Space-Time Self-organization for the Dynamic VRPTW

Besma Zeddini1  Mahdi Zargayouna2  Adnan Yassine1  Moncef Temani3

1 LE HAVRE University, LMH Laboratory, 25, rue Philippe Lebon, 76600 Le Havre, France.
2 INRETS Institute, GRETIA Laboratory, 2, Rue de la Butte Verte, 93166 Noisy le Grand, France.
3 ENSI School of Computer Science, LI3 Laboratory, 2010 La Manouba, Tunisia.
{besma.zeddini,adnan.yassine}@univ-lehavre.fr, zargayouna@inrets.fr, moncef.temani@fst.rnu.tn

Abstract—Vehicle Routing problems are highly complex problems for which different Artificial Intelligence techniques have been used. In this paper, we propose an agent-oriented self-organization model for the dynamic version of the problem with time windows. Our proposal is based on a space-time representation of the Action Zones of the agents, which is able to maintain a good distribution of the vehicles on the environment. This distribution answers the objective of the dynamic problem, since it allows the agents to take their decisions while anticipating future changes in the system’s parameters.

Index Terms—Self-organization, Multi-agent Systems, Applications, Planning, Scheduling

I. INTRODUCTION

The deliveries of goods to stores, the routing of school buses, the distribution of newspapers and mail etc. are instantiations of theoretical problems called the Vehicle Routing Problems (VRP). The VRP have been an intensive research area in the last decades because of their large applicability in real life problems. Several constrained variants were proposed in order to meet specific operational applications. Constraints concern vehicle capacity, time restrictions, requests configuration etc. One of the most widely studied problem is the time (and capacity) constrained version: the Vehicle Routing Problems with Time Windows (VRPTW henceforth). VRPTW and their variants (Pickup and Delivery Problem with TW, Dial A Ride Problem with TW ...) are hard combinatorial optimization problems met in many industrial applications. It can be formally stated as follows:

Let $G = (V, E)$ be a graph with node set $V = N \cup 0$ and edge set $E = \{(i,j) | i \in V, j \in V, i \neq j, N = 1, 2, ..., n \}$ is the customer set with node 0 is the depot. With each node $i \in V$ is associated a customer demand $d_i(q_0 = 0)$, a service time $s_i(s_0 = 0)$, and a hard service-time window $[e_i, l_i]$ i.e. a vehicle must be at $i$ before $l_i$ but can be at $i$ before $e_i$ and must wait until the service starts. For every edge $(i,j) \in A$, a distance $d_{ij} \geq 0$ and a travel time $t_{ij} \geq 0$ are given. Moreover, the fleet of vehicles is homogeneous and every vehicle is initially located and end its route at a central depot. Each customer demand is assumed to be less than the vehicle capacity $Cap$. The objective is to find an optimal set of routes (with the minimal cost) such that:

1. All routes start and end at the depot;
2. each customer in $N$ is visited exactly once within its time window;
3. the total of customer demands for each route cannot exceed the vehicle capacity $Cap$. The performance criteria are in general (following this order):
   1. The number of vehicles used,
   2. the total distance traveled,
   3. the total waiting time.

Since the problem is NP-hard, exact approaches are only of theoretical interest, and heuristics are performed in order to find good solutions, not necessarily optimal, within reasonable computational times. VRPTW can be divided into two sets: static problems and dynamic problems. In the static problems, all the problem data are available before the start of the execution. In order to meet the reactivity requirement of operational applications, the most promising category of problems is the dynamic version where some data could be not initially available, and especially the amount of available customers, before the start of execution. Indeed, operational vehicle routing problems are rarely fully static, and we can reasonably say that today a static system cannot meet the mobility needs of the users. In operational settings, and even if the whole number of customers to be served is known, there is still some elements that makes the problem dynamic. These elements include breakdowns, delays, noshows, etc. It is thus always useful to consider a problem that is not fully static.

A multi-agent modeling of the dynamic VRPTW is relevant for the following reasons. First, since it’s a hard problem, choosing a design allowing for processing distribution can be a solution to propose short answer times to customers requests. Second, with the technological developments, it is reasonable to consider vehicles with onboard calculation capacities. In this context, the problem is, actually, distributed and necessitates an adapted modeling to take profit of the onboard equipments of the vehicles. Finally, the consideration of a multi-agent point of view allows to envision new measures, new heuristics, not envisaged by centralized approaches.

