of 200 integer valued individuals or chromosomes, each representing a complete feasible VRPSTW solution obtained from the modified SPFIH. Figure 4 shows such a chromosome for a twelve customer instance and its interpretation in the GA for new vehicles due to presence of a depot gene or due to the violation of capacity or time window constraints. Two continuous variables ($qsum_0$ = 0 and $twroute_0$) for each vehicle are initiated (as per Eq. (19)) and are updated every time that vehicle travels from i to j according to the Eq. (20) and Eq. (21).

$$twroute_0 = \max[a_0, a_i - t_{0i}] \tag{19}$$

$$qsum_j = qsum_i + d_j \tag{20}$$

$$twroute_j = twroute_i + t_{ij} \tag{21}$$

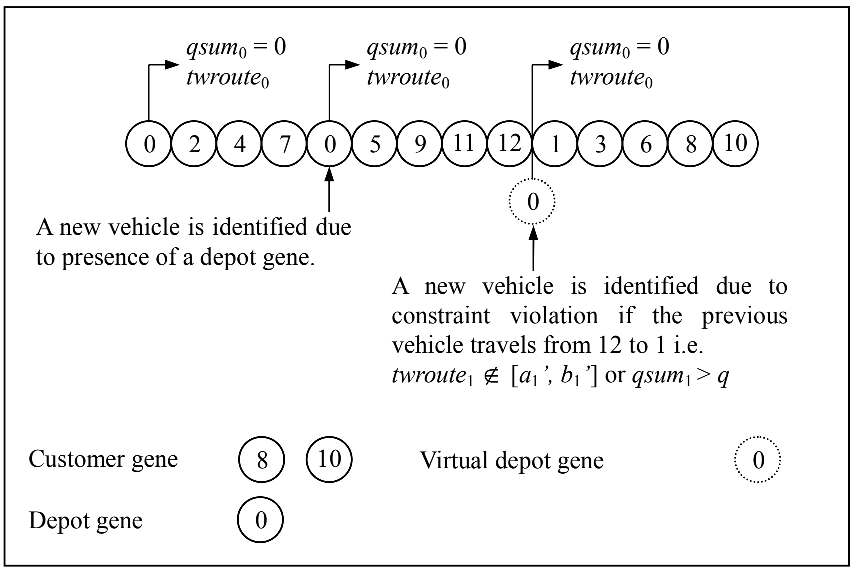

Figure 4 The VRPSTW chromosome coding and interpretation in GA

5.2 Crossover, Mutation and Elitism

To generate population for the next generation, individuals from the present population were selected using Stochastic Universal Selection (SUS) method (Chipperfield *et al.,* 1994) based on their fitness value. Parents were taken from this population and were used in crossover to generate offspring, which were inserted in the next generation's population irrespective of their fitness value. To maintain the feasibility of the chromosomes i.e. to avoid duplication of same customer gene, an ordered-based two-point crossover was used with a crossover rate of 98%. A simple swap mutation was used to stirrup the search pattern at a mutation rate of 10%. To ensure that each iteration of the GA always finds a new or maintains the best solution found so far, elitism was adopted thereby keeping best 2% individuals of the current population in the population of next generation.

The maximum number of iteration (generations) was kept as 25000, and to reduce the effects of the initial solution, the population was re-generated after every 500 generations. During this step, a new population was generated by keeping 2% elite individuals of the current population and the remaining 98% was generated using the same modified SPFIH algorithm as that used in the initialization. Therefore, along with the GA, an implicit comparison of the proposed column generation-based heuristics, with roughly 9800 runs (50 re-generations x 0.98 x size of initial population) of the modified SPFIH, was also made as a byproduct.

6. COMPUTATIONAL RESULTS AND DISCUSSION

The algorithms were implemented in MATLAB, and were run on a computer with 2.41 GHz AMD Athlon with 64 x 2 dual core processors with 2 GB of RAM. The performance of column generation-based heuristics was evaluated using some of the R1-type Solomon's benchmark instances (Solomon, 1987). Customers are located randomly in these instances, and each instance contains 100 customers; for each of the instances ten runs of column generation-based heuristics and GA were made and the results of the best run are given in this paper.

