Journal of Service Science and Management, 2015, 8, 844-859
Published Online December 2015 in SciRes. http://www.scirp.org/journal/jssm
http://dx.doi.org/10.4236/jssm.2015.86085

A Comparative Study of Proposed Genetic Algorithm-Based Solution with Other Algorithms for Time-Dependent Vehicle Routing Problem with Time Windows for E-Commerce Supply Chain

Suresh Nanda Kumar, Ramasamy Panneerselvam

School of Management, Pondicherry University, Pondicherry, India
Email: snkumar27@gmail.com, panneer_dms@yahoo.co.in

Received 26 October 2015; accepted 13 December 2015; published 16 December 2015

Abstract

The vehicle routing problem (VRP) is classified as an NP-hard problem. Hence exact optimization methods may be difficult to solve these problems in acceptable CPU times, when the problem involves real-world data sets that are very large. To get solutions in determining routes which are realistic and very close to the optimal solution, one has to use heuristics and meta-heuristics. In this paper, an attempt has been made to develop a GA based meta-heuristic to solve the time dependent vehicle route problem with time windows (TDVRPTW). This algorithm is compared with five other existing algorithms in terms of minimizing the number of vehicles used as well as the total distance travelled. The algorithms are implemented using Matlab and HeuristicLab optimization software. A plugin was developed using Visual C# and NET Framework 4.5. Results were tested using Solomon's 56 benchmark instances (of which 24 instances are used with 4 in each of the 6 problem classes) classified into groups such as C1, C2, R1, R2, RC1, and RC2. For each of the performance measures, through a complete factorial experiment with two factors, it is proved that the proposed algorithm is the best among all the six algorithms compared in this paper.

Keywords

Vehicle Routing Problem, Exact Methods, Heuristics, Metaheuristics, VRPTW, TDVRPTW, Optimization, Genetic Algorithms, Tabu Search, Matlab, HeuristicLab

How to cite this paper: Kumar, S.N. and Panneerselvam, R. (2015) A Comparative Study of Proposed Genetic Algorithm-Based Solution with Other Algorithms for Time-Dependent Vehicle Routing Problem with Time Windows for E-Commerce Supply Chain. *Journal of Service Science and Management*, **8**, 844-859. http://dx.doi.org/10.4236/jssm.2015.86085

1. Introduction

Vehicle Routing Problem (VRP) is one of the most important topics in operations research. It deals with determining least cost routes from a depot to a set of scattered customers. The routes have to satisfy the following set of constraints:

- Each customer is visited exactly once.
- All routes start and end at the depot.
- Sum of all demands on a route must not exceed the capacity of a vehicle.

The vehicle routing problem is depicted in **Figure 1**. From this figure, one can note that the vehicles originate at the depot, visit all the nodes once, fulfill their demands at the nodes and then return to the depot once again. The nodes can be either supplier sites in the case of ecommerce company supplier site pickups or customer sites where the ecommerce company delivers to the customer sites. It can also be different manufacturers to whom the suppliers deliver raw materials, components, parts and other supplies of items. The various colours depicted in the figure represent the various routes of the vehicles to fulfill the customer demand.

VRP is closely related to TSP, and according to Bullnheimer *et al*. [1], as soon as the customers of the VRP are assigned to vehicles, the problem is reduced to several or multiple TSPs. The Vehicle Routing Problem (VRP) is used to design an optimal route for a fleet of vehicles to service a set of customers subject to a set of givn constraints. The VRP is used in supply chain management in the physical delivery of goods and services. There are several variants to the VRP. These are formulated based on the nature of the transported goods, the quality of service required and the characteristics of the customers and the vehicles. The VRP is of the NP-hard type.

The vehicle routing problem (VRP) has been very extensively studied in the optimization literature. It started with the seminal papers of Dantzig and Ramser [2] and Clarke and Wright [3]. Now, VRP offers a wealth of heuristic and meta-heuristicses, which are surveyed in the papers of Laporte [4], Gendreau *et al*. [5] and Cordeau *et al*. [6]. The VRP is so widely studied because of its wide applicability and its importance in determining efficient strategies for reducing operational costs in distribution networks. Today, exact VRP methods have a size limit of 50 - 100 orders depending on the VRP variant and the time-response requirements. To overcome this limitation, the research on VRP currently concentrates on approximate algorithms and meta-heuristics that are capable of finding high quality solutions in limited time, in order to be applicable to real-life problem instances that are characterized by large vehicle fleets.

The VRP was first stated by Dantzig and Remser [2] which was about the routing of a fleet of gasoline delivery trucks between a bulk terminal and a number of service stations supplied by the terminal. The total distance between any two locations is given and a demand for a given product is specified for the service stations. The Time-Dependent Vehicle Routing Problem (TDVRP) is a class of vehicle routing problems, where the time to serve the customers varies along with the consideration of the traffic conditions in the route. In order to collect

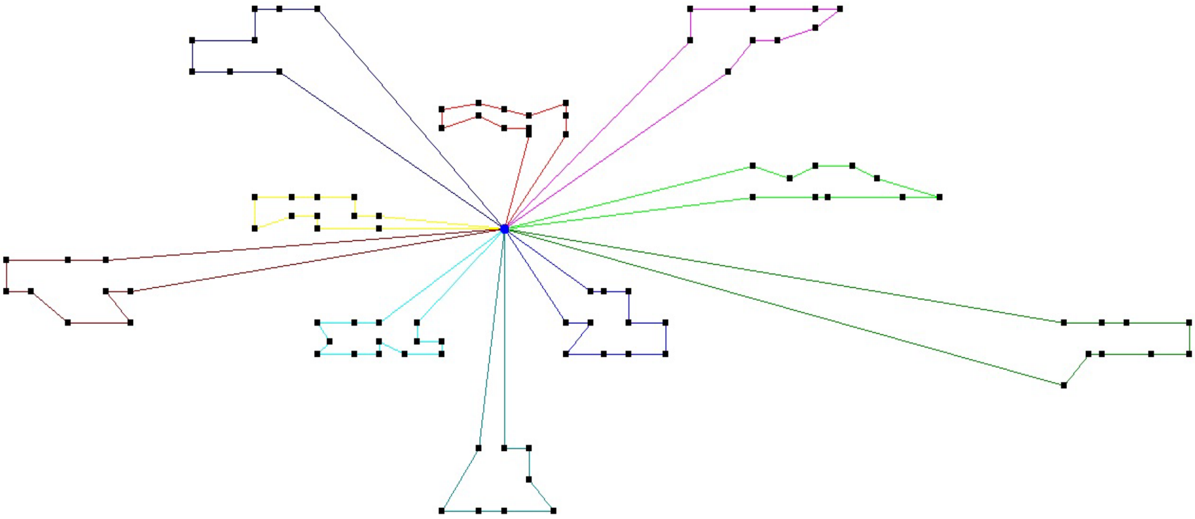

Figure 1. An example solution to a vehicle routing problem.

the items from various suppliers, the 3PL logistics transportation service provider must visit all scheduled suppliers during different opening hours. Some suppliers may further request visits within certain time windows. It is not always easy to meet the time window delivery requirement because the delivery processes are usually affected by traffic flow conditions. The traffic congestion during the rush hours might cause severe delays. Hence in order to pick-up items from the suppliers more efficiently, this study seeks to solve several challenging missions simultaneously among customers and suppliers, in a 3PL kind of arrangement, whereby the buyer's vehicle visits the suppliers' sites to pickup the items ordered, in an e-commerce setup, including: 1) to satisfy suppliers' specific time windows; 2) to approximate travel time affected by urban traffic.

In this paper, the basic vehicle routing problem with time windows (VRPTW) is further extended and the time dependent vehicle routing problem with time windows is studied. The TDVRPTW is gaining more importance in research, since the traffic conditions are different during different times of the day and manufacturers, suppliers and ecommerce retailers need to schedule their pickups and deliveries at the appropriate times of the day, considering the time windows of the customers and suppliers to make their order fulfillment needs efficient and faster. The methodology adopted in this study is based on genetic algorithm. It is important to make this study, since a gap exists in the literature in the area of TDVRPTW used by ecommerce companies in supplier site pickups. This study may help ecommerce companies to optimize their supplier pickup activity.

