Metaheuristics for Dynamic Vehicle Routing

Chapter · January 2013

DOI: 10.1007/978-3-642-30665-5_12

Chapter 1
Metaheuristics for Dynamic Vehicle Routing

Mostepha R. Khouadjia, Briseida Sarasola, Enrique Alba, El-Ghazali Talbi and Laetitia Jourdan

Abstract Combinatorial optimization problems are usually modeled in a static fashion. In this kind of problems, all data are known in advance, i.e. before the optimization process has started. However, in practice, many problems are dynamic, and change while the optimization is in progress. For example, in the Dynamic Vehicle Routing Problem (DVRP), which is one of the most challenging combinatorial optimization tasks, the aim consists in designing the optimal set of routes for a fleet of vehicles in order to serve a given set of customers. However, new customer orders arrive while the working day plan is in progress. In this case, routes must be reconfigured dynamically while executing the current simulation. The DVRP is an extension of the conventional routing problem, its main interest being the connection to many real word applications (repair services, courier mail services, dial-a-ride services, etc.). In this chapter, the DVRP is examined, and a survey on solving methods such as population-based metaheuristics and trajectory-based metaheuristics is exposed. Dynamic performances measures of different metaheuristics are assessed using dedicated indicators for the dynamic environment.

1.1 Introduction

Thanks to recent advances in information and communication technologies, vehicle fleets can now be managed in real-time. When jointly used, techniques like

Enrique Alba · Briseida Sarasola
Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga, E.T.S.I. Informática, Campus de Teatinos, 29071 Málaga, Spain.
e-mail: eat@lcc.uma.es, briseida@lcc.uma.es

Mostepha R. Khouadjia · El-Ghazali Talbi · Laetitia Jourdan
INRIA Lille Nord-Europe, Parc scientifique de la Haute-Borne, Bâtiment A, 40 Avenue Halley, Park Plaza, 59650 Villeneuve d'Ascq Cedex, France.
e-mail: mostepha-redouane.khouadjia@inria.fr, El-talbi@lifl.fr, laetitia.jourdan@inria.fr

geographic information systems (GIS), global positioning systems (GPS), traffic flow sensors, and cellular telephones are able to provide real-time data, such as current vehicle locations, new customer requests, and periodic estimates of road travel times. If suitably processed, this large amount of data can be used to reduce the cost and improve the service level of a modern company. To this end, revised routes have to be timely generated as soon as new events occur [28].

In this context, Dynamic Vehicle Routing Problems (DVRPs) are getting increasingly important [31, 32, 49, 55]. The VRP [17] is a well-known combinatorial problem which consists in designing routes for a fleet of capacitated vehicles to service a set of geographically dispersed points (customers, stores, schools, cities, warehouses, etc.) at the least cost (distance, time, or any other desired measure). It is possible to define several dynamic features which introduce dynamism in the classical VRP: roads between two customers could be blocked off, customers could modify their orders, the travel time for some routes could be increased due to bad weather conditions, etc. This implies that Dynamic VRPs constitute in fact a set of different problems, which are of crucial importance in today's industry, accounting for a significant portion of many distribution and transportation systems.

In this chapter, we first present an overview of different metaheuristics (from trajectory to population-based algorithms) for solving the DVRP. Second, we evaluate these algorithms according to dynamic performance measures.

The remainder of this chapter is organized as follows. Section 1.2 describes the dynamic VRP, its interests in practical applications and its specific characteristics. An overview on the problem representation as well as solving trajectory/population based metaheuristics is given in Section 1.3. In order to measure the dynamic performances of the metaheuristics, Section 1.4 presents certain measures that can be used to this end. The performance evaluation of different metaheuristics: Genetic Algorithm (GA), Ant Colony System (AS), Multi-Particle Swarm Optimization (MAPSO), and Tabu Search (TS) is analyzed in Section 1.5, and finally Section 1.6 presents conclusions and opens some lines for further research.

1.2 The Dynamic Vehicle Routing Problem

In this section, we present a formal description of the problem (Section 1.2.1) and a brief state of the art on the common objectives of the problem in the literature and its variants (sections 1.2.3 and 1.2.4).

1.2.1 Formal Description

The conventional VRP can be mathematically modeled by using an undirected graph $G = (V,E)$, where V is a vertex set, and E is an edge set. They are expressed as $V = \{v_0, v_1, ..., v_n\}$, and $E = \{(v_i, v_j) | v_i, v_j \in V, i < j\}$. D is a matrix of non-negative distances $d_{i,j}$ between customers v_i and v_j. Furthermore, a set of l homogeneous vehicles each with capacity Q originate from a single depot, represented by the vertex v_0, and must service all the customers, represented by the set $V' = V \setminus \{v_0\}$. The quantity of goods q_i requested by each customer i $(i > 0)$ is associated with the corresponding vertex. The goal is to find a feasible set of tours with the minimum total traveled distance. The VRP thus consists in determining a set of m vehicle routes of minimal total cost, $m \leq l$, starting and ending at a depot, such that every vertex in V' is visited exactly once by one vehicle. The total demand of all customers supplied by each vehicle cannot exceed the vehicle capacity Q. The capacity means the quantity of items (goods) that the vehicle can carry during its travel. Let be $S = \{R_1, \ldots, R_m\}$ a partition of V representing the routes of the vehicles to service all the customers. The cost of a given route $R_j = (r_0, r_1, \ldots, r_{k+1})$, where $r_i \in V$ and $r_0 = r_{k+1}$ (denoting the depot), is given by:

$$Cost(R_j) = \sum_{i=0}^{k} d_{r_i, r_{i+1}} \tag{1.1}$$

and the cost of the problem solution S is:

$$F_{VRP}(S) = \sum_{j=1}^{m} Cost(R_j) \tag{1.2}$$

with a constraint on the vehicle capacity:

$$\sum_{i=1}^{k} q_{r_i} \leq Q \tag{1.3}$$

where q_{r_i} is the associated quantity of the customer at r_i (items to be delivered/picked up).

We will consider a service time δ_i (time needed to unload/load all goods), required by a vehicle to load the quantity q_i at v_i. It is required that the total duration of any vehicle route (travel plus service times) may not surpass a given bound T, so, a route $R_j = (r_0, r_1, \ldots, r_{k+1})$ is feasible if the vehicle stops exactly once in each customer and the travel time of the route does not exceed a prespecified bound T corresponding to the end of the working day.

$$\sum_{i=0}^{k} d_{r_i, r_{i+1}} + \sum_{i=1}^{k} \delta_{r_i} \leq T \tag{1.4}$$

There may exist some restrictions such as the total traveling distance allowed for each vehicle, time windows to visit the specific customers, and so forth. The basic VRP deals with customers which are known in advance; all other information such as the driving time between the customers and the service times at the customers are also usually known prior to the planning.

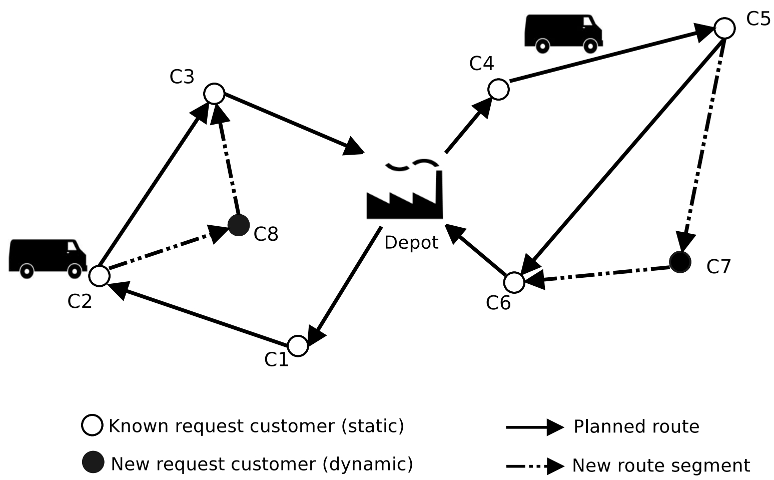

Fig. 1.1: A dynamic vehicle routing case.

The Dynamic Vehicle Routing Problem (DVRP) [55] is strongly related to the static VRP, as it can be described as a routing problem in which information about the problem can change during the optimization process. As conventional static VRPs are NP-hard, DVRP also belongs to the class of NP-hard problems. It is a discrete-time dynamic problem, and can be viewed as a sequence of P instances; each instance is a static problem, which starts at time t and must be solved within a specific time interval $\triangle_t$. We summarize that as follows:

$$P = \{(P_i, t_i, \triangle_t) \;/\; i = 0, 1, \ldots, i_{max}\} \tag{1.5}$$

With this information the duration of the instance i is $t_{i+1} - t_i$. The maximum number of instances i_{max} can be infinite if the problem is open-ended. A new instance P_{i+1} is generated by the action of the environment change ρ_i on the instance i. This is expressed by $P_{i+1} = \rho_i \oplus P_i$. This change in the environment can be due to several factors; for example, travel times can be time- [29] or traffic- [66] dependent, orders may be withdrawn or changed [62], some clients may be unknown when the execution begins [47], etc.