Table 1 provides the solution results obtained using column generation-based heuristics and GA heuristics. Col. (1) shows the test instance; Solomon's benchmark instances are originally developed for the VRPHTW, therefore a suffix STW is added to elaborate the fact that soft time windows have been used in the test instances. Number of vehicles required and cumulative sums of their operation times, early arrival times and later arrival times in the best solution found in column generation-based heuristics (in minutes), are reported in Col.(2)-Col.(5). Col.(6)-Col.(9) contains the corresponding figures for the GA heuristics. The operation time for a vehicle includes the time required to travel from depot to the first customer, between the customers on route and returning back to depot.

Table 1 Results of column generation-based heuristics and GA heuristics

Instance	Column Generation-based Heuristics				Genetic Algorithms (GA) Heuristics			
	Veh.	OT	EAT	LAT	Veh.	OT	EAT	LAT
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
R101-100-STW	13	2677	36.1	474.1	14	2825.2	35.8	467.8
R102-100-STW	12	2572.1	27.3	178.7	13	2640.0	82.4	263.6
R103-100-STW	11	2382	29.0	60.8	12	2580.8	53.9	51.8
R104-100-STW	10	2198.9	49.3	2.0	10	2123.9	8.2	0.0
R105-100-STW	12	2566.3	0.3	293.5	13	2603.4	20.8	115.5

Veh. = number of vehicles OT = operation time EAT = early arrival time LAT = late arrival time

Solutions obtained in column generation-based heuristics contain less number of vehicles and thus have less operation time as compared to the solutions obtained in GA heuristics except in R104-100-STW, where the number of vehicles is the same, and GA heuristics results in less operation time. The early arrival time and late arrival time does not show any trend. The solution costs and the computation time required by the two heuristics are reported in Table 2. The solution cost is composed of fixed vehicle cost, operation cost, early and late arrival penalty costs. The vehicle operation cost (VOC) of 14.02 yen/minute is taken; while the fixed cost for a vehicle is set to 10417.5 yen. The unit early arrival penalty cost is assumed equal to the VOC; whereas, the unit late arrival penalty is taken as five times that of the VOC. These unit cost values are based on a survey of Japanese logistics companies and most commonly used in the city logistics-related literature (for example, see Taniguchi *et al.*, 2001; Yamada *et al.*, 2004; Ando and Taniguchi, 2006; Duin *et al.*, 2007).

A very prominent trend that can be observed from the data of Table 2 is that of very low computation time requirement of the column generation-based heuristics as compared to the GA heuristics. In just over a quarter of computational effort, an average reduction of 4.5% in solution cost was obtained. If both solution cost and computation time are considered together, the overall better performance of column generation-based heuristics is fairly evident as shown in Figure 5. The maximum cost reduction of 10.2% was obtained in R102-100-STW and the maximum computation time saving was observed in R105-100-STW, in which it produced a comparable solution in less than one-fifth of the computation effort required by the GA. In soft time windows environment, always there exists a trade-off between the number of vehicles and delays (late arrival time). Table 1 shows that the column generation-based heuristics has favored less number of vehicles and travel distance over the amount of the late arrival penalties in R105-100-STW, which has resulted in a nominal increase of 0.8% in the solution cost but a corresponding reduction in CO_2 emissions is also obtained (Figure 6). However, in R104-100-STW, GA has produced an overall better solution consuming 2.38 times more computation time. Most of the customers in R104-STW-100 have very wide time windows, which result in a large solution space. The search strategy in GA is global and it has more randomness, which help it to search larger solution spaces in an efficient way. Although, the SPFIH provides randomness and stochastic character in the column generation-based heuristics, it still possesses some characteristics of structural search due to the utilization of dual variables, which may have affected its performance in large scattered solution space.

Table 2 Solution costs and computation times

Instance	Solution Cost (JPY)			Computation Time (seconds)		
	CGH	GAH	Difference (%)	CGH	GAH	Ratio
(1)	(2)	(3)	(4)	(5)	(6)	(7)
R101-100-STW	192680	204729	-5.9	2998.18	11421.70	0.26
R102-100-STW	159960	178054	-10.2	2838.40	7180.98	0.40
R103-100-STW	138637	151560	-8.5	2687.04	6406.65	0.42
R104-100-STW	121815	120047	1.5	1573.35	13153.60	0.12
R105-100-STW	167548	166295	0.8	2072.71	11305.40	0.18