In this paper, the review of literature of VRPTW is followed by genetic algorithm used and the crossover technique developed in this study. The implementations of the GA-based algorithm for solving the TDVRPTW and its steps are also discussed. Finally the proposed algorithm is compared with a set of existing algorithms in terms of number of vehicles utilized as well as total distance travelled.

2. Literature Review

The VRPTW is classified as a NP-hard problem (e.g. Fu, [7]; Meng *et al.*, [8]). Solomon [9] first presented a mixed-integer programming (MIP) for the VRPTW and introduced a set of well-known benchmark problems now known as "Solomon Instances." He subsequently designed and analyzed algorithms for the VRPTW (Solomon, [9]). To consider traffic congestion, the time-dependent traveling time is added into the VRPTW as the time-dependent vehicle routing problem with time windows (TDVRPTW). Malandraki & Daskin [10] discussed diversified traffic conditions at different times of the day; the time horizon is divided into M slices and then a constant travel time is assigned to each arc in every interval. The idea is sound; however, the discontinuous travel time settings may violate the first in, first out (FIFO) property. Hill & Benton [11] also considered TDVRP without time windows but based on time-dependent travel speed. Ichoua *et al.* [12] assigned a speed distribution to each arc during the time horizon and then obtained the travel time distribution by integration.

Many of the most successful meta-heuristics for the large VRPTW instances are based on some form of parallel computation. During the past few years, numerous papers have been written on generating good solutions for VRPTW with GAs. Genetic Algorithms (GA for short) are a class of adaptive heuristics based on the Darwinian concept of evolution survival of the fittest. A brief review of the relevant literature is given below.

2.1. Meta-Heuristics

Le Bouthillier and Crainic [13] have proposed a cooperative parallel meta-heuristic for the VRPTW, based on the solution warehouse strategy. In this work, several search threads cooperate asynchronously, exchanging information on the best solutions identified. The exchanges are performed through a mechanism called solution warehouse, which holds and manages a pool of solutions. Blanton and Wainwright [14] were the first to apply a genetic algorithm to VRPTW. They hybridized a genetic algorithm with a greedy heuristic. Under this scheme, the genetic algorithm searches for a good ordering of customers while constructing a feasible solution using a greedy heuristic. Several papers present hybridizations of a GA with different construction heuristics (Berger *et al.*, [15]), local searches (Thangiah *et al.*, [16]; Potvin and Bengio, [17]; Zhu [18]) and other meta-heuristics such as tabu search (Wee Kit [19]) and ant colony system (Berger *et al.*, [15]).

Homberger and Gehring [20] proposed a two-phase hybrid meta-heuristic for the VRPTW. The objective function of the VRPTW considered here combines the minimization of the number of vehicles (primary criterion) and the total travel distance (secondary criterion). The aim of the first phase is the minimization of the number of vehicles by means of a (l; k)-evolution strategy, whereas in the second phase the total distance is minimized using a tabu search algorithm. Mester and Bräysy [21] present a new and effective meta-heuristic algorithm, active guided evolution strategies, for the VRPTW problem. The algorithm combines the strengths of the guided local search and evolution strategies belonging to meta-heuristics into an iterative two-stage procedure. Guided local search is used to regulate a composite local search in the first stage and the neighbourhood of the evolution strategies algorithm in the second stage. Russell and Chiang [22] used a scatter search meta-heuristic to solve the VRPTW. Both a common arc method and an optimization-based set covering model are used to combine vehicle routing solutions. A reactive tabu search meta-heuristic and a tabu search with an advanced recovery feature, together with a set covering procedure are used for solution improvement. Alba and Dorronsoro [23] proposed a cellular Genetic Algorithm (cGA) which is a kind of decentralized population based heuristic, which is used for solving capacitated vehicle routing problem (CVRP). Tabu search (TS) is a memory based search strategy to guide the local search descent method to continue its search beyond local optimality Glover [24]; Glover [25]. When a local optimum is encountered, a move to the best neighbor is made to explore the solution space, even though this may cause a deterioration in the objective function value. The TS seeks the best available move that can be determined in a reasonable amount of time. More developments and applications have been discussed by Glover, Taillard and De Werra [26]. The tabu search algorithm utilizes three different neighborhoods that have been proposed by Li and Lim [27]. The shift neighborhood considers moves where pickup and delivery customer pairs are shifted from one route to another. In the exchange neighborhood pairs are swapped between two routes. Within one route pairs can be moved to another position in the rearrange neighborhood. As a tabu criterion, a customer cannot be moved back to a route once it has been removed or rearranged in it. Whenever a new request arrives, there are two approaches in integrating it in the current route plan (Ichoua *et al*., [28]). Lau *et al*. [29] introduced a variant of the vehicle routing problem with time windows where a limited number of vehicles is given (m-VRPTW). Under this scenario, a feasible solution is one that may contain either unserved customers and/or relaxed time windows. Kramer *et al*. proposed a Pollution-Routing Problem (PRP) which is a "green" oriented variant of the Vehicle Routing Problem (VRP). In order to solve it, they proposed a meta-heuristic called ILS-SOA-SP, that effectively integrates Iterated Local Search (ILS) with a Set Partitioning (SP) procedure and a Speed Optimization Algorithm (SOA). This approach was also used to solve two other environmental-based VRPs, namely the Fuel Consumption Vehicle Routing Problem (FCVRP) and the Energy Minimizing Vehicle Routing Problem (EMVRP), as well as the well-known Vehicle Routing Problem with Time Windows (VRPTW) with distance minimization. Chu *et al*. [54] developed a mathematical model for solving the TDVRPTWSD problem. They studied TDVRPTWSD and used Genetic Algorithm to solve it. Their study has not considered the objective of minimizing the number of vehicles used and also minimizing the total distance travelled.

2.2. Time-Dependent Vehicle Routing Problem with Time Windows

Time dependent vehicle routing problems received little attention among researchers. The time dependent VRP was first formulated by Malandraki [30] and Malandraki and Daskin [10] using a mixed integer linear programming formulation. They proposed a greedy nearest-neighbor heuristic based on travel time between customers, as well as a branch and cut algorithm to solve TDVRP without time windows. Hill and Benton [11] considered a node-based time dependent vehicle routing problem (without time windows). Computational results for one vehicle and five customers were reported. Ahn and Shin [31] introduced certain modifications to the savings, insertion, and local improvement algorithms to better deal with TDVRP. In randomly generated instances, they reported reductions in computational time as a percentage of the "unmodified" savings, insertion, and local improvement algorithms. Malandraki and Dial [32] proposed a dynamic programming algorithm for the time-dependent traveling salesman problem, *i.e*. for a fleet of just one vehicle. A nearest-neighbor type heuristic was used to solve randomly generated problems.

An important property for time dependent problems is the First In-First Out (FIFO) property proposed by Ahn and Shin [25] and Ichoua *et al*. [12]. A model which has a FIFO property guarantees that if a vehicle leaves customer *i* to go to customer *j* at any time *t*, any identical vehicle with the same destination leaving customer *i* at a time $t + e$, where $e > 0$, will always arrive later. This is an intuitive and desirable property though it is not present in all models. Earlier formulations and solutions methods (Malandraki [30], Malandraki and Daskin [10], Hill and Benton [11], and Malandraki and Dial [32]), do not guarantee the FIFO property as reported by Ichoua *et al*. [12]. Later researchers modeled travel time variability using "constant speed" time periods, which guarantees the FIFO property as shown by Ichoua *et al*. [12]. Ichoua *et al*. [12] proposed a tabu search solution method based on the work of Taillard *et al*. [33] in order to solve time dependent vehicle routing problems with soft time windows. Fleischmann *et al*. [34] utilized a route construction method of savings and insertion, to solve uncapacitated time dependent VRP with and without time windows. They tested their algorithms in instances created from Berlin travel time data. Jung and Haghani [35] proposed a genetic algorithm to solve time dependent problems. Using randomly generated test problems, the performance of the genetic algorithm was evaluated by comparing its results with exact solutions (up to 9 customers and 15 time periods) and a lower bound (up to 25 customers and 10 time periods). Haghani and Jung [50] further propose a formulation for a dynamic vehicle routing problem with time-dependent travel times and real-time vehicle control that is an NP-hard problem. For solving this problem, they proposed a genetic algorithm. This algorithm includes a vehicle merging operator in addition to the generic genetic operators, namely the crossover and the mutation operators.