In this paper, we propose a distributed version of an insertion heuristic with a special focus on the insertion cost of a customer in the route of a vehicle. Several multi-agent proposals in the literature have been proposed to distribute insertion heuristics, consisting in inserting the customers fol-
lowing their revealing order, and by choosing the vehicle that has to make the minimal detour to visit the new customer. But very few proposals in the literature propose new measures of the insertion cost of a customer in the route of a vehicle, instead of the detour. In the present work, we do propose such a new measure, based on a space time representation of Vehicle agents’ action zones. The objective is to allow the MAS to self-adapt exhibiting an equilibrated distribution of his Vehicle agents, and to decrease this way the number of vehicles mobilized to serve the customers.

The remainder of this paper is structured as follows. In section II, we briefly discuss previous proposals for the dynamic VRPTW w.r.t our approach. Section III presents the architecture of the MAS that we propose. In the section IV, we detail the space-time representation of the Action zones of the vehicles and its use as a measure for the insertion decision of the customers. We report our experimental results in section V before to conclude.

II. RELATED WORK

As we said in the introduction, exact approaches cannot meet operational settings, and interested readers in the optimization approaches can refer to [2] for a survey. In fact, most of the proposed solution methods are heuristic or metaheuristic methods. Artificial Intelligence metaheuristics have shown better performances than heuristics in average with benchmarking problems; for instance local search [8], genetic algorithms [5], simulated annealing [1], tabu search [11], ant colony [4] etc. (see [12] for various heuristics with artificial intelligence based techniques). Note that these approaches generally need several parameters, which values are closely dependent of the input data and are set after several system runs. That is why we are trying, in order to assess the impact of our approach, to minimize the necessary parameters to run the system.

These approaches perform well with static problems. However, the final solutions they provide are very constrained (tight spatiotemporal gap between customers in a route), and the insertion of new requests probably leads to new vehicles’ creation, or to incoming requests rejection. Generally speaking, two approaches can be envisioned to deal with dynamic requests. We can solve a static problem every time a change in the problem data occurs, which is obviously very expensive and is not realistic in operational settings. The other approach, which is in fact one of the most popular approaches in solving the dynamic versions is insertion methods. Insertion methods are greedy algorithms, meaning that they do not cancel a previous decision to insert a certain request in a specific route. Two versions are possible, sequential and parallel insertion. Parallel in this context means that several routes are created in parallel, in opposition with the sequential version which constructs only one route until no customers can be inserted. In [7], the authors show that parallel insertion procedures outperform sequential approaches.

The ADART [3] system is based on an a priori geographical segmentation of the network, by allowing each segment to some vehicles. For the dynamic management of customers arrival, the communication is established between the customer that has called the service and the onboard computer of the vehicle. Vehicles of the same regional zone negotiate the insertion of the customer, and the one with the minimal cost is chosen. In [13] and in [6], the authors propose a multi-agent architecture. The principle is the same: distribute an insertion heuristic, followed by a post-optimization step. In-Time [6] is a system composed of Customer agents and Vehicle agents. The Customer agent announces himself and all the Vehicle agents calculate his insertion cost in their routes. As usual, the Customer agent selects the cheapest offer.

From a protocol and an architecture point of view, our system sticks with the systems we have just described, since we propose a distributed version of insertion heuristics. But the traditional insertion cost of a customer in the route of a vehicle, based on the incurred detour of the vehicle, is the measure that is widely used. We propose a new insertion cost measure, focused on the space-time coverage of the vehicles that aims at counterbalancing the myopy of the traditional measures by privileging an insertion process that is future-centered.

III. MULTI-AGENT SYSTEM FOR THE DYNAMIC VRPTW

Our system is composed of a dynamic set of agents which interact to solve the dynamic VRPTW. A solution consists of a series of vehicles routes, each route consist of a sequence of customers with their associated visit time. We define three categories of agents. Customer agents, which represent users of the system (persons or goods), Vehicle agents, which represent vehicles in the MAS and Interface agents which represent an access point to the system (Web server, GUI, simulator, etc). When a user logs in the MAS, the data he provides are verified (existing node, valid time windows, etc.) and, if the data are correct, a Customer agent representing him and described by the data he provided is created.