CGH = column generation-based heuristics GAH = genetic algorithms heuristics JPY = Japanese Yen
Difference (%) = (Col.(3) - Col.(2))x100/ Col.(3) Ratio = Col.(5)/ Col.(6)

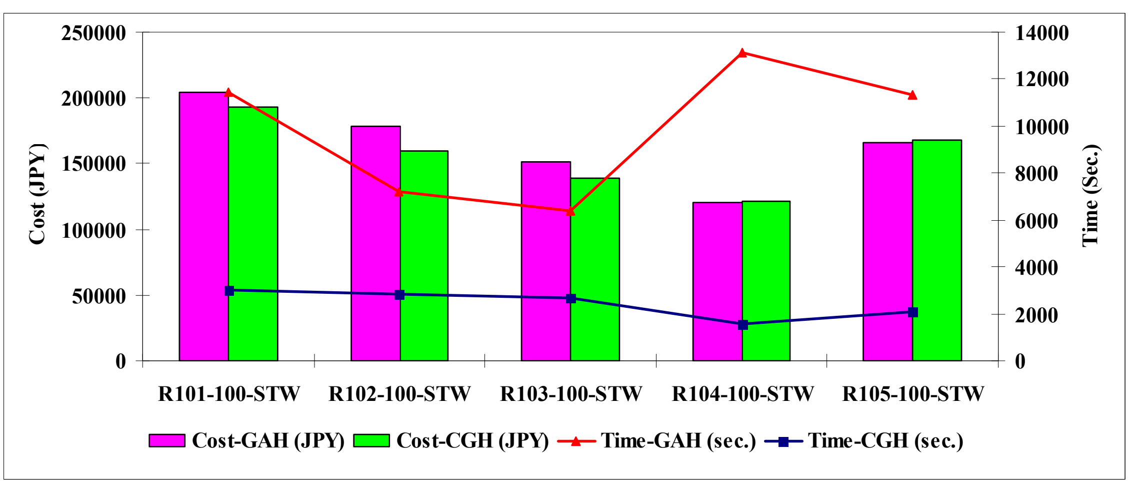

Figure 5 Comparison of solution costs and computation times

To capture the environmental aspects of the solutions produced by the column generation-based and GA heuristics, CO_2 emissions are compared in this paper (Figure 6). These emissions are estimated using relationships (equations) described in a report by National Institute for Land and Infrastructure Management, Japan (NILIM 2003) assuming light delivery vehicles using diesel fuel. Furthermore, as the Solomon's benchmark instances are based on Euclidean distances, an average speed of 20 km/h is assumed on all arcs. As compared to GA heuristics, in almost all instances the column generation based heuristics produced environmentally better results except in R104-100-STW, for which it produced an overall inferior solution. The reduction in the CO2 emissions averaged at 4% in all instances.