Donati *et al*. [36] proposed a solution that adapted the ant colony optimization meta-heuristic and a local search improvement method, which stores and updates the slack times or feasible delays. They used Solomon's benchmark instances as the test data. More recently, Soler *et al*. [37] proposed a method to solve TDVRP instances optimally that are too small for practical purposes and it experiences exponential growth of computational time as a function of problem size. Dabia *et al*. [38] dealt with a one-vehicle vehicle routing problem (TSP) using a dynamic programming approach. Kok [39] dealt with the TDVRP with a focus on departure time optimization and driver break scheduling. The study by Kok uses a modification of the set of benchmark instances for the VRP with time dependent travel speeds proposed by an early working paper by Figliozzi [40]. Ichoua *et al*. [12] used the well-known Solomon's 56 benchmark problems for the VRP with time windows.

An important real-life property found in transportation problems in the retail industry are time-dependent travel times, also known as dynamic travel times, where travel time depends on the time of departure, modeling traffic conditions such as rush hours. The VRP with dynamic travel times is described by van Woensel *et al*. [41] using queueing theory to obtain travel times. Kok *et al*. [42] proposed a post-processor to determine optimal departure times for the vehicle routes. Kuo *et al*. [43] introduces a separate calculation model to calculate the total operation time of all vehicles and used a modified tabu search to optimize the sequence of customers visited in the routes. Bettinelli *et al*. [44] described a version, where multiple warehouses are considered using a branch-and-cut-and-price algorithm.

Jung and Haghani [50] proposed a genetic algorithm to solve time dependent problems. Using randomly generated test problems, the performance of the genetic algorithm was evaluated by comparing its results with exact solutions (up to 9 customers and 15 time periods) and a lower bound (up to 25 customers and 10 time periods). As the time dependent vehicle routing problem with time windows (TDVRTW) considered in this paper is NP-hard, the computation time to solve an instance optimality increases exponentially with the size of the instance. Because of this reason, in this paper, an attempt has been made to develop a meta-heuristic and compare it with existing meta-heuristics.

3. Components of Genetic Algorithm

Genetic Algorithm was first developed by J. Holland at the University of Michigan in 1975. Solutions to a combinatorial problem are encoded as chromosomes. The chromosomes are evaluated for their fitness by an evaluation function, and good properties of a generation of solutions are propagated to the following generations.

The genetic algorithms typically have the following structure. A typical genetic algorithm (GA) will start with a set of chromosomes called the initial population. Each chromosome represents a solution to the problem. The initial population is either randomly generated (in which case it would take longer time for the algorithm to converge to the solution) or generated using some form of heuristics (in which case the population is already closer to the solution, and would hence take less time to converge). The next step in the GA method is the selection mechanism where the selection of the prospective parents based on their fitness is carried out and which is computed by the evaluation function. The selected parent chromosomes will then be recombined via the crossover operation to create their offspring. After the crossover process, the next step will be to mutate a small number of the newly obtained offspring, in order to introduce a level of randomness that will preserve the GA from converging to a local optimum. A mutation is typically a random swap in a gene sequence, or a random negation of a bit if the offspring is bit-encoded. Finally, new population will then be formed by substituting the offspring in place of their corresponding chromosomes.

The genetic algorithm will continue through this process until a stopping criterion is met, which can be one of the following:

- predefined number of generations has been produced;
- there was no improvement in the population, which would mean that the GA has found an optimal solution;
- a predefined level of fitness has been reached.

Tournament Selection

In this study, tournament selection method is used as the method of selecting the parent chromosomes for crossover operation. In tournament selection, two identical (though differently ordered) copies of the population are kept. In every generation, adjacent chromosomes in one copy of the population pair by pair are compared, and the chromosome with greater fitness value (lower fitness value in this case because we are minimizing the total distance travelled) is selected. Then the second copy of the population in treated the same way to select the other half of the selected population.

4. Genetic Algorithm for Solving Time-Dependent Vehicle Routing Problem with Time Windows

In this section, the methodology used to develop the genetic algorithm-based solution for the time-dependent vehicle routing problem with time windows (TDVRPTW) is given. A new crossover technique called the random sequence insertion-based crossover (RSIX) is described in detail.

4.1. Chromosome Representation

The representation of the chromosome as a set of genes representing the nodes in a route is shown in **Figure 2**.

In **Figure 2**, *n* is the total number of customers. Each chromosome consists of a set of genes. In the chromosome shown in the **Figure 2**, the genes c1, *c*2, …, *cn* are defined as Customer IDs/Nodes. A road chromosome contains a list of elements or genes. Every chromosome is initialized as *the route which contains the* source *location* and *the* destination *location* at the start and end of the array, respectively. Each chromosome is a solution *path*.

A crossover operator is a major process of producing offspring from the current population. There are many methods for crossover operation according to different problems. In this paper, "Random Sequence Insertion-Based Crossover" Kumar and Panneerselvam [53] method is used, which is explained in the next section.

4.2. Random Sequence Insertion-Based Crossover (RSIX)

Consider two parent chromosomes with seven genes in each of them as shown in **Figure 3(a)**. The gene elements in each of the par *nt* chromosomes are from 1 to 7.

The steps of the Random Sequence Insertion-based Crossover (RIX) method are presented below.

Step 1: Two chromosomes are randomly chosen as parents.

Step 2: Generate two crossover points, which will lead to three segments in each chromosome as shown in **Figure 3(a)**.

Step 3: Next, swap the middle crossover genes segment, as in **Figure 3(b)**.

Step 4: Next validity checking is carried out, taking into consideration the constraints of VRPTW, with each demand point (customer node) allowed to be visited once, and if we assume triangular inequality, *i.e.* $\forall(i, j)$, $d_{0i} < d_{0j} + d_{ji}$, then the time window constraint is satisfied, too, except waiting time is likely to be incurred. Retain the crossover gene section, and then removing the same number of the gene in their parent, such as **Figure 3(c)**.

Step 5: This results in getting two new offspring with crossover gene segment and they are saved to the next generation as shown in **Figure 3(d)**.

- The new offspring are tested for fitness values.
- The smaller the "fitness-value", the stronger road chromosome is obtained.

4.3. Evaluation of Fitness Function

According to Sivasankaran and Shahabudeen [45], the fitness function of the chromosome is obtained by assigning the nodes serially from left to right from its ordered vector into customer IDs or nodes for a given travel route.

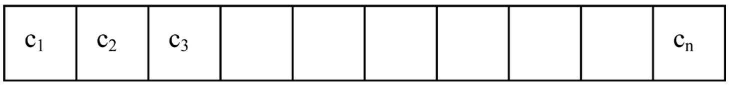

Figure 2. Chromosome sample.

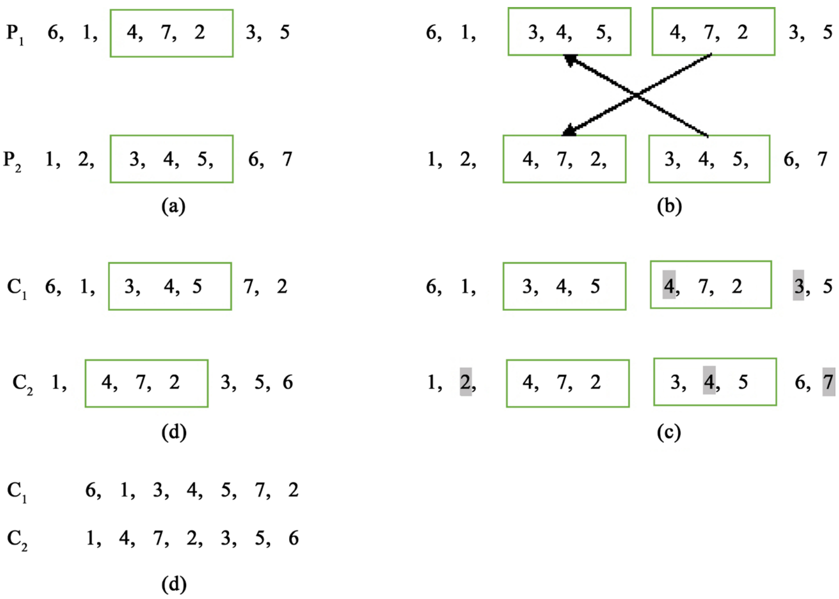

Figure 3. Crossover operation.

