On-line transportation Scheduling using Spatio-Temporal Reasoning

Maroua Bouzid

GREYC - Université de Caen

BD du Maréchal Juin

14032 Caen, France

`bouzid@info.unicaen.fr`

Abstract

In this paper we address the problem of the on-line transportation scheduling. Previous approaches use an off-line scheduling algorithm that accounts for new transportation tasks only when trucks are at a destination. To increase the performance of the system, we need to introduce the ability to account for the new transportation tasks immediately and to delegate it to a truck while this latter is moving toward a destination. For this purpose, the system needs to address two issues : (1) to determine the current location of the truck and its proximity to the departure and arrival of the new transportation task, and (2) to respect the temporal constraints. To this purpose, we use a space-time approach allowing us to deal with the first issue using a spatial reasoning, then address the second issue using a temporal reasoning. This approach is a first step towards a design of temporally situated multi-agent system that allows us to take location and the time of agents existing at this location to determine the suitable actions.

Topics: Temporal Representation and Reasoning in IA, Temporal aspects of agent-based system

1. Introduction

The growing interest in the development of methods for large and complex applications requires new extensions to existing methods in order to increase their efficiency and expressiveness. Indeed, these techniques need to be adapted to provide adequate solutions to cope with applications characterized by a high level of uncertainty and rapid change. These characteristics are common to several domains such as transportation scheduling and crisis situations. Algorithms used to efficiently solve static scheduling problems such as classical techniques, Operations research, and centralized approaches have failed to address open dynamic scheduling problems in the presence of uncertainty. Some investigators have offered approaches dedicated to dynamic scheduling problems in presence of uncertainty [6, 7].

Transportation scheduling application consists of distributed transportation companies that have to carry out transportation orders received asynchronously. Each company is provided with a set of trucks. Each company should optimize the use of its trucks when executing orders. Several approaches have been suggested to this application such as MARS [5], TRACONET [1] based on the task delegation method and using the Contract Net Protocol techniques (CNP).

Existing approaches assume that each local plan (for each truck) is known with precision thus ignoring the presence of uncertainties. However, the task of transportation is characterized by a high level of uncertainty regarding different factors such as the traffic density, the power of the truck used, etc... that cannot be ig-

1

nored. Consequently, the time at which a truck reaches destination is usually not known with precision, however, the uncertainty of the time interval during which the truck is due to arrive is well known. During this interval the availability of the truck is weighted by uncertainty. Consequently, during task transportation, there exist intervals during which truck's availability is not known precisely but it is assumed that it can be represented by a degree of uncertainty. Furthermore, the formal framework used to represent the availability of trucks, in the existing approach, is so weak that no powerful reasoning can be performed.

We have developed a suitable temporal representation of trucks based on *Temporal Characteristic functions* (TCF) [3, 4]. A characteristic function is a (possibly partial) function describing for how long a logical property holds (or does not hold) – thus it refers to the idea of characteristic function of a set. Roughly speaking, it can be also considered as a kind of two-valued 'trajectory' or 'history' characterizing the logical behavior given by certain atemporal formulae. Moreover, the possibility of representing the uncertainty on the property validity over time follows naturally from our functional approach and leads to the *Fuzzy Temporal Characteristic Function* (FTCF). FTCFs allow the representation of the uncertainty on the behavior of trucks due to traffic density and unpredictable events that prevent them from respecting their schedules. Thus, the FTCFs allow the representation of the uncertainty that characterizes the availability of trucks during task transportation.

In [4], we have addressed the problem of allocating orders to trucks using an off-line FTCF-based algorithm, where new orders are taken into consideration only for the parked trucks. In this paper, we extend our previous approach using a space-time reasoning in order to also consider the new orders that are for trucks in movement and to assess their ability to accept the new order. For that purpose, a representation of the spatial information of trucks will be introduced, in particular, their itinerary and their neighborhood. This spatial information allow us to develop a space-time reasoning

to, 1. determine the trucks in movement to be considered and 2. to select the most suitable to respond to this new order. In what follows, we mean by a system a temporally situated multi-agent system that is able to deal with time and space. This ability allows this situated multi-agent system a good interaction between agents and their environments by using spatial structures and when to interact by using the temporal dimension. This kind of multi-agent system is based on a perception-deliberation and action mechanism where the position in the environment defines the situations in which actions (considering a transportation order) can be considered. In the same way, this system considers also the time at which agents are located in a given site to take actions and the time when they could be located in a future site to make predictions on their future actions.