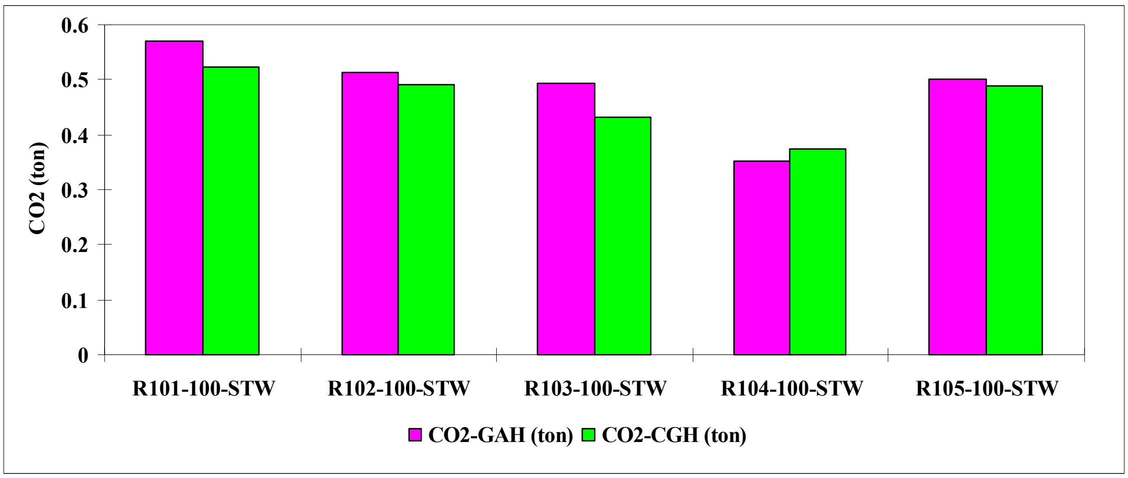

Figure 6 Comparison of CO_2 emissions

7. CONCLUSIONS AND FUTURE RESEARCH

Considering its efficiency for the hard time windows and taking advantage of the flexibility offered by the column generation scheme to work with variety of subproblems, even the heuristics ones, this paper presented a column generation-based heuristics for the Vehicle Routing and scheduling Problem with Soft Time Windows (VRPSTW). The dual information (i.e., shadow prices) from master problem guided the heuristic subproblem and it was able to provide negative reduced cost columns of good quality. This resulted in a rapid decrease in the objective function value, which saved considerable computation time as compared to the slow convergence of a traditionally used genetic algorithms (GA) heuristics. Computational evaluations on Solomon's benchmark instances also confirmed this, and on average, the column generation-based heuristics was able to save three-forth of the computation time required by the simple GA heuristics. The column generation-based heuristics was able to produce better solutions, both economically and environmentally, in most instances and very closely comparable solutions in other instances. At the end of the column generation procedure, to obtain a feasible integer solution, a problem reduction strategy has been used. Although the strategy worked well yet it is too coarse (i.e., a sudden reduction of instance size to half), this may become a significant weakness while attempting even larger instances (>100 customers). Therefore, a possible future research can be towards improving this strategy, evaluating their effects on solution quality and computation of the proposed heuristics.

REFERENCE

Alvarenga, G. B., Mateus, G. R. and Tomi, G. de. (2007) A genetic and set partitioning two-phased approach for the vehicle routing problem with time windows, **Computers & Operations Research, Vol. 34**, 1561-1584.

Ando, N., and Taniguchi, E. (2006) Travel time reliability in vehicle routing and scheduling with time windows, **Networks and Spatial Economics, Vol. 6**, 293-311.

Balakrishnan, N. (1993) Simple heuristics for the vehicle routing problem with soft time windows, **Journal of the Operational Research Society, Vol. 44**, 279-287.

Braysy, O., and Gendreau, M. (2005a) Vehicle routing problem with time windows, Part I: Route construction and local search algorithms, **Transportation Science, Vol. 39**, 104-118.

Braysy, O., and Gendreau, M. (2005b) Vehicle routing problem with time windows, Part II: Metaheuristics, **Transportation Science, Vol. 39**, 119-139.

Calvete, H. I., Gale, C., Oliveros M. J. and Valverde, B. S. (2007) A goal programming approach to vehicle routing problem with soft time windows, **European Journal of Operational Research, Vol. 177**, 1720-1733.

Chabrier, A. (2006) Vehicle routing problem with elementary shortest path based column generation, **Computers & Operations Research, Vol. 33**, 2972-2990.

Chen, Z., and Xu, H. (2006) Dynamic column generation for dynamic vehicle routing with time windows, **Transportation Science, Vol. 40** 74-88.

Chipperfield, A., Fleming, P., Pohlhiem, H. and Fnoseca, C. (1994) **User guide: Genetic algorithm toolbox for use with MATLAB**, Department of Automatic Control and Systems Engineering, University of Sheffield. Available at: http://www.shef.ac.uk/uni/projects/gaipp/gatbx.html [accessed on November 10, 2004].

Dana, E., and Pape, C. Le. (2005) Branch and price heuristics: A case study on the vehicle routing problem with time windows. In Desaulniers, G., Desrosiers, J. and Solomon, M. M. (eds.), **Column Generation**, 99-129, Springer, New York.

Desrochers, M., Desrosiers, J. and Solomon, M. (1992) A new optimization algorithm for the vehicle routing problem with time windows, **Operations Research, Vol. 40**, 342-354.

Dror, M. (1994) Note on the complexity of the shortest path models for column generation in VRPTW, **Operations Research, Vol. 42**, 977-978.

Duin, van.J.H.R., Tavasszy, L. A. and Taniguchi, E. (2007) Real time simulation of auctioning and re-scheduling process in hybrid freight markets, **Transportation Research Part B, Vol. 41**, 1050-1066.