While assigning a customer ID into a node, that road pertaining to all the models should be assigned to the same node. If a road is available in only one model then that can be independently assigned to the current node.

Every road or route chromosome has its own fitness value, defined as fitness value, where

1) Fitness value = The sum of route distance cost for every road in a route chromosome.

2) The smaller the "fitness value" value, the stronger road chromosome is obtained.

3) Every chromosome is initialized as nodes which contain the source location and the destination location, which are fixed at the start and end of the array.

4.4. Genetic Algorithm (SNRPGA)

The steps of the proposed genetic algorithm (SNRPGA) for the time dependent vehicle routing problem with time windows are presented below.

Step 1: Input the following:

- Number of customer nodes (n).
- Number of vehicles (k).
- Capacity of the vehicles (a).
- Set Generation Count (GC) = 1.
- Maximum number of generations to be carried out (MNG) = 1000.

Step 2: Generate a random initial population (L) of 100 (N) chromosomes (suitable solutions routes for the problem).

Step 3: Evaluate the fitness function f(x) of each chromosome in the population L.

Step 4: Selection.

Sort the population L by the objective function (fitness function) value in the ascending order, since the objective of the study is minimization of the total distance travelled. Copy a top 30% of the population to form a subpopulation S rounded to the whole number. Smaller fitness value is preferred here.

Step 5: Randomly select any two unselected parent chromosomes from the subpopulation S. Let them be *c*1 and *c*2 using tournament selection.

Step 5.1: Perform two-point random Cross-Over using the random sequence insertion-based crossover (RSIX) described in the earlier section among the chromosomes *c*1 and *c*2 to obtain their offspring d1 and d2 by assuming a crossover probability of 0.7.

Step 5.2: Perform mutation on each of the offspring using a mutation probability of 0.3.

Step 5.3: Evaluate the fitness function with respect to the total distance travelled and number of vehicles utilized value for each of the offspring d1 and d2.

Step 5.4: Replace the parent chromosomes *c*1 and *c*2 in the population with the offspring d1 and d2, respectively, if the fitness function of the offspring is less than that of the parent chromosomes.

Step 6: Increment the generation count (GC) by 1.

i.e., GC = GC + 1.

Step 7: If GC ≤ MNG, then go to step 4, else go to step 8.

Step 8: The topmost chromosome in the last population serves as the solution for implementation.

Print the tour along with the total distance travelled and number of vehicles used.

Step 9: Stop.

5. Comparison of Proposed Algorithm with Existing Algorithms

The time dependent vehicle routing problem with time windows plugin for HeuristicLab using genetic algorithm was implemented using Visual C#. The standard test data used as input for the solving the TDVRPTW using genetic algorithm (GA) is the Solomon's 56 benchmark instances.

The input parameters of the vehicle routing problem with time windows are listed below.

1) Number of nodes (customer locations) = 100.

2) Vehicle capacity = C1 = 200, C2 = 700, R1 = 200, R2 = 1000, RC1 = 200, RC2 = 1000.

3) Number of Vehicles = 25.

4) The demand at each customer location is given.

5) Distance of the customer location from the depot and from each other is given in the distance matrix.

The comparison of proposed algorithm is done in two stages as listed below.

- Comparison of the proposed algorithm with five existing algorithms in terms of number of vehicles utilized.
- Comparison of the proposed algorithm with five existing algorithms in terms of total distance travelled.

The experiments were run on a computer with Windows 8.1 OS and an Intel Core i3, 1.70 GHz Processor (CPU). The 56 benchmark instances are divided into 6 groups or classes C1, C2, R1, R2, RC1, RC2.

5.1. Comparisons of Algorithms in Terms of Number of Vehicles Utilized

In this section, the proposed genetic algorithm is compared with five existing algorithms to solve the vehicle routing problem with time windows using a complete factorial experiment with two factors, viz. "problem Size" and "Algorithm". The primary objective is to minimise the number of vehicles and the secondary objective is to minimise the total distance travelled, which is the same objective of the study of Figliozzi [46] which uses constant speed problems. The cumulative values show the total average number of vehicles utilized for all the problem classes and the total average distance travelled for all the problem classes. The number of levels for the problem size is 6, viz. C1, C2, R1, R2, RC1, RC2 from Solomon's benchmark instances. The number levels for the algorithm is 6, viz. Solomon [9], CTA by Thompson [47], GIDEON by Thangiah [48], GenSAT by Thangiah [16]), SNRPGA [Proposed], TABU by Potvin *et al*. [49] in terms of the total distance travelled. The number of replications under each experimental combination of the factorial experiment is 4. The results obtained as per the factorial experiment are shown in **Table 1**.

The model of ANOVA is given as below:

$$Y_{ijk} = \mu + A_i + B_j + AB_{ij} + e_{ijk}$$

where,

Y_{ijk} is the number of vehicles utilized w.r.t the *k*th replication under the *i*th treatment of factor A (Problem Size) and the *j*th treatment of factor B (Algorithm), *i.e.* j^{th} Algorithm.

μ is the overall mean of the response variable.

A_i is the effect of the *i*th treatment of factor A (Problem Size) on the response variable.

B_j is the effect of the *j*th treatment of factor B (Algorithm) on the response variable.

AB_{ij} is the interaction effect of the *i*th Problem Size and *j*th Algorithm on the response variable.

e_{ijk} is the random error associated with the kth replication under the ith Problem Size and the jth Algorithm.

In this model, Factor A (Problem Size/Problem Class) is a random factor and the Factor B (algorithm) is a fixed factor. Since the factor A is a random factor, the interaction factor AB_{ij} is also a random factor. The replications are always random and the number of replications under each experimental combination is k. The derivation of the expected mean square (EMS) is given in Panneerselvam [51]. To test the effect of A_i as well as AB_{ij}, the respective F ratio is formed by dividing the mean sum of squares of the respective component (A_i or AB_{ij}), by the mean sum of squares of error. The F ratio of the component B_j is formed by dividing its mean sum of squares by the mean sum of squares of AB_{ij}.

The alternative hypotheses of the model are as given below.

H_1: There are significant differences between the different pairs of treatments of Factor A (Problem Size) in terms of the number of vehicles utilized.

H_1: There are significant differences between the different pairs of treatments of Factor B (Algorithm) in terms of the number of vehicles utilized.

H_1: There are significant differences between the different pairs of interaction between Factor A and Factor B in terms of number of vehicles utilized.

The ANOVA results of the data given in **Table 1** are shown in **Table 2**. From the ANOVA results shown in **Table 2**, one can infer that the factors "Problem Size" and "Algorithm" and "Interaction of "Problem size" and "Algorithm" have significant effects on the response variable "Number of Vehicles Utilized". Since there are significant differences among the algorithms, the best algorithm is obtained using Duncan's multiple range test.

Table 1. Results of number of vehicles utilized.