One standard approach to deal with this change is to consider the entire problem as a sequence of instances related to the events that happen in the environment. Each change corresponds to the arrival of new optimization problem that has to be solved. The time consecrated for solving each instance depends on the frequency of changes [12]. The aim is to design an optimization algorithm that is able of continuously adapting the solution to a changing environment. This technique is now commonly followed by the community that works on the DVRP domain [31, 39, 49]. Therefore, a partial static VRP has to be solved each time a new request is received. A simple example of a dynamic vehicle routing situation is shown in Figure 1.1. In the example, two uncapacitated vehicles must service both known and new customer requests. Designing a real-time routing algorithm depends to a large extent on how much the problem is dynamic. To quantify this concept, [46] and [45] have defined the degree of dynamism of a problem (dod). Without loss of generality, we assume that the planning horizon is a given interval $[0,T]$, possibly divided into a finite number of smaller intervals. Let n_s and n_d be the number of static and dynamic requests, respectively. Moreover, let $t_i \in [0,T]$ be the occurrence time of request i. Static requests are such that $t_i = 0$ while dynamic ones have $t_i \in]0,T]$. The degree of dynamism is defined as:

$$dod = \frac{n_d}{n_s + n_d} \tag{1.6}$$

which may vary between 0 and 1. If it is equal to 0, all requests are known in advance (static problem), while if it is equal to 1, all requests are dynamic (completely dynamic problem). Larsen [42] generalizes the definition proposed by Lund *et al.* [46] in order to take into account both dynamic request occurrence times and possible time windows. He observed that a system in which dynamic requests are received late over the planning horizon $[0,T]$ is more dynamic than another one in which the requests occur at the beginning of the working day. Thus, he introduces a new measure of dynamism:

$$edod = \frac{\sum_{i=1}^{n_s+n_d} (t_i/T)}{n_s + n_d}$$

The effective degree of dynamism then represents an average of how late the requests are received compared to the latest possible time the requests could be received. It can easily be seen $edod$ ranges between 0 and 1. It is equal to 0 if all user requests are known in advance while it is equal to 1 if all user requests occur at time T. Finally, Larsen extends the definition of $edod$ to take into account possible time windows on user service time. Let $[a_i, b_i]$ be the interval time of the client i referred as time windows, with a_i and b_i corresponding to the earliest and the latest possible time when the service should begin, respectively.

$$edod_{tw} = \frac{\sum_{i=1}^{n_s+n_d} [T - (b_i - t_i)]/T}{n_s + n_d}$$

It is also obvious that $edod_{tw}$ varies between 0 and 1. Moreover, if no time windows are imposed (i.e. $a_i = t_i$ and $b_i = T$), then $edod_{tw} = edod$.

Larsen *et al.* [43] describe and test several dynamic policies to minimize routing costs for the Partially Dynamic Traveling Repairman Problem (PDTRP) with various degrees of dynamism.

1.2.2 DVRP Interests

There are several important problems that must be solved in real-time. In [27, 28, 42], the authors list a number of real-life applications that motivate the research in the domain of dynamic vehicle routing problems.

- *Supply and distribution companies*: In seller-managed systems, distribution companies estimate customer inventory level in such a way to replenish them before stock depletion. Hence, demands are known beforehand in principle and all customers are static. However, since the actual demand quantity is uncertain, some customers might run out their stock and have to be serviced urgently.
- *Courier Services*: It refers to the international express mail services that must respond to customer requests in real-time. The load is collected at different customer locations and have to be delivered at another location. The package to be delivered is brought back to a remote terminal for further processing and shipping. The deliveries form a static routing problem since recipients are known by the driver. However, most pickup requests are dynamic because neither the driver nor the planner knows where the pickups are going to take place.
- *Rescue and repair service companies*: Repair services usually involve a utility firm (broken car rescue, electricity, gas, water and sewer, etc) that responds to customer requests for maintenance or repair of its facilities.
- *Dial-a-ride systems*: Dial-a-ride systems are mostly found in demand-responsive transportation systems aimed at servicing small communities or passengers with specific requirements (elderly, disabled). These problems are of the many-to-many when any node can serve as a source or destination for any commodity or service. Customers can book a trip one day in advance (static customers) or make a request at short notice (dynamic customers) [3, 21, 60].
- *Emergency services*: They cover the police, firefighting and ambulance services [30, 61]. By definition, the problem is pure dynamic since all customers are unknown beforehand and arrive in real-time. In most situations, routes are not formed because the requests are usually served before a new request appears. The problem then is to assign the best vehicle (for instance the nearest) to the new request. Solving methods are based on location analysis for deciding where to dispatch the emergency vehicles or to escape the downtown traffic jam.
- *Taxi cab services*: Managing taxi cabs is still another example of a real-life dynamic routing problem. In most taxi cab systems the percentage of dynamic customers is very high, i.e. only very few customers are known by the planner before the taxi cab leaves the central at the beginning of its working day [20].

1.2.3 Objectives

Depending on the nature of the system, the objective to be optimized is often a combination of different measures. DVRP inherits the classical objectives defined in the conventional VRP. Moreover, the dynamic nature of the problem leads to the emergence of new objectives. For instance, in weakly dynamic systems the focus is on minimizing routing cost [31, 49]. However, in strongly dynamic systems such as emergency services, the interest is to minimize the expected response time (i.e. the expected time lag between the moment when the user request occurs and its service time) [25, 43, 48]. Furthermore, there are other objectives such as maximizing the expected number of requests serviced during a given period of time [4, 5].

1.2.4 Related Works

In this section, we present a classification and an overview on the state of the art of dynamic vehicle routing problems. Different surveys have been proposed in scientific articles on DVRPs [8, 11, 28]. Psaraftis [54] defines a VRP to be dynamic when some input to the problem is revealed during the execution of the algorithm. Solutions to the problem should change as new information is revealed to the algorithm and to the decision maker. Possible information attributes may include the evolution of information (static / dynamic), the quality of information (known-deterministic / forecast / probabilistic / unknown), the availability of information (local / global), and the processing of information (centralized / decentralized).

We propose to classify the DVRPs according to the degree of knowledge that we have on the input data of the problem. A dynamic problem can be either deterministic or stochastic (see Figure 1.2). DVRP is deterministic if all data related to the customers are known when the customer demands arrive, otherwise it is stochastic. Both of these classes can be subject to different factors such as service time window, traffic jam, road maintenance, weather changes, breakdown of vehicles and so on. These factors often change the speed of vehicles and the travel time of arriving at the depot. Consequently, they lead to other sub-variants of the problem (see Table 1.1):

1. **Deterministic**: In the deterministic case, all the data related to the inputs are known. For instance, when a new customer demand appears, the customer location and the quantity of its demand are known. Different types of deterministic DVRP can be found in the literature as:

   a. **Dynamic Capacitated Vehicle Routing Problem with Dynamic Requests (DCVRP):** An important number of works exists on this variant [26, 40, 49] which represents the conventional definition of the problem, and where the existence of all customers and their localizations are deterministic, but their order can arrive at any time. The objective is to find a set of routes with the lowest traveled distance, observing the vehicle capacity limit.
   b. **Dynamic Vehicle Routing Problem with Time Window (DVRPTW):** It is one of the most well-studied variant of the DVRP [1, 18, 19, 32, 44, 48, 66]. Besides the possibility of requiring services in real time, the time window associated to each customer must be respected. The DVRPTW is closely related to the Dynamic Traveling Repairman Problem (DTRP) [6, 7], in which

m identical vehicles must service the upcoming demands. At each location, the vehicle serving the demand must spend some amount of time in on-site service. This service time is a random variable that is realized only when the service is completed. The objective is to find service policies that minimize the expected waiting time of the demands. Larsen *et al.* [43] proposed on-line policies for the Partially Dynamic Traveling Salesman Problem with Time Windows (PDTSPTW) that could be considered as an instance of DVRPTW with a single vehicle. The objective is to minimize the total or maximum lateness over the set of customers. A simple policy consists in requiring the vehicle to wait at the current customer location until it can service another customer without being early. Other policies, may suggest repositioning the vehicle at a location different from that of the current customer, based on prior information on future requests.

c. **Dynamic Vehicle Routing Problem with Time-Dependent Travel Times (DVRPTT):** Described in [29], it assumes that the travel times from the customer i to the customer j is variable over time. This variation could occur due to the type of the road, weather and traffic conditions that may strongly influence the speed of vehicles and hence travel times.
d. **Dynamic Pickup and Delivery Vehicle Routing Problem (DPDVRP):** It is based on the conventional Pickup and Delivery Vehicle Routing Problem (PDVRP) [59]. The problem consists of determining a set of optimal routes for a fleet of vehicles in order to serve customer requests. The objective is to minimize the total route length, i.e. the sum of the distances traveled by all the vehicles, under the following constraints: all requests must be served, each request must be served entirely by one vehicle (pairing constraint), and each pickup location has to be served before its corresponding delivery location (precedence constraint). The dynamic version arises when not all requests are known in advance [48].
Attanasio *et al.* present in [2] parallel implementations of a tabu search method developed previously by Cordeau and Laporte for the Dial-a-Ride Problem (DARP) [16]. Gendreau *et al.* [25] developed a tabu search heuristic where the neighborhood structure is based on ejection chains heuristic. Yang *et al.* [69] introduce a real-time multi-vehicle truckload pickup and delivery problem. They propose a mixed-integer programming formulation for the off-line version of the problem and propose a new rolling horizon re-optimization strategy for a dynamic version.