In [16], we have designed, implemented and compared three possible architectures to model the dynamic VRPTW problem: a centralized architecture, a decentralized architecture and a hybrid architecture. We present them briefly in the following sections.

A. Centralized architecture

In this architecture, all the requests are treated by the same “agent”. He has all the required information about each vehicle and each customer: the occupancy rate of vehicles, their current positions and the traffic conditions in real time. Having all these information, he assigns to each customer the most appropriate vehicle for the service, i.e the one having the minimal overcost related to the customer insertion. Fig. 1 illustrates this architecture, in which, besides the three agents described above, we add a Planner agent which represent the decision-making center, he has in charge the routes computation and vehicles notification of his decision.

The scenario that we have proposed to study is the following: At a given moment, a user interacts with the Interface agent, which creates a Customer agent to represent him in the system. Once created, the Customer agent sends his request to the Planner agent which tries to insert him in each vehicle’s
If there is no Vehicle agent which can insert the customer, a new vehicle is created. Finally, the Planner agent sends the current route to each vehicle and informs the Customer agent of his vehicle and his visit times. The Vehicle agents don’t execute any operation, thus, they merely receive their current route and update their information. The centralized approach poses obvious problems. Indeed, sequential treatment of the customer requests slow down the system response time, which goes against the requirement of fast response to dynamic customers. Moreover, the failure of the Planner agent leads to a blackout at the global level. Nevertheless, the centralized architecture has the advantage of minimizing communications and updates at the agent level.

B. Decentralized architecture

The decentralized architecture is illustrated in Fig. 2. In this architecture, there is no bottleneck for the routes computation. Each Vehicle agent tries to insert the new customer in his route, proposes a cost for its insertion, and the vehicle with the minimal additional cost is selected. At each appearance of a new customer, the Customer agent broadcasts his request to all the vehicles in the system. Vehicle agents exchange their overcosts via messages. Each Vehicle agent compares his own cost with other agents’ costs, and stops bidding if the cost that is being offered to him is better than hers. Finally, the winner agent (the Vehicle agent with the minimal insertion cost) communicates with the Customer agent and both (the Vehicle agent and Customer agent) update their information. This architecture offers the advantage of a distributed processing and to be fault-tolerant. However, the communication costs explode with this architecture: the number of messages exchanged between Vehicle agents is of quadratic complexity.

C. Hybrid architecture

The hybrid architecture (cf. Fig. 3) is a compromise between the centralized and the decentralized approach. A new agent Dispatcher is inserted between the Customer and Vehicle agents and he has the role of dispatching the customer’s request, collecting bids from the Vehicle agents and choosing the one offering the minimal cost. The process describes a CNP (Contract Net Protocol) [9] where, in each occurrence of a Customer agent, the Dispatcher agent receives a set of proposals and selects those with a minimal cost.

Our proposal is based on this architecture. In the previous description, we use the additional cost related to the customer insertion to make a decision to include the new customer in the vehicle route. This cost is function of the detour made by the vehicle to integrate the new customer. We propose the variation of Vehicle agents’ action zones as a cost for customers’ insertion, as an alternative to the traditional measure.

IV. SELF-ORGANIZATION MODEL

The self-organization model that we propose has the objective of allowing the Vehicle agents to cover a maximal space-time zone of the transportation network. A space-time pair \(\langle n, t \rangle\) - with \(n\) a node and \(t\) a moment - is said to be “covered” by the Vehicle agent \(v\) if \(v\) can be in the node \(n\) at moment \(t\). In the context of the dynamic VRPTW, to maximise the space-time coverage of the Vehicle agents is to give them the maximum chances of satisfying the demand of a new customer in the future. This measure breaks with the logic of traditional measures which focus on the increase of the traveled distance, neglecting the impact of the current decision on future insertions.