Dumas, Y., Somis, F. and Desrosiers, J. (1990) Optimizing the schedule for a fixed vehicle path with convex inconvenience costs, **Transportation Science, Vol. 24**, 145-152.

Fagerholt, K. (2001) Ship scheduling with soft time windows: An optimisation based approach, **European Journal of Operational Research, Vol. 131**, 559-571.

Feillet, D., Dejax, P., Gendreau, M. and Gueguen, C. (2004) An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems, **Networks, Vol. 44**, 216-229.

Hashimoto, H., Ibaraki, T., Imahori, S. and Yagiura, M. (2006) The vehicle routing problem with flexible time windows and traveling times, **Discrete Applied Mathematics, Vol. 154**, 2271-2290.

Ioachim, I., Gelinas, S., Soumis, F. and Desrosiers, J. (1998) A dynamic programming algorithm for the shortest path problem with time windows and linear node costs, **Networks, Vol. 31**, 193-204.

Irnich, S., and Villeneuve, D. (2003) **The Shortest Path Problem with K-Cycle Elimination (K≥3): Improving a Branch and Price Algorithm for the VRPTW**, Technical Report G-2003-55, GERAD.

Kohl, N., Desrosiers, J., Madsen, O. B. G., Solomon, M. M. and Soumis, F. (1999) 2-Path cuts for the vehicle routing problem with time windows, **Transportation Science, Vol. 33**, 101-116.

NILIM, National Institute of Land and Infrastructure Management (2003) **Quantitative appraisal index calculations used for basic unit computation of CO2, NOx, SPM** (in Japanese).

Qureshi, A. G., Taniguchi, E. and Yamada, T. (2009) Exact solution for vehicle routing problem with semi soft time windows and its application, **City Logistics VI**, Proceedings of Sixth International Conference on City Logistics, 30th June – 2nd July, 2009, Puerto Vallarta, Mexico, 49-64.

Qureshi, A. G., Taniguchi, E. and Yamada, T. (2007) Effects of relaxing time windows on vehicle routing and scheduling, **Infrastructure Planning Review, Vol. 24**, 927-936.

Qureshi, A. G., and Hanaoka, S. (2005) Analysis of the effects of cooperative delivery system in Bangkok. In Taniguchi, E., and Thompson, R. G. (eds.), **Recent Advances in City Logistics Proceedings of the 4th International Conference on City Logistics, Langkawi, Malaysia, 2005**, 59-73, Elsevier, Oxford.

Rochat Y., and Taillard, E. D. (1995) Probabilistic diversification and intensification in local search for vehicle routing, **Journal of Heuristics, Vol. 1**, 147-167.

Sexton, R. T., and Bodin, L. D. (1985) Optimizing single vehicle many to many operations with desired delivery times: I. Scheduling, **Transportation Science, Vol. 19**, 378-410.

Solomon, M. M. (1987) Algorithms for the vehicle routing and scheduling problem with time windows constraints, **Operations Research, Vol. 35**, 254-265.

Tagmouti, M., Gendreau, M. and Potvin, J. Y. (2007) Arc routing problem with time-dependent service costs, **European Journal of Operational Research, Vol. 181**, 30-39.

Taniguchi, E., Thompson, R. G., Yamada, T. and Duin, R. V. (2001) **City logistics; Network Modeling and Intelligent Transport Systems**, Pergamon, Oxford.

Taniguchi, E., and Heijden, R. E. C. M. van der. (2000) An evaluation methodology for city logistics, **Transport Reviews, Vol. 20**, 65-90.

Thompson, R.G., and Taniguchi, E. (2001) City logistics and freight transport. In Brewer, A. M., Button, K. J. and Hensher, D. A. (eds.), **Handbook of Logistics and Supply-Chain Management**, 393-405, Pergamon, Oxford.

Yamada, T., Yoshimura, Y. and Mori, K. (2004) Road network reliability analysis using vehicle routing and scheduling procedures. In Taniguchi, E., and Thompson, R. G. (eds.), **Logistics systems for sustainable cities, proceedings of the 3rd international conference on city logistics**, 111-122, Elsevier, UK.

Yamada, T., Taniguchi, E. and Itoh, Y. (2001) Co-operative vehicle routing model with optimal location of logistics terminals. In Taniguchi, E., and Thompson, R. G. (eds.), **City Logistics II**, 139-153, Institute for City Logistics, Japan.