2. **Stochastic**: In stochastic dynamic problems (also known as probabilistic dynamic problems) uncertain data are related to customer demands and are represented by random variables.

   a. **Dynamic and Stochastic Capacitated Vehicle Routing Problem (DSCVRP)**: It considers customer requests are unknown and revealed over time. In addition, customer locations and service times are random variables and are realized dynamically during the plan execution. Bent and Van Hentenryck [4, 5] considered dynamic DVRPTW with stochastic customers.

They proposed a multiple scenario approach that continuously generates routing plans for scenarios including known and immediate requests to maximize the number of serviced customers. The approach was adapted from Solomon benchmarks with varying degree of dynamism. Hvattum *et al.* [33] addressed this variant of the problem. The authors consider that both customer locations and demands may be unknown in advance. They formulate the problem as a multi-stage stochastic programming problem, and a heuristic method was developed to generate routes by exploiting the information gathered on future customer demand.

b. **Dynamic and Stochastic Vehicle Routing Problem with Time Window (DSVRPTW)**: It has been introduced in [51]. In this problem, each service request is generated according to a stochastic process; once a service request appears, it remains active for a certain deterministic amount of time, and then expires. The objective is to minimize the number of possible vehicles and ensure that each demand is visited before its expiration.
c. **Dynamic Vehicle Routing Problem With Stochastic Travel Times (DVRP-STT)**: It assumes that the problem is subject to stochastic travel times. The travel times may change from one period to the next one. Some works present this version of the problem as in [52], where the travel time to the next destination is perturbed by adding a value generated with a normal probability law. This perturbation represents any unforeseen events that may occur along the current travel journey. It is known to the dispatching system only when the vehicle arrives at its planned destination.
d. **Dynamic and Stochastic Pickup and Delivery Vehicle Routing Problem (DSPDVRP)**: In this version of the problem, the stochastic process concerns the demand quantity that the vehicle must pick or delivery to each customer. Thus, we have uncertain quantities to pick up or deliver at the customers' location [68]. This distribution can be modelled by using a probabilistic law, such as a normal law for example, or by using fuzzy logic.

1.3 Solving Methods

In this part we present a common solution representation of the problem in the Section 1.3.1, and the major classes of metaheuristics proposed to solve this problem in Section 1.3.2 and Section 1.3.3.

1.3.1 Solution Representation

The solution representation in dynamic vehicle routing problems takes its source from representations that have already been proposed in the literature for the con-

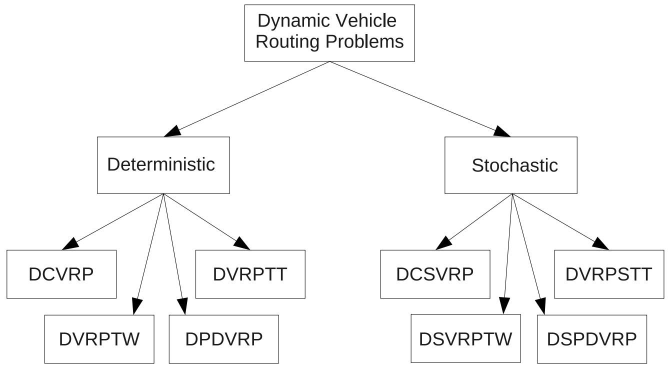

Fig. 1.2: Classification of DVRPs according to deterministic and stochastic information related to customer requests.

Table 1.1: Major publications on different variants of Dynamic Vehicle Routing problems.

DVRPs	Authors	Class	Characteristics	Objectives
Deterministic	Psaraftis *et al.* [54] Kilby *et al.* [40] Montemanni *et al.* [49] Hanshar *et al.* [31] Branching *et al.* [11] Khouadjia *et al.* [38] Sarasola *et al.* [58]	DCVRP	Dynamic requests Capacitated vehicles	Minimize the total traveled distance
	Oliveira *et al.* [18] Gendreau *et al.* [26] Mitrović-Minić *et al.* [48] Larsen *et al.* [44] Housroum *et al.* [32] Alvarez *et al.* [1]	DVRPTW	Dynamic requests Time windows	Minimize the total traveled distance and minimize the total lateness at the customer
	Haghani *et al.* [29] Kritzinger *et al.* [41]	DVRPTT	Variable travel times	Minimize the total traveled distance Minimize the total lateness/tardiness at the customer
	Gendreau *et al.* [25] Mitrović-Minić *et al.* [48]	DPDVRP	Dynamic requests Pickup and Delivery	Minimize the total travel time, tardiness over all pick-up and delivery locations, and sum of overtime over all vehicles
Stochastic	Bent *et al.* [4, 5] Hvattum *et al.* [33]	DSCVRP	Random customer locations Random service times	Maximize the number of serviced customers Minimize the total traveled distance
	Pavone *et al.* [51] Bertsimas *et al.* [6, 7]	DSVRPTW	Random customer locations Random service times Time window	Minimize the number of vehicles and the traveled distance Minimize the wait for completion of service

ventional static version of the problem [53, 56, 63]. Different representations for the DVRP solutions have been proposed in the literature either for trajectory-based metaheuristics or population-based metaheuristics [26, 31, 38, 49].

The difference between a dynamic representation and a static representation of the vehicle routing problem consists in the fact that given the dynamic nature of the problem, a representation with a variable length is required. It is justified by the fact that demands arrive over time and have to be inserted in the existing routes or by creating new ones. This representation is encoded as a set of vehicle routes. Each route includes some information on committed customers that have been visited by a given vehicle as well as those that are waiting for completion of service, or new customers that have been added to the day's service, but not yet assigned to any vehicle. Another important point is the information related to vehicles. The current vehicle position in the service area must be known by the dispatcher at any moment of the day's service. This allows to redirect the vehicle when new requests arrive into the system.

In [49], Montemanni *et al.* propose a representation for their Ant Colony System (ACS) algorithm. The authors consider v dummy depots (one for each vehicle of the fleet) and they refer to them as $d_1, \ldots, d_v$. Solutions retrieved by ants will be represented as long, single tours. In this context, nodes contained within two consecutive dummy depots d_a and d_b (with d_a, $d_b \in \{1, ..., v\}$) form the (partial) tour associated with vehicle a. The partial tour associated with vehicle b will start from the dummy depot d_b, which corresponds to the location of the last customer committed to vehicle b. The starting time from d_b corresponds to the end of the serving time for the last customer committed to vehicle b, while the capacity of b will be equal to the residual capacity of b, i.e. Q_b minus the quantity ordered by customers already committed to vehicle b.

Another representation is proposed by Hanshar *et al.* [31] for a Genetic Algorithm. Their chromosomal representation consists of two types of nodes: a node with a positive integer number representing a single customer (which has not yet been assigned to a vehicle) and a node depicted with a negative integer number representing a group of clustered customers that have already been committed to a given vehicle. Thus, the chromosome consists of integers, where new customers are directly represented on a chromosome with their corresponding positive index number and each committed customer is indirectly represented within one of the groups representing a given deployed vehicle. When the chromosome is decoded, new customers could be added to these pre-existing vehicles (i.e. groups) if they still have the capacity to accommodate new customer orders.

Garrido *et al.* [24] have tackled the DVRP using Evolutionary Hyper-Heuristics (EH-DVRP). The authors propose a chromosome representation for the low-level heuristics composed by two main data structures; a list of new unassigned customers represented by their identifier, and a set of routes which represents a set of partial solutions or states of the problem, formed by committed and uncommitted requests.