A. Action Zone of a Vehicle Agent

Following the description provided above, the Dispatcher agent chooses between several Vehicle agents the one with the minimal proposed insertion cost. The systems that are based on this kind of heuristics - said insertion heuristics - utilise generally the measure used by Solomon [10] as an...
insertion cost. This measure consists in inserting the customer that result in a minimal increase of the general cost of the vehicle (function of the detour to be made by the vehicle). This measure is simple and is the most intuitive but unfortunately it suffers from an obvious drawback, because the insertion of the current customer might induce the insertion impossibility of a great number of future customers. Its problem is that it generates vehicle routes that are very constrained in time and space, i.e. routes that offer very few insertion possibilities between every pair of adjacent customers in the route of a vehicle. The appearance of new customers might mobilise new vehicles to serve them. With the modeling of Vehicle agents’ action zones, we propose a new measure of the insertion cost of a customer in the route of a vehicle, and therefore a new choice criterion between candidate vehicles for the same customer. We propose a measure which objective is to choose the Vehicle agent for whom “the decrease in the probability of participating to future insertions is minimal". We use the variation of the action zone of the Vehicle agent as an insertion cost of a customer in his route.

B. Intuition of the Action Zones

Consider a Vehicle agent $v$ that has an empty route. In order for this agent to be able to insert a new customer $c$ described by $n$ a node, $[c,l]$ a time window, $s$ a service time, and $q$ a quantity, $l$ has to be big enough to allow $v$ to be in $n$ without violating his time constraints. More precisely, the current time $t$, plus the travel time between the depot and $n$ has to be less or equal to $l$ (cf. Fig. 4).

![Image](Image 4 to 287x366)

Fig. 4. Feasible insertion

Starting from this observation, we define the Action Zone of a Vehicle agent as the number of potential customers that satisfy this constraint. To do so, we define “the physical environment" as a set of pairs $\langle\text{node}, \text{time}\rangle$, and the Action Zone of a Vehicle agent as the number of pairs that remain valid given his current route. When a Vehicle agent inserts a customer in his route, his Action Zone is recomputed, since some $\langle\text{node}, \text{time}\rangle$ pairs become not valid because of his insertion. The associated cost to an offer from a Vehicle agent $v$ for the insertion of a Customer agent $c$ corresponds to the hypothetical decrease of the action Zone of $v$ following the insertion of $c$ in his route.

The idea is that the chosen Vehicle for the insertion of a customer is the one that loses the minimal chance to be candidate for the insertion of future customers. Thus, the criterion that is maximized by the society of Vehicle agents is the sum of their Action Zones, i.e. the capacity that the MAS has to react to the appearance of Customer agents, without mobilizing new vehicles.

To illustrate the Action Zones and their dynamics, we present the version of the measure that is related to an Euclidean problem, i.e. where travel times are computed following the Euclidean metric. The following paragraphs detail the measure as well as its dynamics.

C. The Computation of Action Zones

In the Euclidean case, the transportation network is a plane, and the travel times between two points $i$ (described by $(x_i,y_i)$) and $j$ (described by $(x_j,y_j)$) is equal to

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

Therefore, if a vehicle is in $i$ at the moment $t_i$, he cannot be in $j$ earlier than $t_i + \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$.

We can compute at any time, from the current position of a vehicle, the set of triples $(x,y,t)$ where he can be in the future. Indeed, considering a plane with an X-axis in $[x_{\text{min}}, x_{\text{max}}]$ and a Y-axis in $[y_{\text{min}}, y_{\text{max}}]$, the set of space-time positions is the set of points in the cube delimited by $[x_{\text{min}}, x_{\text{max}}],[y_{\text{min}}, y_{\text{max}}]$ and $[t_0, l_0]$ ($t_0$ and $l_0$ are the minimal and maximal values for the time windows). Consider a vehicle in the depot $(x_0,y_0)$ at $t_0$. The set of points $(x,y,t)$ that are accessible by this vehicle are described by the following inequality:

$$\sqrt{(x - x_0)^2 + (y - y_0)^2} \leq (t - t_0)$$

The $(x,y,t)$ satisfying this inequality are those that are positioned inside the cone $C$ of vertex $(x_0,y_0,l_0)$ and with the equation $\sqrt{(x - x_0)^2 + (y - y_0)^2} = (t - t_0)$ (cf Fig. 5). This cone represents the Action Zone of a Vehicle agent in the Euclidean case. It represents all the possible space-time positions that this Vehicle agent is able to have in the future.