		Algorithm (B)					
	Replication	Solomon	CTA	Gideon	GenSAT	SNRPGA	Tabu
Problem size/problem class (A)	R1 1	21	19	20	18	10	18
	R1 2	19	17	17	17	10	20
	R1 3	14	15	13	13	10	16
	R1 4	15	15	15	14	10	16
	C1 1	10	10	10	10	9	10
	C1 2	10	10	10	10	9	10
	C1 3	10	10	10	10	9	11
	C1 4	10	10	10	10	10	10
	RC1 1	16	16	15	14	11	17
	RC1 2	15	14	14	13	10	15
	RC1 3	16	12	14	14	11	13
	RC1 4	13	15	12	13	10	16
	R2 1	4	4	4	4	3	6
	R2 2	4	4	4	4	3	5
	R2 3	3	3	3	3	2	4
	R2 4	4	3	3	3	2	4
	C2 1	3	3	3	3	3	3
	C2 2	3	3	3	3	3	4
	C2 3	3	3	3	3	3	4
	C2 4	4	3	3	3	3	4
	RC2 1	4	4	4	4	3	7
	RC2 2	4	4	4	4	3	7
	RC2 3	4	4	4	4	3	4
	RC2 4	5	4	4	3	3	6

The standard error used in this test is computed as shown below using the mean sum of squares of the interaction terms (Problem Size × Algorithm) and the number of replications under each of the algorithms (24).

$$SE = \left(MSS_{AB}/n\right)^{0.5} = \left(4.37/24\right)^{0.5} = 0.4267$$

The least significant ranges (LSR) are calculated from the significant ranges of Duncan's multiple range tests table for $\alpha = 0.05$ and 25 degrees of freedom as shown in **Table 3**.

The results of Duncan's multiple range test are shown in **Figure 4**. In this figure, the algorithms are arranged as per the descending order of their mean number of vehicles utilized from left to right. From this figure, it is clear that the proposed algorithm "SNRPGA" is significantly different from all other algorithms and the mean number of vehicles utilized given by this algorithm is the least. Hence, the proposed algorithm "SNRPGA" is the best algorithm among all the algorithms used in the comparison presented in this section in terms of the number of vehicles used.

5.2. Comparison of Algorithms in Terms of Total Distance Travelled

The proposed GA-based meta-heuristic (SNRPGA) is compared with five other existing meta-heuristics, viz. Solomon [9], GIDEON by Thangiah [48], GenSAT (Thangiah [16]), PTABU (Potvin *et al*. [49]), CTA (Thompson and Psaraftis, [47]) in terms of total distance travelled using a complete factorial experiment with two factors, viz. "Problem Size" and "Algorithm". The number of levels for the problem size is 6, viz. C1, C2, R1, R2, RC1, RC2 from Solomon's benchmark instances. The number of levels for "Algorithm" is 6 as already stated above. The number of replication under each experimental combination is 4. The results of the factorial experiment in terms of the total distance travelled are shown in **Table 4**. The application of ANOVA to the data given in **Table 4** gives the results as shown in **Table 5**.

The model of ANOVA is given as below:

$$Y_{ijk} = \mu + A_i + B_j + AB_{ij} + e_{ijk}$$

where,

Y_{ijk} is the total distance travelled w.r.t the *k*th replication under the *i*th treatment of factor A (Problem Size) and the *j*th treatment of factor B (Algorithm).

μ is the overall mean of the response variable total distance travelled.

A_i is the effect of the *i*th treatment of factor A (Problem Size) on the response variable.

B_j is the effect of the *j*th treatment of factor B (Algorithm) on the response variable.

AB_{ij} is the interaction effect of the *i*th Problem Size and *j*th Algorithm on the response variable.

e_{ijk} is the random error associated with the *k*th replication under the *i*th Problem Size and the *j*th Algorithm.

Table 2. Analysis of variance for number of vehicles utilized.

Source of variation	Sum of squares	Degrees of freedom	Mean square	Calculated F ratio	F ratio ($\alpha = 0.05$)	Remark
Problem size (A)	3585.035	5	717.007	527.59897	2.32	Significant
Algorithm (B)	139.29	5	27.857	6.3746	2.32	Significant
Problem size × algorithm (AB)	109.25	25	4.37	3.216	1.64	Significant
Error	146.750	108	1.359			
Total	3980.326	143				

Table 3. Least significant ranges for various treatments.

Range j	Significant range	Standard error	LSR = significant range × standard error
2	2.900	0.4267	1.2383
3	3.070	0.4267	1.3100
4	3.150	0.4267	1.3441
5	3.225	0.4267	1.3761
6	3.275	0.4267	1.3974

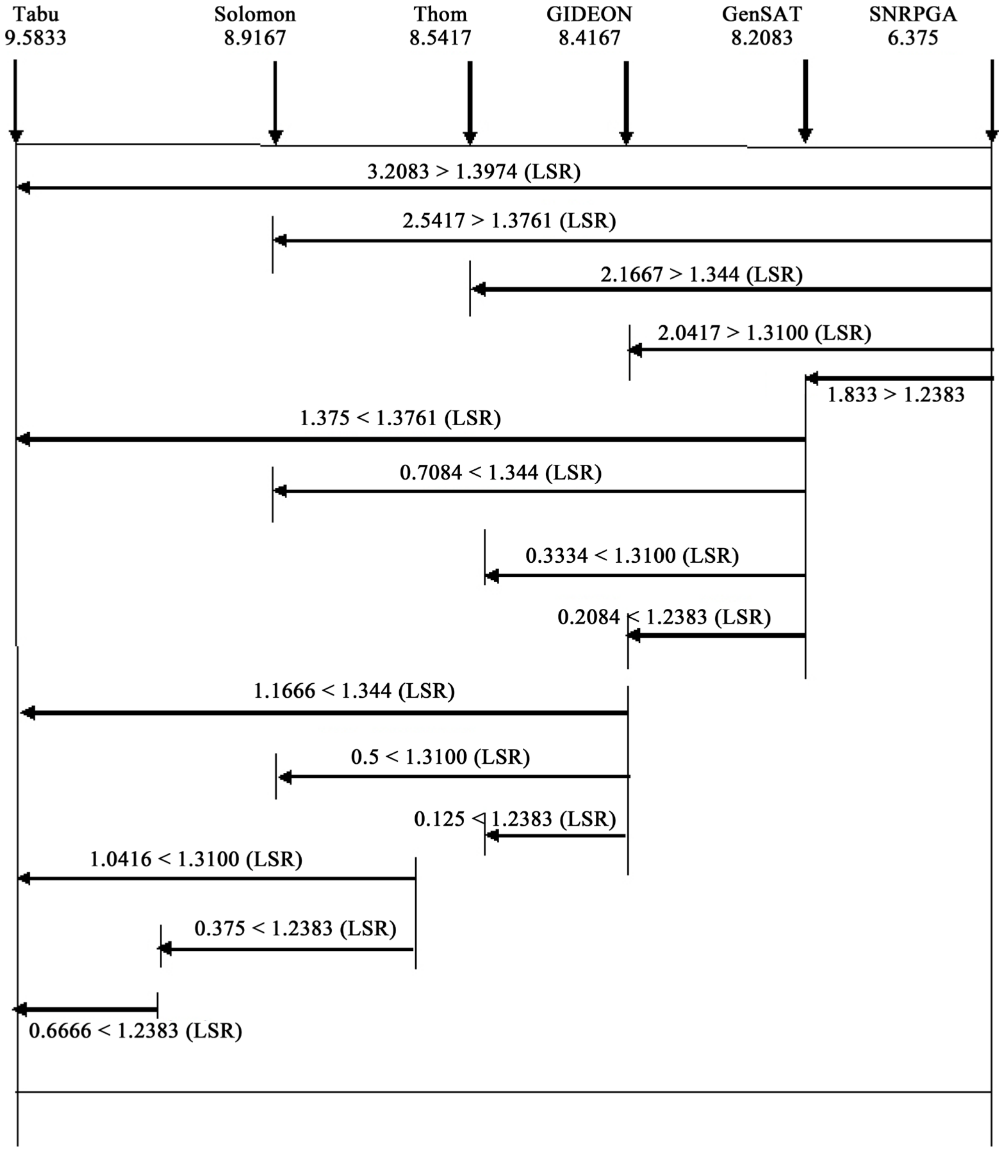

Figure 4. Duncan's multiple range test w.r.t algorithm in terms of number of vehicles utilized.