In [38], Khouadjia *et al.* propose a representation for a Particle Swarm Optimization (PSO) Algorithm. It consists in a discrete representation which expresses the route of m vehicles over the n customers to serve. The encoding allows the insertion of dynamic customers in the already planned routes. The representation of each route R_k is a permutation of n customers $R_k : (v_0, v_1, v_2, ..., v_i, ..., v_n, v_{n+1})$. This representation handles the static and dynamic aspects of the problem. On the one side, it allows the insertion of dynamic customers in the already planned routes. On the other side, if the customer is served, it cannot be shifted from its current route to another one. For the simulation process, the authors keep some information, such as the state of each customer (served / not served) and its time of service, the state of vehicles as their current position in the service region, their remaining capacity, the traveled distance, and their condition (committed / not committed).

1.3.2 Trajectory-Based Metaheuristics

Many works are related to trajectory-based metaheuristics for solving the DVRP (see Table 1.2). Gendreau *et al.* in [26] propose a parallel tabu search heuristic with an adaptive memory. The adaptive memory stores previously found elite solutions and uses them to generate new starting points for the tabu search. This is achieved by combining routes taken from different solutions in the memory. Any new solution produced by the tabu search is included in the memory if it is not filled yet. Otherwise, the new solution replaces the worst solution in memory, if it is better. The parallelization of the procedure was achieved at two different levels: (1) different tabu search threads run in parallel, each of them starting from a different initial solution; and (2) within each search thread, many tabu searches run independently on subproblems obtained through a decomposition procedure of the whole problem. For the parallel implementation a masterslave scheme was chosen to implement the procedure. The master process manages the adaptive memory and creates initial solutions for the slave processes that run the tabu search. Ichoua *et al.* in [34] reused the same algorithm with some enhancement related to the strategy for assigning customer requests to vehicles.

Mitrović -Minić *et al.* [48] dealt with the Dynamic Pickup and Delivery Problem with Time Windows (DPDVRPTW) and applied the cheapest insertion procedure in order to determine the overall best insertions for the locations of a request before its insertion. The improvement procedure is based on Tabu Search (TS). It is applied after the reinsertion procedure and it runs while new requests are being received.

Hanshar *et al.* have implemented a basic tabu search in [31]. Two operators are employed as neighborhood structure procedures: an inversion operator and a λ-exchange operator [50], each one applied according to some probability. Furthermore, Montemanni *et al.* [49] have implemented a GRASP (Greedy Randomized Adaptive Search Procedure) for dealing with the DVRP. Basically, initial tours are generated by iteratively selecting the next customers to visit. The procedure is repeated until a complete solution is built. Sarasola *et al.* [58] designed a flexible VNS for the VRP with dynamic requests. The flexibility strategy is based on the relaxation of the maximum tour length constraint.

Attanasio *et al.* present in [2] parallel implementations of a tabu search method developed previously by Cordeau and Laporte for the static Dial-a-Ride Problem (DARP) [16]. In this latter the requests are received throughout the day and the primary objective is to accept as many requests as possible as possible with the available fleet of vehicles. Furthermore, the routes are constructed under the constraint that users specify pick-up and drop-off requests between origins and destinations.

1.3.3 Population-Based Metaheuristics

Several population-based metaheuristics have been proposed in the literature (see Table 1.2). Next, we outline the major works that follow this research line.

1.3.3.1 Ant Colony Optimization

Ant System (AS) has been applied to tackle a large variety of Dynamic Vehicle Routing Problems [23, 37, 49, 57, 64].

Tian *et al.* [64] present a hybrid Ant System to handle the dynamism by means of modifying the pheromone matrix in order to take advantage of the old information gathered during the previous search. They propose a new pheromone initialization for new demands, which works better than a re-start optimization. Furthermore, they use a simple strategy that consists in grouping new requests at every fixed interval-time before their introduction into the system. In addition, they make further improvements on vehicle routes with the local search 2-Opt heuristic.

Jun *et al.* [37] addressed a hybrid multi-objective ant colony algorithm for solving DVRPTW. They consider two sub-objectives such as the vehicle number and the time cost. In their Ant Colony Algorithm, an Evolutionary Algorithm (EA) is embedded to increase the pheromone update. They explain that EA participates to speed up the convergence of their algorithm.

Montemanni *et al.* [49] exploit some features of the Ant Colony System optimization paradigm to smoothly save information about promising solutions when the optimization problem evolves because of the arrival of new orders. One of these characteristics is the pheromone conservation procedure, which contains information about good solutions features. In particular, pairs of customers that have been visited in sequence in good solutions, will have high values in the corresponding entries of the pheromone matrix. In the dynamic context, it is used to pass information about the properties of good solutions from previously obtained results in the new changing environment, since the two problems are potentially very similar. This operation avoids restarting the optimization from scratch. Based on the Montemanni's algorithm, Rizzoli *et al.* [57] discuss the applications of ACO on a number of real-world problems. They propose some results obtained by their algorithm on an on-line VRP for fuel distribution in the city of Lugano (Switzerland).

Oliveira at al. [18] propose an Ant Colony Algorithm for the DVRPTW with two different forms of attractiveness (time windows and distance) for building the vehicles routes. According to their experiments, the higher the degree of dynamism, the fewer customers will be served.

Chitty *et al.* [14] introduce a hybrid dynamic programming-ant colony optimization approach to solve bi-criterion Vehicle Routing Problems. The aim is to find routes that have both shortest overall travel time and smallest variance in travel time. The hybrid approach uses the principles of dynamic programming to first solve simple problems using ACO (routing from each adjacent node to the end node), and then builds on this preliminary solution to eventually provide solutions (i.e. Pareto fronts) for routing between each node in the network and the destination node. However, the hybrid technique updates the pheromone concentrations only along the first edge visited by each ant. This technique is shown to provide the overall solution faster than an established bi-criterion ACO technique that is concerned only with routing between the start and destination nodes, allowing re-routing vehicle to dynamic changes within the road network.

1.3.3.2 Evolutionary Algorithms

Hanshar [31] proposes a Genetic Algorithm (GA) that handles the optimization of the static VRP like-instances that correspond to the whole dynamic optimization problem. The GA is launched at each fixed duration and must run within an efficient amount of time. The fitness evaluation involves the vehicle routes obtained after the translation of the chromosome representation. It returns the total travel distance / cost of the routes. The Best-Cost Route Crossover (BCRC) is used as the crossover operator and the inversion operator is used as the mutation operator.

Housroum *et al.* [32] deal with the Dynamic Vehicle Routing Problem with Time Windows (DVRPTW). The authors propose an approach based on genetic algorithms. For their algorithm, they use the PMX crossover and different mutation operators such as Or-Opt, 1-Opt, or swap. They validate their approach on modified Solomon's benchmarks which have been proposed by Gendreau *et al.* [26]. Zhao *et al.* in [70] use a GA similar to Housroum's algorithm [32] for solving the Dynamic Vehicle Routing Problem with time-dependent Travel Times (DVRPTT).

Alvarenga *et al.* propose in [1] a hybrid GA with Column Generation Heuristic for the DVRPTW. The authors propose a specific crossover that, at the first step, makes a random choice of routes from each parent involved. After all feasible routes have been inserted in the offspring, remaining customers are inserted into existing routes, if possible (second step). New routes are created if some customers remain after this insertion step. Eight different operators are used as mutation operators.

Branke *et al.* [13] propose a GA with different waiting strategies of vehicles for the DCVRP. A two-point crossover is chosen and the mutation is done by adding to each value a normally distributed random value.

For their Evolutionary Hyper-Heuristics (EH-DVRP) [24], Garrido *et al.* propose a high-level algorithm which evolves and combines different types of low-level heuristics (constructive, perturbation, and noise heuristics) to solve the problem. Each individual of the population refers to a sequence of genes that corresponds to a constructive and improvement heuristics which gradually inserts customers and repairs the set of routes created so far. These dedicated heuristics are applied to construct and improve partial states of the problem. The hyper-heuristic uses four operators to find new individuals: one recombination and three mutation-like operators. The recombination operator performs a one point crossover to generate two new offsprings. For the mutation operators, the first one randomly selects and copies one of the heuristics to another position in the chromosome which allows to include new heuristics in different steps of the algorithm. The second operator selects and replaces a gene by one single heuristic. The authors' idea is to give an alternative heuristic which may perform better in cooperation with existing ones. The last operator deletes a gene from the chromosome and discards some heuristics which cannot be useful to improve candidate solutions.

Wang *et al.* [66] have proposed an EA for solving the DVRPTW. For the algorihtm's reproduction phase, the authors used two-points crossover operator and a mutation operator that consists in changing the assignment of unserved customers to another vehicle. In order to enhance their algorithm, the authors propose to hybridize their algorithm with a modified Dijkstra's algorithm for finding real-time shortest paths.