We use the Action Zone of the Vehicle agents when a Customer agent has to choose between several Vehicle agents for his insertion. We have to be able to compare the Action Zones of different Vehicle agents. To do so, we propose to quantify it, by computing the volume of the cone $C$ representing the future possible positions of the vehicle:

$$Volume(C) = \frac{1}{3} \times \pi \times (l_0 - e_0)^3$$

This is the quantification of the initial Action Zone of any new Vehicle agent joining the MAS. When a new Customer
agent appears, a Vehicle agent computes his new Action Zone, the cost that he proposes to the Dispatcher agent is the difference between his old Action Zone and his new one. The new Action zone computation is detailed in the following paragraph.

D. Dynamics of the Action Zones

Consider a customer \( c_2 \) (of coordinates \((x_2, y_2)\) and with a time window \([e_2, l_2]\)) that joins the system, and suppose that \( v \) is temporarily the only available Vehicle agent of the system and has an empty route. The agent \( v \) has to deduce his new space-time action zone, i.e. the space-time nodes that he can still reach without violating the time constraints of \( c_2 \). The new action zone answer the following questions: “if \( v \) had to be in \((x_2, y_2)\) at \( l_2 \), where would he have been before? And if he had to be there at \( e_2 \) where would he be after \( e_2 + s_2 \) ?”. The triples \((x, y, t)\) where the Vehicle agent can be before visiting \( c_2 \) are described by the inequality \([a]\), and the triples \((x, y, t)\) where he can be after visiting \( c_2 \) are describe by the inequality \([b]\).

\[
\sqrt{(x - x_2)^2 + (y - y_2)^2} \leq (l_2 - (t + s)) \quad [a]
\]

\[
\sqrt{(x - x_2)^2 + (y - y_2)^2} \leq (t - (e_2 + s_2)) \quad [b]
\]

The new Action Zone is illustrated by the Fig. 6: the new measure consists in the intersection of the initial cone \( C \) with the union of the two new cones described by the inequalities \([a]\) and \([b]\) (denoted respectively by \( C_1 \) and \( C_2 \)). The new measure of the Action Zone is equal to the volume of the intersection of \( C \) with the union of \( C_1 \) and \( C_2 \). The complete computation of the volume of the intersection of these two cones is reported in [15].

![Fig. 6. Space-Time Action Zone after the insertion of \( c_2 \)](image)

The cost of the insertion of a customer in the route of a vehicle is equal to the measure associated with the old Action Zone of the vehicle minus the measure of the new Action Zone, after the insertion of the customer. The quantity measured represents the space-time positions that the vehicle cannot have anymore, if he had to insert this customer in his route. The retained Vehicle agent to visit a given customer is the one for which the insertion of the customer causes less loss in his space-time Action Zone. This corresponds to choosing the vehicle that looses the minimal possibilities to be candidate for future customers.

The physical environment in the non-Euclidian case is not a space-time cube, but a space-time network. In [14], we propose a method for the self-organization of the MAS in the general case, where each Vehicle agent associates with each node \( n \) of the network an interval defining the moments where he can be in \( n \), which represents his Action Zone.

<table>
<thead>
<tr>
<th>Number of vehicles</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>6.4</td>
</tr>
<tr>
<td>50</td>
<td>10.7</td>
</tr>
<tr>
<td>100</td>
<td>19.1</td>
</tr>
</tbody>
</table>

TABLE I

<table>
<thead>
<tr>
<th>Number of vehicles</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>25</td>
<td>6.3</td>
</tr>
<tr>
<td>50</td>
<td>10.6</td>
</tr>
<tr>
<td>100</td>
<td>18.8</td>
</tr>
</tbody>
</table>

TABLE II

V. Results

Marius M. Solomon [10] has created a set of different problems for the dynamic VRPTW. In the Solomon benchmarks, six different sets of problems have been defined: C1, C2, R1, R2, RC1 and RC2. The customers are uniformly distributed in the problems of type R, clustered in the problems of type C, and a mix of the two is used in the problems of type RC.