In this model, the factor A is a random factor and the factor B is a fixed factor. Since the factor A is a random factor, the interaction factor is also a random factor. The replications are always random and the number of replications under each experimental combination is *k*. The derivation of the expected mean square (EMS) is given in Panneerselvam [51]. To test the effect of A_i as well as AB_{ij} the respective F ratio is formed by dividing the mean sum of squares of the respective component (*Ai* or AB_{ij}), by the mean sum of squares of error. The F ratio of the component B_j is formed by dividing its mean sum of squares by the mean sum of squares of AB_{ij}.

The alternative hypothesis of the model is stated as below:

H_1: There are significant differences between the different pairs of treatments of Factor A (Problem Size) in terms of the total distance travelled.

H_1: There are significant differences between the different pairs of treatments of Factor B (Algorithm) in terms of the total distance travelled.

H_1: There are significant differences between the different pairs of interaction between Factor A and Factor B in terms of total distance travelled.

From the ANOVA results shown in **Table 5**, one can infer that the factors "Algorithm" and "Problem Size" have significant effects on the total distance travelled. Since, there is a significant difference among the 6

Table 4. Results of Algorithms in terms of total distance travelled.

	Algorithm (Factor B)						
	Replication	Solomon	CTA	GIDEON	GenSAT	SNRPGA	TABU
Problem size/problem class (Factor A)	1 R1	1873	1734	1700	1644	1016	1749
	2 R1	1843	1881	1549	1493	963	1527
	3 R1	1484	1530	1448	1442	1096	1340
	4 R1	1673	1535	1363	1350	1000	1542
	1 C1	968	934	904	829	829	914
	2 C1	1059	956	926	829	829	1005
	3 C1	1282	1130	928	835	828	907
	4 C1	904	926	957	835	825	946
	1 RC1	1867	1851	1767	1669	1292	1766
	2 RC1	1760	1644	1569	1557	1161	1649
	3 RC1	1673	1465	1612	1602	1230	1632
	4 RC1	1922	1809	1608	1420	1128	1537
	1 R2	1741	1786	1478	1354	748	1344
	2 R2	1730	1736	1279	1176	820	1248
	3 R2	1578	1309	1274	8338	720	1187
	4 R2	1542	1392	1269	1052	773	1160
	1 C2	731	664	753	591	589	737
	2 C2	811	653	756	707	588	759
	3 C2	758	684	855	791	588	742
	4 C2	730	670	803	685	588	691
	1 RC2	2103	1959	1823	1294	970	1577
	2 RC2	1799	1858	1594	1291	794	1404
	3 RC2	2134	1521	1530	1389	810	1494
	4 RC2	1632	1988	1501	1213	805	1392

Table 5. Analysis of variance for total distance travelled.

Source of variation	Sum of squares	Degrees of freedom	Mean sum of squares	Calculated F ratio	F ratio (α = 0.05)	Remark
Problem size (A)	17,565,288.891	5	3,513,057.778	9.536	2.30	Significant
Algorithm (B)	6,160,662.029	5	1,232,132.406	2.93	2.30	Significant
Algorithm × problem size (AB)	10,515,346.157	25	420,613.846	1.141	1.61	Insignificant
Error	39,787,307.795	108	368,400.998			
Total	74,028,604.872	143				

algorithms compared in terms of the total distance travelled, Duncan's multiple range test is next conducted to identify the best algorithm by arranging the algorithms in the descending order of their mean total distance travelled from left to right as shown in **Figure 4**.

The treatment means for the Factor B (Algorithm) in terms of the total distance travelled are arranged in the descending order from left to right. The standard error for the performance measure is calculated using the formula and found to be 132.3842. One can notice the fact that the mean sum of squares of the interaction term AB is used in estimating the standard error (SE), because the F ratio for the factor "Algorithm" is obtained by dividing its mean sum of squares by the mean sum of squares of the interaction term AB_{ij} (Panneerselvam [51]).

The least significant ranges (LSR values) are calculated from the significant ranges of Duncan's multiple range test table for α = 0.05 and 25 degrees of freedom. These are shown in **Table 6**.

Next by comparing the actual differences between the various treatment means of Factor B (Algorithm) with the corresponding calculated LSR values as shown in **Figure 5**, it is found that the proposed algorithm (SNRPGA) is significantly different from all other existing algorithms considered in this research. Further, the mean of the

Table 6. Least significant ranges for various treatments.

No. of treatments − 1 (j)	significant range	Standard error	LSR = significant range × standard error
2	2.900	132.3842	383.9142
3	3.070	132.3842	406.4195
4	3.150	132.3842	417.0102
5	3.225	132.3842	426.9391
6	3.275	132.3842	433.5583

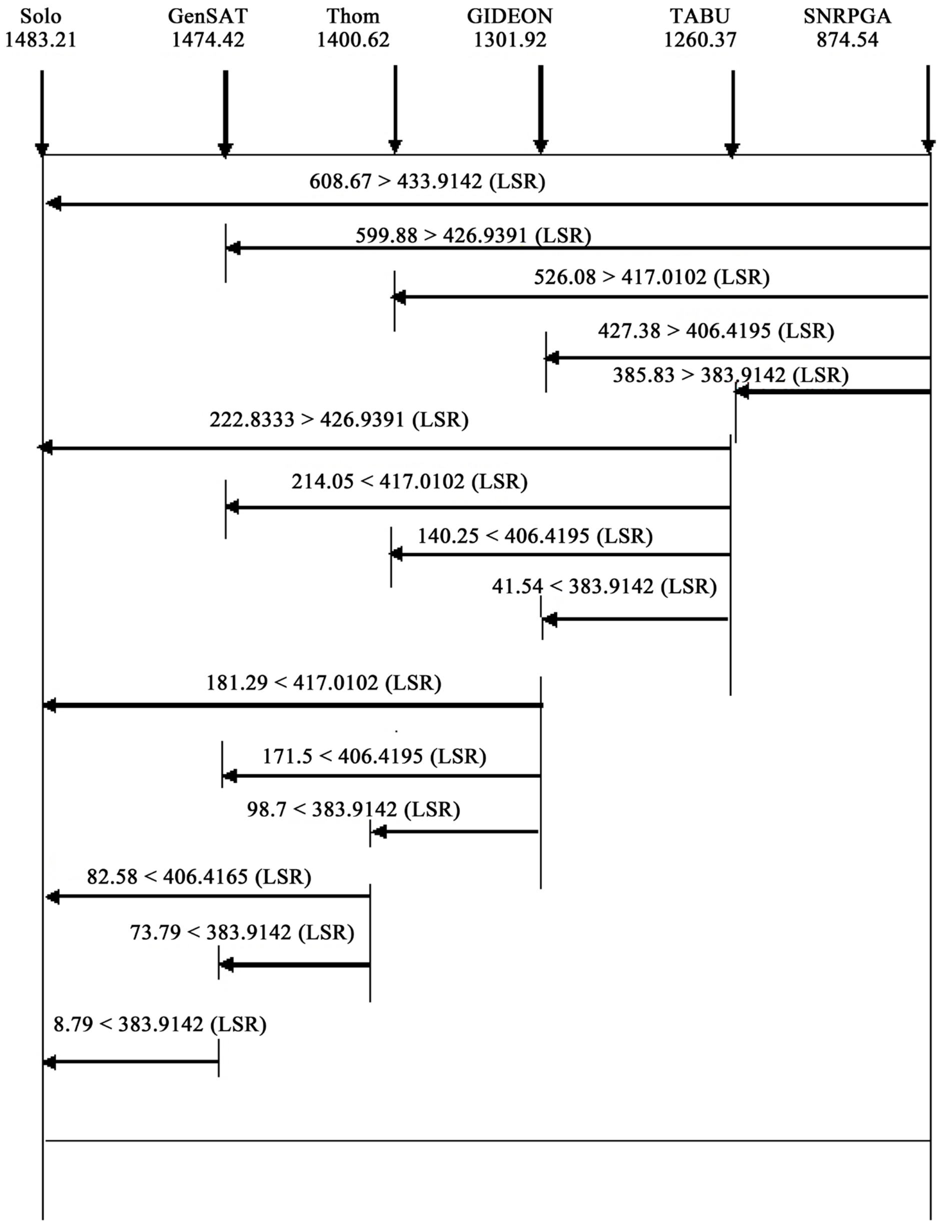

Figure 5. Duncan's multiple range test w.r.t algorithm in terms of total distance travelled.

total distance travelled for the proposed algorithm is the least among all the algorithms. Hence, it is concluded that the proposed algorithm (SNRPGA) is the best among all the algorithms with respect to the total distance travelled.