Jih *et al.* [36] address a hybrid genetic algorithm for solving single-vehicle pickup and delivery problem with time windows and capacity constraints (DPDVRPTW). Their approach enables dynamic programming to achieve real-time performance and genetic algorithms to approximate optimal solutions. The initial population is created by the dynamic programming instead of generating it randomly. The dynamic programming passes the unfinished routes to the genetic algorithm in order to produce final solutions. The authors compare the performance of four crossover operators. These operators are order crossover (OX), uniform order-based crossover (UOX), merge cross #1 (MX1) and merge cross #2 (MX2) [9]. In addition, they consider three mutation operators:(i) two genes are selected randomly, and their positions are interchanged (swap operator); (ii) two break points are selected randomly and the order of the sub-route specified by the genes is inverted (inverse operator); (iii) if the vehicle arrives at the i_{th} stop and violates the constraints, the order of the genes within the first i_{th} sub-route is disturbed (rearrangement operator).

Haghani and Jung [29], deal with the pick-up and delivery vehicle routing problem with soft time windows, where they consider multiple vehicles with different capacities, real-time service requests, and real-time variations in travel times between demand nodes. This algorithm includes a vehicle merging operator in addition to the generic genetic operators, namely the crossover and the mutation operators.

Bosman *et al.* [10] introduce a probabilistic model to describe the behavior of the load announcements. This allows the routing to be informed about customer positions where loads are expected to arrive shortly. This approach outperforms the EA that only considers currently available loads. Only mutation is considered. In the mutation of an individual, two vehicles are chosen randomly (could be the same), and two customers from their respective routes are chosen randomly, and are swapped. This operator allows visits to customers to be exchanged between vehicles or to be re-ordered in the route of a single vehicle directly.

Van Hemertand and La Poutré [65] present an evolutionary algorithm that is able to provide solutions in real-time for the DVRP. The authors analyze the benefit of anticipatory vehicle moves within regions that have a high potential of generating loads (fruitful regions). Only mutation is considered. Two vehicles, possibly the same one, are chosen uniform randomly. In both vehicles two nodes are selected uniform randomly. If only one vehicle is chosen these nodes are chosen to be distinct. Then, the nodes are swapped.

1.3.3.3 Particle Swarm Optimization

Khouadjia *et al.* [39] have proposed a Particle Swarm Optimization (PSO) for solving DCVRP. The authors suggest a discrete optimization of the problem and some adaptive mechanisms. Since PSO is intrinsically a memory-based approach, due to the memorization by each particle of its current and best position in the search space, they propose to reuse the best positions gathered in the past to face the changing environment. At each new sub-problem, the algorithm selects the positions with the best solution cost in the new search landscape. From these positions, the particles are re-positioned (re-initialized) for the new optimization process. The velocity vector of each particle corresponds to the likely routes in which a customer could belong. The updating of the position vector is the application of the velocity vector. It is summarized in shifting customers from their respective route to another one according to the velocity vector and with cheapest strategy insertion (i.e. by minimizing the cost of the insertion). The updating process is very similar to the *ejection chain* method that has been applied successfully to vehicle routing [56].

Khouadjia *et al.* [38] have enhanced their algorithm, particularly against the early well-known convergence of PSO algorithm. They propose in [38], a multi-swarm approach called MAPSO (Multi-Adaptive Particle Swarm Optimization) to investigate whether a multi-population metaheuristic might be beneficial in dynamic vehicle routing environments. The aim is to place different swarms on the search space to counterbalance the loss of diversity population and to provide better reactivity to the arrival of new customers.

1.4 Dynamic Performance Measures

The goal of optimization in dynamic environments is not only to find an optimum within a given number of generations, but rather a perpetual adjustment to changing environmental conditions. Besides the accuracy of an approximation at time t, the stability of the algorithm is also of interest as well as the recovery time to reach again a certain approximation quality. We report here some measures that could be used for evaluating the performance of an algorithm designed for the DVRP.

Weicker [67] proposes three features for describing the goodness of a dynamic adaptation process: accuracy, stability, and ε-reactivity.

The optimization *accuracy* at time t for a fitness function F and optimization algorithm A is defined as

$$accuracy^t_{F,A} = \frac{Min^t_F}{F(best^t_A)} \tag{1.7}$$

where $best^t_A$ is the best candidate solution in the population at time t and Min^t_F the best fitness value in the search space (best known solution). The optimization accuracy ranges between 0 and 1, where accuracy 1 is the best possible value.

Table 1.2: State of the art metaheuristics for DVRP and its variants.

Metaheuristics		Authors	Problem	Operators or Neighborhood
Trajectory-Based	Tabu Search	Hanshar *et al.* [31]	DCVRP	λ-interchange and inversion
		Gendreau1 *et al.* [26] Ichoua *et al.* [34, 35]	DVRPTW	CROSS exchange
		Mitrović-Minić *et al.* [48] Attanasio *et al.* [2]	DPDVRP	
	GRASP	Montemanni *et al.* [49]	DCVRP	Greedy insertion
	VNS	Sarasola *et al.* [58]	DCVRP	Swap, insertion, 2-Opt, 2-Opt*
Population-Based	Optimization Ant Colony	Montemanni *et al.* [49] Rizzoli *et al.* [57] Tian *et al.* [64] Chitty [14]	DCVRP	Greedy heuristic – 2-opt heuristic Dynamic programming
		Jun *et al.* [37] Oliveira *et al.* [18]	DVRPTW	Cooperation with EA Greedy heuristic
	Evolutionary Algorithms	Hanshar *et al.* [31]	DCVRP	BCRC crossover Mutation (inversion)
		Branke *et al.* [13]		Two-point crossover Mutation (replacement)
		Garrido2010 *et al.* [24]		One-point crossover 3 mutation operators (replacement,insertion,deletion)
		Van Hemertand and La Poutré [65]		mutation (CROSS exchange)
		Housroum *et al.* [32]	DVRPTW	PMX crossover 3 mutations (Or-Opt, 1-Opt, swap)
		Alvarenga *et al.* [1]		Specific crossover 8 mutations (insertion, exchange, ...)
		Wang *et al.* [66]		Two-points crossover Mutation (Insertion)
		Jih *et al.* [36]	DPDVRPTW	3 crossovers (OX, UOX MX1, MX2) 3 mutations (rearrangement, swap, 2-Opt)
		Bosman *et al.* [10]		CROSS exchange mutation
		Zhao *et al.* [70]	DVRPTT	PMX crossover 3 mutations (Or-Opt, 1-Opt, swap)
	Particle Swarm Optimization	Khouadjia *et al.* [38, 39]	DCVRP	2-Opt heuristic Cheapest insertion heuristic

As a second goal, stability is an important issue in optimization. In the context of dynamic optimization, an adaptive algorithm is called stable if changes in the environment do not affect the optimization accuracy severely. Even in the case of drastic changes an algorithm should be able to limit the respective fitness drop. The stability at time t is defined as

$$stability^{t}_{F,A} = max\{0, accuracy(t) - accuracy(t-1)\} \tag{1.8}$$

and ranges between 0 and 1. A value close to 0 implies a high stability.

Finally, another aspect to be considered is the ability of the algorithm to react quickly to changes. This is measured by the ε-reactivity, which ranges in $[1, maxgen]$ (a smaller value implies a higher reactivity):

$$\varepsilon - reactivity_i = \min\{i' - i | i < i' \leq maxgen, i \in \mathbb{N}, \frac{accuracy_{i'}}{accuracy_i} \geq (1-\varepsilon)\}$$

1.5 Performance Assessment

This section is devoted to the performance evaluation of different recent metaheuristics proposed in the literature [31, 38, 39, 49]. We justify this choice by the fact that these approaches follow the same experimental protocol, from the simulation framework to the set of benchmarks. Thus, it is easy to have an idea about the performances of these algorithms. Besides, all the classes of population-based metaheuristics described in Section 1.3 are represented.

Several benchmarks have been used. The most used ones are those of Kilby [40]. They were derived from publicly available VRP benchmark data from three separate VRP sources, namely Taillard [63] (13 instances), Christofides and Beasley [15] (7 instances) and Fisher *et al.* [22] (2 instances). These instances were organized and extended by Kilby *et al.* [40]. Kilby *et al.* organized the instances into two groups, pickup and delivery and gave each request an available time which signifies when the order was placed in the system and a duration, which represents the minimum amount of time a vehicle waits at a customer. In [31, 39, 49] authors use the *dod* described in the Section 1.2 in order to determine the percentage of dynamic requests over the entire working day. The degree of dynamism was fixed to 0.5; this means that a half of the customers is considered as static, while the other half is dynamic. The optimization begins to plan routes with the known static customers at the beginning of the working day.