We have used instances from both the classes R and C.

When the size of the fleet of vehicles is fixed in advance, the central concern with a dynamic VRPTW system is the amount of rejected requests, which should be limited. However, since we create vehicles dynamically when no vehicles can serve a new customer, our system does not reject any request; hence, our central concern becomes the size of this fleet. We have implemented a system which behavior is similar to ours. The only difference is the cost computed by a Vehicle agent, which is equal to the increase of the traveled distance, and not the loss of Action zone as we do. Table I reports the results with files of the class R1 where we consider successively 25, 50 and 100 customers, while Table II reports the results with files of the class C1.

The results show, with both classes of the problem, that the use of our measure mobilizes less vehicles than the traditional measure, and this is the case whatever the number of considered customers. Note that we had a different result - the traditional measure behaving better than our measure w.r.t the number of vehicles used - with only one file, the file 5 with 100 customers of the C1 class. This result validates the intuition of the measure which consists on the maximization of future insertion possibilities for a Vehicle agent. However, since our measure focuses exclusively on insertion feasibilities, the total distance traveled by all the vehicles with our measure is superior to the distance traveled with the traditional measure. We think that a compromise between the two measures, e.g. a weighted sum of the increase of the distance and the loss of Action Zones is able to give better results w.r.t the two criteria.
VI. CONCLUSION ET PERSPECTIVES

In this paper, we have proposed an agent-oriented self-organization model for the dynamic VRPTW based on the agents’ action zones. The action zones of the Vehicle agents reflect their space-time coverage of the environment. We use the variation of these action zones as a new metric between Vehicle agent to reduce the myopic behavior of traditional metrics. By optimizing the space-time coverage of the environment by the Vehicle agents, our model allows the MAS to self-adapt by exhibiting an equilibrated space-time distribution of the Vehicle agents, and to lessen this way the number of vehicles mobilized to serve the customers. Our current works are oriented towards taking into account historic data of customers requests on the network nodes. We use these data as a weighting of the action zones of the Vehicle agents that concern the nodes frequently requested, and this to make them converge towards high density zones.

REFERENCES


TABLE II

<table>
<thead>
<tr>
<th>Δ Distance</th>
<th>Number of vehicles</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>25</td>
<td>3.3</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>5.9</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>11.9</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>21.4</td>
</tr>
</tbody>
</table>

VI. CONCLUSION ET PERSPECTIVES

In this paper, we have proposed an agent-oriented self-organization model for the dynamic VRPTW based on the agents’ action zones. The action zones of the Vehicle agents reflect their space-time coverage of the environment. We use the variation of these action zones as a new metric between Vehicle agent to reduce the myopic behavior of traditional metrics. By optimizing the space-time coverage of the environment by the Vehicle agents, our model allows the MAS to self-adapt by exhibiting an equilibrated space-time distribution of the Vehicle agents, and to lessen this way the number of vehicles mobilized to serve the customers. Our current works are oriented towards taking into account historic data of customers requests on the network nodes. We use these data as a weighting of the action zones of the Vehicle agents that concern the nodes frequently requested, and this to make them converge towards high density zones.

REFERENCES

Besma Zeddini is Ph.D Student in le Havre University (France) and in the ENSI School of Computer Science (Tunisia). She is also a Computer Science Lecturer in Evry University (France). Her research interests are Multi-agent Systems and Vehicle Routing Problems.

Mahdi Zargayouna is Research Assistant in the French National Institute for Transport and Safety Research (INRETS) and currently Visiting Researcher in TU-Delft University (The Netherlands). His research interests are the formal specification of Multi-agent Systems and transportation problems such as Traveler Information and Dial A Ride Problems.

Adnan Yassine is Professor in Le Havre University and is Head of the LMAH Laboratory. His research interests are centered on the numeric analysis, convex and nonconvex analysis, convex and nonconvex optimization, global optimization, etc. with varied application domains such as image processing, finance and transportation problems.

Moncef Temani is Head of the Computer Science Institute (ISI) in Tunis (Tunisia) and member of the LI3 Laboratory. His research interests concern mainly Artificial Intelligence Techniques and Multi-agent Systems for routing and scheduling. He’s also interested in Web Services and Security.