6. Conclusion and Suggestions for Future Research

In this research, a GA based meta-heuristic is developed using Random Sequence-based Insertion Crossover (RSIX) method (SNRPGA) for solving the time dependent vehicle routing problem (TDVRP) with time windows. It is compared with five other existing mete-heuristics in terms of two performance measures, viz. number of vehicles utilized and total distance travelled. Through a complete factorial experiment with two factors, viz. "Problem Size" and "Algorithm", it is proved that there are significant differences among the algorithms in terms of the number of vehicles utilized. So, in the next stage using Duncan's multiple range test, it is found that the proposed algorithm "SNRPGA" is the best among all the algorithms in terms of minimizing the number of vehicles utilized.

Next, through another factorial experiment with two factors, viz. "Problem Size" and "Algorithm", it is proved that there are significant differences among the algorithms in terms of the total distance travelled. So, in the next stage using Duncan's multiple range test, it is found that the proposed algorithm "SNRPGA" is the best among all the algorithms in terms of minimizing the total distance travelled.

This study can be useful for planning the supplier site pickups by e-commerce companies, taking into consideration of traffic conditions during different periods of the day with time window requirements of the suppliers. Future researchers can implement the TDVRP using other meta-heuristics and compare the efficiencies of the various meta-heuristics. The Solomon's benchmark instance is got from SINTEF [52].

Acknowledgements

The authors wish to express their deepest gratitude to the researchers Mr. Andreas Scheibenpflug and Mr. Philipp Fleck of FH Hagenberg, Austria, for their help.

References

[1] Bullnheimer, B., Hartl, R.F. and Strauss, C. (1997) Applying the Ant System to the Vehicle Routing Problem. Department of Management Science, University of Vienna.

[2] Dantzig, G. and Ramser, J. (1959) The Truck Dispatching Problem. *Management Science*, **6**, 80-91. http://dx.doi.org/10.1287/mnsc.6.1.80

[3] Clarke, G. and Wright, J.R. (1964) Scheduling of Vehicle Routing Problem from a Central Depot to a Number of Delivery Points. *Operations Research*, **12**, 568-581. http://dx.doi.org/10.1287/opre.12.4.568

[4] Laporte, G. (1992) The Vehicle Routing Problem: An Overview of Exact and Approximate Algorithms. *European Journal of Operational Research*, **59**, 345-358. http://dx.doi.org/10.1016/0377-2217(92)90192-C

[5] Gendreau, M., Laporte, G. and Potvin, J.-Y. (2002) Metaheuristics for the VRP. In: Toth, P. and Vigo, D., Eds., *The Vehicle Routing Problem*, SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, 129-154. http://dx.doi.org/10.1137/1.9780898718515.ch6

[6] Cordeau, J.-F., Gendreau, M., Hertz, A., Laporte, G. and Sormany, J.S. (2005) New Heuristics for the Vehicle Routing Problem. In: Langevin, A. and Riopel, D., Eds., *Logistics Systems*: *Design and Optimization*, Springer, New York, 279-297. http://dx.doi.org/10.1007/0-387-24977-x_9

[7] Fu, L. (2002) Scheduling Dial-a-Ride Paratransit under Time-Varying, Stochastic Congestion. *Transportation Research Part B*, **36**, 485-506. http://dx.doi.org/10.1016/S0191-2615(01)00014-5

[8] Meng, Q., Lee, D.H. and Cheu, R. (2005) Multiobjective Vehicle Routing and Scheduling Problem with Time Window Constraints in Hazardous Material Transportation. *Transportation Engineering*, **131**, 699-707. http://dx.doi.org/10.1061/(ASCE)0733-947X(2005)131:9(699)

[9] Solomon, M. (1987) Algorithms for the Vehicle Routing and Scheduling Problem with Time Window Constraints. *Operations Research*, **35**, 254-265. http://dx.doi.org/10.1287/opre.35.2.254

[10] Malandraki, C. and Daskin, M.S. (1992) Time-Dependent Vehicle-Routing Problems—Formulations, Properties and Heuristic Algorithms. *Transportation Science*, **26**, 185-200. http://dx.doi.org/10.1287/trsc.26.3.185

[11] Hill, A.V. and Benton, W.C. (1992) Modeling Intra-City Time-Dependent Travel Speeds for Vehicle Scheduling Problems. *Journal of the Operational Research Society*, **43**, 343-351. http://dx.doi.org/10.1057/jors.1992.49

[12] Ichoua, S., Gendreau, M. and Potvin, J.Y. (2003) Vehicle Dispatching with Time-Dependent Travel Times. *European Journal of Operational Research*, **144**, 379-396. http://dx.doi.org/10.1016/S0377-2217(02)00147-9

[13] Le Bouthillier, A. and Crainic, T.G. (2005) A Cooperative Parallel Meta-Heuristic for the Vehicle Routing Problem with Time Windows. *Computers & Operations Research*, **32**, 1685-1708. http://dx.doi.org/10.1016/j.cor.2003.11.023

[14] Blanton, J.L. and Wainwright, R.L. (1993) Multiple Vehicle Routing with Time and Capacity Constraints using Genetic Algorithms. In: Forrest, S., Ed., *Proceedings of the Fifth International Conference on Genetic Algorithms*, Morgan Kaufmann Publishing, San Francisco, 452-459.

[15] Berger, J., Salois, M. and Begin, R.A. (1998) Hybrid Genetic Algorithm for the Vehicle Routing Problem with Time Windows. *Lecture Notes in Artificial Intelligence*, **1418**, 114-127. http://dx.doi.org/10.1007/3-540-64575-6_44

[16] Thangiah, S.R., Osman, I.H. and Sun, T. (1995) Hybrid Genetic Algorithms, Simulated Annealing and Tabu Search Methods for Vehicle Routing Problems with Time Windows. Technical Report UKC/OR94/4, Institute of Mathematics and Statistics, University of Kent, Canterbury.

[17] Potvin, J.-Y. and Bengio, S. (1996) The Vehicle Routing Problem with Time Windows—Part II: Genetic Search. *INFORMS Journal on Computing*, **8**, 165-172. http://dx.doi.org/10.1287/ijoc.8.2.165

[18] Zhu, Z. and Kenny, Q.L. (2000) A New Genetic Algorithm for VRPTW. National University of Singapore, Singapore.

[19] 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*, **10**, 431-449. http://dx.doi.org/10.1142/S021821300100060X

[20] Homberger, J. and Gehring, H. (2005) A Two-Phase Hybrid Meta-Heuristic for the Vehicle Routing Problem with Time Windows. *European Journal of Operational Research*, **162**, 220-238. http://dx.doi.org/10.1016/j.ejor.2004.01.027

[21] David, M. and Bräysy, O. (2005) Active Guided Evolution Strategies for Large-Scale Vehicle Routing Problems with Time Windows. *Computers & Operations Research*, **32**, 1593-1614. http://dx.doi.org/10.1016/j.cor.2003.11.017

[22] Russell Robert, A. and Chiang, W.-C. (2006) Scatter Search for the Vehicle Routing Problem with Time Windows. *European Journal of Operational Research*, **169**, 606-622. http://dx.doi.org/10.1016/j.ejor.2004.08.018

[23] Alba, E. and Dorronsoro, B. (2005) The Exploration/Exploitation Tradeoff in Dynamic Cellular Genetic Algorithms. *IEEE Transactions on Evolutionary Computation*, **9**, 126-142. http://dx.doi.org/10.1109/TEVC.2005.843751

[24] Glover, F. (1993) Tabu Thresholding: Improved Search by Non-Monotonic Trajectories. Working Paper, Graduate School of Business, University of Colorado, Boulder, 80309-0419.