We report in the Table 1.3 the best found solutions from the literature on metaheuristics; Adaptive Particle Swarm Optimization (APSO) [39], Multi-Adaptive Particle Swarm Optimization (MASPO) [38], Genetic Algorithm(GA) [31], Tabu Search (TS) [31], and Ant System(AS) [49] on Kilby's instances. These metaheuristics deal with pickup instances. In this case, the driver of the vehicle is not concerned with what is being transported, but only the quantity that has to be picked from the customer.

We highlight the best found solutions in dark shaded cells and the average results in light shaded cells. For each instance, 30 runs of the algorithms have been considered. We can see that the multi-swarm *MAPSO* is able to provide higher quality solutions than the other algorithms. It outperforms the other metaheuristics, and gives 18 new best solutions out of the 21 Kilby's instances. *MAPSO* algorithm provides also the shortest total traveled distance over all instances. The improvement brought by *MAPSO* ranges between $[2.23-5.76]$ compared to the others metaheuristics on the total traveled distance. As to the dynamic performance measures, we have computed the accuracy at the end of the working day. Table 1.4 shows the accuracy of the previous algorithms. It reports the best obtained distances and the bounds Min_F^T (best known solutions) found by an (ideal) off-line algorithm which had access to the entire instance, including dynamic requests, beforehand. These solutions can be found in the literature[1] over the 21 Kilby's instances .

These best known solutions consider all customers to be static, and then are not feasible solutions for the DVRP. They work as a bound for the algorithms. From

[1] http://neo.lcc.uma.es/radi-aeb/WebVRP/

Table 1.3: Numerical results obtained by the state-of-the-art metaheuristics on Kilby's instances

Instances	Metaheuristics									
	APSO [39]		MAPSO [38]		AS [49]		GA [31]		TS [31]	
	Best	Average	Best	Average	Best	Average	Best	Average	Best	Average
c50	575.89	647.75	571.34	610.67	631.30	681.86	570.89	593.42	603.57	627.90
c75	970.45	1046.25	931.59	965.53	1009.36	1042.39	981.57	1013.45	981.51	1013.82
c100	988.27	1087.96	953.79	973.01	973.26	1066.16	961.10	987.59	997.15	1047.60
c100b	924.32	970.66	866.42	882.39	944.23	1023.60	881.92	900.94	891.42	932.14
c120	1276.88	1450.82	1223.49	1295.79	1416.45	1525.15	1303.59	1390.58	1331.22	1468.12
c150	1371.08	1499.54	1300.43	1357.71	1345.73	1455.50	1348.88	1386.93	1318.22	1401.06
c199	1640.40	1751.63	1595.97	1646.37	1771.04	1844.82	1654.51	1758.51	1750.09	1783.43
f71	279.52	339.08	287.51	296.76	311.18	358.69	301.79	309.94	280.23	306.33
f134	15875	16477.4	15150.5	16193	15135.51	16083.56	15528.81	15986.84	15717.90	16582.04
tai75a	1816.07	1978.51	1794.38	1849.37	1843.08	1945.20	1782.91	1856.66	1778.52	1883.47
tai75b	1447.39	1489.24	1396.42	1426.67	1535.43	1704.06	1464.56	1527.77	1461.37	1587.72
tai75c	1481.35	1555.36	1483.1	1518.65	1574.98	1653.58	1440.54	1501.91	1406.27	1527.72
tai75d	1414.28	1481.05	1391.99	1413.83	1472.35	1529.00	1399.83	1422.27	1430.83	1453.56
tai100a	2249.84	2378.26	2178.86	2214.61	2375.92	2428.38	2232.71	2295.61	2208.85	2310.37
tai100b	2238.42	2426.58	2140.57	2218.58	2283.97	2347.90	2147.70	2215.93	2219.28	2330.52
tai100c	1532.56	1612.1	1490.40	1550.63	1562.30	1655.91	1541.28	1622.66	1515.10	1604.18
tai100d	1955.06	2092.31	1838.75	1928.69	2008.13	2060.72	1834.60	1912.43	1881.91	2026.76
tai150a	3400.33	3581.66	3273.24	3389.97	3644.78	3840.18	3328.85	3501.83	3488.02	3598.69
tai150b	3013.99	3391.08	2861.91	2956.84	3166.88	3327.47	2933.40	3115.39	3109.23	3215.32
tai150c	2714.34	2859.97	2512.01	2671.35	2811.48	3016.14	2612.68	2743.55	2666.28	2913.67
tai150d	3025.43	3143.16	2861.46	2989.24	3058.87	3203.75	2950.61	3045.16	2950.83	3111.43
Total	50190.87	53260.37	48104.13	50349.66	50876.23	53794.02	49202.73	51089.37	49987.8	52725.85

Table 1.4, we see that MAPSO has the best accuracy average at the end of the simulation. This accuracy is equal to 0.89 (being 1.0 the best value) which denotes that the algorithm is able to produce good solutions on the conventional dynamic benchmarks. We do not report the reactivity and stability because we would need all the minimum traveled distances (exact optimal cost) at each arrival of a new customer.

1.6 Conclusions & Future Works

The Dynamic Vehicle Routing Problem (DVRP) has been surveyed in this chapter. This problem is important both in research and industrial domains due to its many real world applications. The state of the art presented in this chapter covers the problem representation as well as the existing solving metaheuristics. In addition, a practical study of several metaheuristics in terms of the solution quality is reported.

Table 1.4: Accuracy of different metaheuristics on the Kilby's instances

Instance	Min_F^T	Metaheuristics									
		APSO [39]		MAPSO [38]		AS [49]		GA [31]		TS [31]	
		Best	Accu.	Best	Accu.	Best	Accu.	Best	Accu.	Best	Accu.
c50	521	575.89	0.90	571.34	0.91	631.3	0.83	570.89	0.91	603.57	0.86
c75	832	970.45	0.86	931.59	0.89	1009.36	0.82	981.57	0.85	981.51	0.85
c100	817	988.27	0.83	953.79	0.86	973.26	0.84	961.1	0.85	997.15	0.82
c100b	820	924.32	0.89	866.42	0.95	944.23	0.87	881.92	0.93	891.42	0.92
c120	1042.11	1276.88	0.82	1223.49	0.85	1416.45	0.74	1303.59	0.8	1331.22	0.78
c150	1028.42	1371.08	0.75	1300.43	0.79	1345.73	0.76	1348.88	0.76	1318.22	0.78
c199	1291.45	1640.4	0.79	1595.97	0.81	1771.04	0.73	1654.51	0.78	1750.09	0.74
f71	237	279.52	0.85	287.51	0.82	311.18	0.76	301.79	0.79	280.23	0.85
f134	11620	15875	0.73	15150.5	0.77	15135.51	0.77	15528.81	0.75	15717.9	0.74
tai75a	1618.36	1816.07	0.89	1794.38	0.90	1843.08	0.88	1782.91	0.91	1778.52	0.91
tai75b	1344.64	1447.39	0.93	1396.42	0.96	1535.43	0.88	1464.56	0.92	1461.37	0.92
tai75c	1291.01	1481.35	0.87	1483.1	0.87	1574.98	0.82	1440.54	0.90	1406.27	0.92
tai75d	1365.42	1414.28	0.97	1391.99	0.98	1472.35	0.93	1399.83	0.98	1430.83	0.95
tai100a	2041.33	2249.84	0.91	2178.86	0.94	2375.92	0.86	2232.71	0.91	2208.85	0.92
tai100b	1940.61	2238.42	0.87	2140.57	0.91	2283.97	0.85	2147.7	0.90	2219.28	0.87
tai100c	1406.2	1532.56	0.92	1490.4	0.94	1562.3	0.9	1541.28	0.91	1515.1	0.93
tai100d	1581.25	1955.06	0.81	1838.75	0.86	2008.13	0.79	1834.6	0.86	1881.91	0.84
tai150a	3055.23	3400.33	0.90	3273.24	0.93	3644.78	0.84	3328.85	0.92	3488.02	0.88
tai150b	2656.47	3013.99	0.88	2861.91	0.93	3166.88	0.84	2933.4	0.91	3109.23	0.85
tai150c	2341.84	2714.34	0.86	2512.01	0.93	2811.48	0.83	2612.68	0.90	2666.28	0.88
tai150d	2645.39	3025.43	0.87	2861.46	0.92	3058.87	0.86	2950.61	0.90	2950.83	0.90
Average	1976.03	2390.04	0.86	2290.67	0.89	2422.68	0.83	2342.99	0.87	2380.37	0.86

Besides, according to dynamic performance measures, the accuracy of different algorithms is calculated.

This study shows that the multi-population-based metaheuristics are able to find high quality solutions comparatively to the rest of metaheuristics. This is easy to understand since they offer a rich diversity in the exploration, which allows the algorithm to easily track the moving optima throughout the search space.

Different issues remain open. One of them is the landscape analysis of the DVRP and the severity of the changes that can occur. Landscape study techniques will allow the algorithm to locate better the current optimum and anticipate its movements, given that, unless the change in the problem is strong, the new problem can be similar to the old one. Concerning the severity, if the change is strong and frequent, trajectory-based metaheuristics usually fail to react and to track the optima. Enhancing the diversity within this class of metaheursitcs is an inescapable issue.

Another prospect is the flexibility and robustness of the solutions. The best solution in terms of fitness quality may not be the most flexible or robust one when it comes to updating it when the problem changes. The underlying idea is searching for robust solutions or the manner to obtain them. Robust solutions are those that promise high quality even if the environment changes. One way to ensure robustness is to introduce flexibility in these solutions. Through anticipating the changes in the environment, we will be able to provide solutions that not only have a high quality, but that allow the adaptation to high quality solutions after the environment has changed. To preserve flexibility we could construct initial solutions being aware about the potential arrival of new orders; in order to do so, we can imagine to adjust dynamically the length of the working day, making it smaller at the beginning of the optimization process and letting it increase until it reaches the value defined by the problem instance. In this way, we can expect to get solutions with a larger number of shorter routes at the beginning of the simulation time. If there are more routes available and they are not built to use the whole working day length, it will be easier to place new customers in a good position in vehicle routes.

On this way, future approaches will integrate new mechanisms that handle these issues and will be able to respond better and react faster to the changing environment. New approaches will provide solutions which are comparable to the solutions obtained in the static case.

Acknowledgment

Authors acknowledge funds from the Associated Teams Program MOMDI of the French National Institute for Research in Computer Science and Control INRIA (http://www.inria.fr), the Spanish Ministry of Sciences and Innovation European FEDER under contract TIN2008-06491-C04-01 (M* project http://mstar.lcc.uma.es), and CICE, Junta de Andalucía under contract P07-TIC-03044 (DIRICOM project http://diricom.lcc.uma.es). Briseida Sarasola is supported by grant AP2009-1680 from the Spanish government.

References

[1] G. B. Alvarenga, R. M. A. Silva, and G. R. Mateus. A hybrid approach for the dynamic vehicle routing problem with time windows. In *Proceedings of the Fifth International Conference on Hybrid Intelligent Systems*, pages 61–67. IEEE Computer Society Washington, DC, USA, 2005.

[2] A. Attanasio, J. F. Cordeau, G. Ghiani, and G. Laporte. Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. *Parallel Computing*, 30(3):377–387, 2004.

[3] A. Beaudry, G. Laporte, T. Melo, and S. Nickel. Dynamic transportation of patients in hospitals. *OR spectrum*, 32(1):77–107, 2010.

[4] R. Bent and P. Van Hentenryck. Dynamic vehicle routing with stochastic requests. In G. Gottlob and T. Walsh, editors, *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pages 1362–1363, San Francisco, CA, USA, 2003. Morgan Kaufmann Publishers Inc.

[5] R. Bent and P. Van Hentenryck. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52(6):977–987, 2004.

[6] D. J. Bertsimas and G. J. Van Ryzin. A stochastic and dynamic vehicle routing problem in the euclidean plane. *Operations Research*, 39(4):601–615, 1991.
[7] D. J. Bertsimas and G. J. Van Ryzin. Stochastic and dynamic vehicle routing with general demand and interarrival time distributions. *Advanced Applied Probability*, 25:947–978, 1993.
[8] L. Bianchi. Notes on dynamic vehicle routing -the state of the art-. Technical report, Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale, 2000.
[9] J. L. Blanton, Jr. and R. L. Wainwright. Multiple vehicle routing with time and capacity constraints using genetic algorithms. In S. Forrest, editor, *Proceedings of the 5th International Conference on Genetic Algorithms*, pages 452–459, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.
[10] P. Bosman and H. La Poutré. Computationally intelligent online dynamic vehicle routing by explicit load prediction in an evolutionary algorithm. In T. Runarsson, H. G. Beyer, E. Burke, J. Merelo-Guervós, L. Whitley, and X. Yao, editors, *Parallel Problem Solving from Nature - PPSN IX*, volume 4193 of *Lecture Notes in Computer Science*, pages 312–321. Springer, Berlin / Heidelberg, 2006.
[11] R. M. Branchini, V. A. Armentano, and A. Løkketangen. Adaptive granular local search heuristic for a dynamic vehicle routing problem. *Computers & Operations Research*, 36(11):2955–2968, 2009.
[12] J. Branke. *Evolutionary optimization in dynamic environments*. Kluwer Academic Publishers, 2002.
[13] J. Branke, M. Middendorf, G. Noeth, and M. Dessouky. Waiting strategies for dynamic vehicle routing. *Transportation Science*, 39(3):298–312, 2005.
[14] D. M. Chitty and M. L. Hernandez. A hybrid ant colony optimisation technique for dynamic vehicle routing. In K. Deb, R. Poli, W. Banzhaf, H. G. Beyer, E. K. Burke, P. J. Darwen, D. Dasgupta, D. Floreano, J. A. Foster, M. Harman, O. Holland, P. L. Lanzi, L. Spector, A. Tettamanzi, D. Thierens, and A. M. Tyrrell, editors, *Genetic and Evolutionary Computation – GECCO 2004*, volume 3102 of *Lecture Notes in Computer Science*, pages 48–59, Berlin / Heidelberg, 2004. Springer.
[15] N. Christofides and J. Beasley. The period routing problem. *Networks*, 14(2):237–256, 1984.
[16] J. F. Cordeau and G. Laporte. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological*, 37(6):579–594, 2003.
[17] G. B. Dantzig and J. H. Ramser. The truck dispatching problem. *Operations Research, Management Sciences*, 6(1):80–91, 1959.
[18] S. M. de Oliveira, S. R. de Souza, and M. A. L. Silva. A solution of dynamic vehicle routing problem with time window via ant colony system metaheuristic. In *Proceedings of the 2008 10th Brazilian Symposium on Neural Networks*, SBRN '08, pages 21–26, Washington, DC, USA, 2008. IEEE Computer Society.

[19] A. Fabri and P. Recht. On dynamic pickup and delivery vehicle routing with several time windows and waiting times. *Transportation Research Part B: Methodological*, 40(4):335–350, 2006.

[20] K. Fagerholt, BA Foss, and OJ Horgen. A decision support model for establishing an air taxi service: a case study. *Journal of the Operational Research Society*, 60(9):1173–1182, 2009.

[21] C. Fiegl and C. Pontow. Online scheduling of pick-up and delivery tasks in hospitals. *Journal of Biomedical Informatics*, 42(4):624–632, 2009.

[22] M. Fisher. Vehicle routing. In C. L. Monma M. O. Ball, T. L. Magnanti and G. L. Nemhauser, editors, *Network Routing*, volume 8 of *Handbooks in Operations Research and Management Science*, pages 1–33. Elsevier, 1995.

[23] L. M. Gambardella, A. E. Rizzoli, F. Oliverio, N. Casagrande, A. V. Donati, R. Montemanni, and E. Lucibello. Ant Colony Optimization for vehicle routing in advanced logistics systems. In *Proceedings of MAS 2003 - International Workshop on Modeling & Applied Simulation*, pages 3–9, 2003.

[24] P. Garrido and M. C. Riff. DVRP: a hard dynamic combinatorial optimisation problem tackled by an evolutionary hyper-heuristic. *Journal of Heuristics*, 16:795–834, 2010.

[25] M. Gendreau, F. Guertin, J. Y. Potvin, and R. Séguin. Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries. *Transportation Research Part C: Emerging Technologies*, 14(3):157–174, 2006.

[26] M. Gendreau, F. Guertin, J. Y. Potvin, and E. Taillard. Parallel tabu search for real-time vehicle routing and dispatching. *Transportation Science*, 33(4):381–390, 1999.

[27] M. Gendreau and J. Y. Potvin. Dynamic vehicle routing and dispatching. 1998.

[28] G. Ghiani, F. Guerriero, G. Laporte, and R. Musmanno. Real-time vehicle routing: Solution concepts, algorithms and parallel computing strategies. *European Journal of Operational Research*, 151:1–11, 2003.

[29] A. Haghani and S. Jung. A dynamic vehicle routing problem with time-dependent travel times. *Comput. Oper. Res.*, 32:2959–2986, 2005.

[30] A. Haghani and S. Yang. Real-time emergency response fleet deployment: Concepts, systems, simulation & case studies. *Dynamic Fleet Management*, pages 133–162, 2007.

[31] F. T. Hanshar and B. M. Ombuki-Berman. Dynamic vehicle routing using genetic algorithms. *Applied Intelligence*, 27:89–99, 2007.