[25] Glover, F. (1990) Tabu Search-Part II. *ORSA Journal on Computing*, **2**, 4-32. http://dx.doi.org/10.1287/ijoc.2.1.4

[26] Glover, F., Taillard, E. and de Werra, D. (1993) A User's Guide to Tabu Search. *Annals of Operations Research*, **41**, 3-28. http://dx.doi.org/10.1007/BF02078647

[27] Li, H. and Lim, A. (2001) A Metaheuristic for the Pickup and Delivery Problem with Time Windows. *IEEE Proceedings of the* 13*th International Conference on Tools with Artificial Intelligence*, Dallas, 7-9 November 2001, 160-167. http://dx.doi.org/10.1109/ictai.2001.974461

[28] Ichoua, S., Gendreau, M. and Potvin, J.Y. (2007) Planned Route Optimization for Real-Time Vehicle Routing. In: Zeimpekis, V., Tarantilis, C.D., Giaglis, G.M. and Minis, I., Eds., *Dynamic Fleet Management*, Springer, New York, 1-18. http://dx.doi.org/10.1007/978-0-387-71722-7_1

[29] Lau, H.C., Sim, M. and Teo, K.M. (2003) Vehicle Routing Problem with Time Windows and a Limited Number of Vehicles. *European Journal of Operational Research*, **148**, 559-569. http://dx.doi.org/10.1016/S0377-2217(02)00363-6

[30] Malandraki, C. (1989) Time Dependent Vehicle Routing Problems: Formulations, Solution Algorithms and Computational Experiments. PhD Dissertation, Northwestern University, Evanston.

[31] Ahn, B.H. and Shin, J.Y. (1991) Vehicle-Routing with Time Windows and Time-Varying Congestion. *Journal of the Operational Research Society*, **42**, 393-400. http://dx.doi.org/10.1057/jors.1991.81

[32] Malandraki, C. and Dial, R.B. (1996) A Restricted Dynamic Programming Heuristic Algorithm for the Time Dependent Travelling Salesman Problem. *European Journal of Operational Research*, **90**, 45-55. http://dx.doi.org/10.1016/0377-2217(94)00299-1

[33] 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. http://dx.doi.org/10.1287/trsc.31.2.170

[34] Fleischmann, B., Gietz, M. and Gnutzmann, S. (2004) Time-Varying Travel Times in Vehicle Routing. *Transportation Science*, **38**, 160-173. http://dx.doi.org/10.1287/trsc.1030.0062

[35] Jung, S. and Haghani, A. (2001) Genetic Algorithm for the Time-Dependent Vehicle Routing Problem. *Transportation Research Record*, **1771**, 164-171. http://dx.doi.org/10.3141/1771-21

[36] Donati, A.V., Montemanni, R., Casagrande, N., Rizzoli, A.E. and Gambardella, L.M. (2008) Time Dependent Vehicle Routing Problem with a Multi Ant Colony System. *European Journal of Operational Research*, **185**, 1174-1191. http://dx.doi.org/10.1016/j.ejor.2006.06.047

[37] Soler, D., Albiach, J. and Martínez, E. (2009) A Way to Optimally Solve a Time-Dependent Vehicle Routing Problem with Time Windows. *Operations Research Letters*, **37**, 37-42. http://dx.doi.org/10.1016/j.orl.2008.07.007

[38] Dabia, S., Van Woensel, T. and De Kok, A.G. (2010) A Dynamic Programming Approach to Multi-Objective Time-Dependent Capacitated Single Vehicle Routing Problems with Time Windows. Working Paper 313, School of Industrial Engineering, Eindhoven University of Technology, Eindhoven.

[39] Kok, A.L. (2010) Congestion Avoidance and Break Scheduling within Vehicle Routing. PhD Thesis, University of Twente, Enschede.

[40] Figliozzi, M.A. (2009) A Route Improvement Algorithm for the Vehicle Routing Problem with Time Dependent Travel Times. *Proceedings of the* 88*th Transportation Research Board Annual Meeting*, Washington DC, 11-15 January 2009.

[41] van Woensel, T., Kerbache, L., Peremans, H. and Vandaele, N. (2008) Vehicle Routing with Dynamic Travel Times: A Queueing Approach. *European Journal of Operational Research*, **186**, 990-1007. http://dx.doi.org/10.1016/j.ejor.2007.03.012

[42] Kok, A.L., Hans, E.W. and Schutten, J.M.J. (2011) Optimizing Departure Times in Vehicle Routes. *European Journal of Operational Research*, **210**, 579-587. http://dx.doi.org/10.1016/j.ejor.2010.10.017

[43] Kuo, Y., Wang, C.-C. and Chuang, P.-Y. (2009) Optimizing Goods Assignment and the Vehicle Routing Problem with Time-Dependent Travel Speeds. *Computers and Industrial Engineering*, **57**, 1385-1392. http://dx.doi.org/10.1016/j.cie.2009.07.006

[44] Bettinelli, A., Ceselli, A. and Righini, G. (2011) A Branch-and-Cut-and-Price Algorithm for the Multi-Depot Heterogeneous Vehicle Routing Problem with Time Windows. *Transportation Research Part C*, **19**, 723-740. http://dx.doi.org/10.1016/j.trc.2010.07.008

[45] Sivasankaran, P. and Shahabudeen, P. (2013) Genetic Algorithm for Concurrent Balancing of Mixed-Model Assembly Lines with Original Task Times of Models. *Intelligent Information Management*, **5**, 84-92. http://dx.doi.org/10.4236/iim.2013.53009

[46] Figliozzi, M.A. (2012) The Time Dependent Vehicle Routing Problem with Time Windows: Benchmark Problems, an Efficient Solution Algorithm, and Solution Characteristics. *Transportation Research Part E*, **48**, 616-636. http://dx.doi.org/10.1016/j.tre.2011.11.006

[47] Thompson, P.M. and Psaraftis, H. (1989) Cyclic Transfer Algorithms for Multi-Vehicle Routing and Scheduling Problems. *Operations Research*, **41**, 935-946. http://dx.doi.org/10.1287/opre.41.5.935

[48] Thangiah, S.R., Nygard, K.E. and Juell, P.L. (1991) GIDEON: A Genetic Algorithm System for Vehicle Routing with Time Windows. In: *Proceedings of the 7th IEEE Conference on Artificial Intelligence Applications*, IEEE Computer Society Press, Los Alamitos, 322-328. http://dx.doi.org/10.1109/caia.1991.120888

[49] Potvin, J.-Y., Kervahut, T., Garcia, B.-L. and Rousseau, J.-M. (1992) A Tabu Search Heuristic for the Vehicle Routing Problem with Time Windows. Technical Report, CRT-855, Centre de Recherchesur les Transports, Universite de Montreal, Montreal.

[50] Haghani, A. and Jung, S. (2005) A Dynamic Vehicle Routing Problem with Time-Dependent Travel Times. *Computers & Operations Research*, **32**, 2959-2986. http://dx.doi.org/10.1016/j.cor.2004.04.013

[51] Panneerselvam, R. (2012) Design and Analysis of Experiments. PHI Learning Pvt. Limited, New Delhi.

[52] SINTEF. Solomon's Benchmark Instances. https://www.sintef.no/projectweb/top/vrptw/solomon-benchmark/

[53] Kumar, S.N. and Panneerselvam, R. (2015) A Time-Dependent Vehicle Routing Problem with Time Windows for E-Commerce Supplier Site Pickups Using Genetic Algorithm. *Intelligent Information Management*, **7**, 181-194. http://dx.doi.org/10.4236/iim.2015.74015

[54] Chu, C.-P., Chen, C.-C., Ho, M.-C. and Wang, C.-Y. (2013) Time-Dependent Vehicle Routing Problems with Time Windows and Split Delivery in City Logistics. *Proceedings of the Eastern Asia Society for Transportation Studies*, **9**.