[32] H. Housroum, T. Hsu, R. Dupas, and G. Goncalves. A hybrid GA approach for solving the dynamic vehicle routing problem with time windows. In *2nd International Conference on Information & Communication Technologies: Workshop ICT in Intelligent Transportation Systems, ICTTA'06*, volume 1, pages 787–792, 2006.

[33] L. M. Hvattum, A. Løkketangen, and G. Laporte. Solving a dynamic and stochastic vehicle routing problem with a sample scenario hedging heuristic. *Transportation Science*, 40:421–438, 2006.

[34] S. Ichoua, M. Gendreau, and J. Y. Potvin. Diversion issues in real-time vehicle dispatching. *Transportation Science*, 34:426–438, 2000.

[35] S. Ichoua, M. Gendreau, and J. Y. Potvin. Vehicle dispatching with time-dependent travel times. *European Journal of Operational Research*, 144:379–396, 2003.

[36] W. R. Jih and J. Y. J. Hsu. Dynamic vehicle routing using hybrid genetic algorithms. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 453–458, Detroit, Michigan, May 1999.

[37] Q. Jun, J. Wang, and B. Zheng. A hybrid multi-objective algorithm for dynamic vehicle routing problems. In M. Bubak, G. D. Albada, J. Dongarra, and P. M. Sloot, editors, *Proceedings of the 8th International Conference on Computational Science, Part III*, ICCS '08, pages 674–681, Berlin / Heidelberg, 2008. Springer-Verlag.

[38] M. R Khouadjia, E. Alba, L. Jourdan, and E. G. Talbi. Multi-swarm optimization for dynamic combinatorial problems: a case study on dynamic vehicle routing problem. In M. Dorigo, M. Birattari, G. A. Di Caro, R. Doursat, and A. P. Engelbrecht, editors, *Proceedings of the 7th International Conference on Swarm Intelligence*, (ANTS'10), pages 227–238, Berlin / Heidelberg, 2010. Springer-Verlag.

[39] M. R. Khouadjia, L. Jourdan, and E. G. Talbi. Adaptive particle swarm for solving the dynamic vehicle routing problem. In *IEEE/ACS International Conference on Computer Systems and Applications (AICCSA'2010)*, pages 1–8. IEEE Computer Society, 2010.

[40] P. Kilby, P. Prosser, and P. Shaw. Dynamic VRPs: A study of scenarios. Technical report, University of Strathclyde, U. K., 1998.

[41] S. Kritzinger, F. Tricoire, K. F. Doerner, and R. F. Hartl. A variable neighborhood search for the vehicle routing problem with time dependent travel times and soft time windows. In C. A. Coello Coello, editor, *Learning and Intelligent Optimization*, volume 6683 of *Lecture Notes in Computer Science*, pages 61–75. Springer, 2011.

[42] A. Larsen. *The Dynamic Vehicle Routing Problem*. PhD thesis, Technical University of Denmark, 2000.

[43] A. Larsen, O. B. G. Madsen, and M. M. Solomon. Partially dynamic vehicle routing-models and algorithms. *Journal of the Operational Research Society*, 53(6):637–646, 2002.

[44] A. Larsen, O. B. G. Madsen, and M. M. Solomon. The a priori dynamic traveling salesman problem with time windows. *Transportation Science*, 38(4):459–472, 2004.

[45] A. Larsen, O. B. G. Madsen, and M. M. Solomon. Recent developments in dynamic vehicle routing systems. In B. Golden, S. Raghavan, and E. Wasil, editors, *The Vehicle Routing Problem: Latest Advances and New Challenges*, volume 43 of *Operations Research/Computer Science Interfaces Series*, pages 199–218. Springer US, 2008.

[46] K. Lund, O. B. G. Madsen, and J. M. Rygaard. Vehicle routing problems with varying degrees of dynamism. Technical report, IMM, The Department of Mathematical Modelling, Technical University of Denmark, 1996.
[47] J. M. De Magalhães and J. Pinho De Sousa. Dynamic VRP in pharmaceutical distribution -a case study. *Central European Journal of Operations Research*, 14(2):177–192, 2006.
[48] S. Mitrović-Minić, R. Krishnamurti, and G. Laporte. Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows. *Transportation Research Part B: Methodological*, 38(8):669–685, 2004.
[49] R. Montemanni, L. M. Gambardella, A. E. Rizzoli, and A. V. Donati. A new algorithm for a dynamic vehicle routing problem based on ant colony system. *Journal of Combinatorial Optimization*, 10:327–343, 2005.
[50] I. H. Osman. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Annals of Operations Research*, 41(4):421–451, 1993.
[51] M. Pavone, N. Bisnik, E. Frazzoli, and V. Isler. A stochastic and dynamic vehicle routing problem with time windows and customer impatience. *Mobile Networks and Applications*, 14:350–364, 2009.
[52] J. Y. Potvin, Y. Xu, and I. Benyahia. Vehicle routing and scheduling with dynamic travel times. *Comput. Oper. Res.*, 33:1129–1137, April 2006.
[53] C. Prins. A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, 31(12):1985–2002, 2004.
[54] H. N. Psaraftis. Dynamic vehicle routing problems. *Vehicle Routing: Methods and Studies*, 16:223–248, 1988.
[55] H. N. Psaraftis. Dynamic vehicle routing: status and prospects. *Annals of Operations Research*, 61:143–164, 1995.
[56] C. Rego. Node-ejection chains for the vehicle routing problem: Sequential and parallel algorithms. *Parallel Computing*, 27(3):201–222, 2001.
[57] A. Rizzoli, R. Montemanni, E. Lucibello, and L. Gambardella. Ant colony optimization for real-world vehicle routing problems. *Swarm Intelligence*, 1:135–151, 2007.
[58] B. Sarasola, M. R. Khouadjia, E. Alba, L. Jourdan, and E. G. Talbi. Flexible variable neighborhood search in dynamic vehicle routing. In *8th European event on Evolutionary Algorithms in Stochastic and Dynamic Environments (EvoSTOC'11)*, 27-29 April 2011.
[59] M. W. P. Savelsbergh and M. Sol. The general pickup and delivery problem. *Transportation Science*, 29(1):17–29, 1995.
[60] M. Schilde, K. F. Doerner, and R. F. Hartl. Metaheuristics for the dynamic stochastic dial-a-ride problem with expected return transports. *Computers & OR*, 38(12):1719–1730, 2011.
[61] V. Schmid and K. F. Doerner. Ambulance location and relocation problems with time-dependent travel times. *European Journal of Operational Research*, 207(3):1293–1303, 2010.
[62] L. Sun, X. Hu, Z. Wang, and M. Huang. A knowledge-based model representation and on-line solution method for dynamic vehicle routing problem.

In Y. Shi, G. D. van Albada, J. Dongarra, and P. M. A. Sloot, editors, *ICCS '07: Proceedings of the 7th International Conference on Computational Science, Part IV*, Lecture Notes in Computer Science, pages 218–226, Berlin / Heidelberg, 2007. Springer.

[63] É. Taillard. Parallel iterative search methods for vehicle routing problems. *Networks*, 23(8):661–673, 1993.

[64] Y. Tian, J. Song, D. Yao, and J. Hu. Dynamic vehicle routing problem using hybrid ant system. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, volume 2, pages 970–974, 2003.

[65] J. Van Hemert and J. La Poutré. Dynamic routing problems with fruitful regions: Models and evolutionary computation. In *Parallel Problem Solving from Nature - PPSN VIII*, volume 3242 of *Lecture Notes in Computer Science*, pages 692–701. Springer, Berlin / Heidelberg, 2004.

[66] J. Q. Wang, X. N. Tong, and Z. M. Li. An improved evolutionary algorithm for dynamic vehicle routing problem with time windows. In *ICCS'07: Proceedings of the 7th International Conference on Computational Science, Part IV*, pages 1147–1154, Berlin / Heidelberg, 2007. Springer.

[67] K. Weicker. Performance measures for dynamic environments. In J. J. Merelo Guervós, P. Adamidis, H. G. Beyer, J. L. Fernández-Villacañas Martín, and H.-P. Schwefel, editors, *Parallel Problem Solving from Nature PPSN VII*, volume 2439 of *Lecture Notes in Computer Science*, pages 64–76. Springer, 2002.

[68] J. Xu, G. Goncalves, and T. Hsu. Genetic algorithm for the vehicle routing problem with time windows and fuzzy demand. In *2008 IEEE World Congress on Computational Intelligence, WCCI 2008*, pages 4125–4129, 2008.

[69] J. Yang, P. Jaillet, and H. Mahmassani. Real-time multivehicle truckload pickup and delivery problems. *Transportation Science*, 38:135–148, 2004.

[70] X. Zhao, G. Goncalves, and R. Dupas. A genetic approach to solving the vehicle routing problem with time-dependent travel times. In *16th Mediterranean Conference on Control and Automation*, pages 413–418, 2008.