In the rest of the paper we introduce some preliminary definitions necessary for our approach, we then recall the off-line scheduling algorithm [4] and finally we describe our new approach to consider real-time issues when executing orders.

2. Preliminaries

2.1. Temporal characteristic functions

In this Section, we introduce the idea of a temporal approach to knowledge representation based on characteristic functions [3]. Basically, a characteristic function is a function specifying the time intervals during which a logical property holds (or does not hold) – thus it refers to the idea of characteristic functions of a set. A characteristic function for a given set Φ is any function ψ_Φ of the form: $\psi_\Phi : T \longrightarrow L$, where T is the global domain of interest and L denotes some set of values describing to 'what degree' an element of T belongs to Φ. In this paper, only the sets Φ which have the form of $\{x : \phi(x)\}$ are considered; where ϕ is a given property. Thus, a characteristic function is defined to be a function of the form $\psi_\phi : T \longrightarrow L$ describing to "what degree" an element of T satisfies the property (formula) ϕ, or, for convenience, to what degree the property expressed

2

by ϕ is satisfied by any of the elements of T. For further discussion, let us first establish the domain (maximal time interval) of interest. We shall consider events happening after a given time lapse 0 and before $+\infty$. This defines the domain for all characteristic functions. The formal definition of a characteristic function is as follows:

Definition 1 *Let $T = [0, +\infty[$ be the time domain of interest and let $L = \{0, 1\}$ be the set of distinguished values. Any mapping $\psi : T \longrightarrow L$ will be called a Temporal Characteristic Function (TCF). Any mapping $\psi : T' \longrightarrow L$ will be called a* **weak (partial) characteristic function***, if $T' \subseteq T$*

It is assumed that a TCF changes its value over T only a finite number of times. Thus any TCF is an interval-stable function taking values 0 or 1 over time. Any point of the domain where the function changes its value will be referred to as a change point, specific point or landmark. When considering weak TCFs one may be especially interested in *positive* weak functions and *negative* weak ones.

Definition 2 *A positive weak TCF is a weak TCF which takes as its value only 1 ($inf\{x | x = \psi(t) \quad for \quad t \in T\} = 1$). A negative weak TCF is a weak TCF which takes as its value only 0 ($sup\{x | x = \psi(t) \quad for \quad t \in T\} = 0$).*

Note that any TCF can be represented as a finite union of convex intervals denoting the largest intervals within which the function does not change its value[2]. Thus ψ can be given by $\{(\alpha_1, \beta_1), \ldots, (\alpha_k, \beta_k)\}$ provided that α-values denote the beginnings of intervals for which the function has value 1 or the end of intervals for which the function has as value 0 and β-values denote the ends of respective intervals for which the function has the value 1 or the beginnings of intervals for which the function is equal to 0. A similar representation can be applied to weak TCFs; however, the domain of the function must be given explicitly. The problem of what value (0 or 1) is taken at the α and β-values can be solved arbitrarily depending on current needs.

The idea of the knowledge representation language to be used consists in associating a propositional symbol p and a TCF ψ for expressing explicitly when p is true, false or unknown over the time domain T. Thus, the elementary objects of the language are pairs of the form $\langle p, \psi \rangle$. The basic intuition concerning the semantics of $\langle p, \psi \rangle$ is that some property p holds over time if the associated characteristic function ψ takes 1 as its value, and does not hold if it takes 0 (for weak episodes, p is undetermined for undefined ψ). TCFs constitute formal means to represent some properties over time; they seem to be more general and more powerful than intervals. Basically, a single function can represent 'behavior' of some property over the whole time domain T; thus, it represents the "history" of the a certain phenomenon. Moreover, as opposed to intervals, they can be easily extended to deal with multiple-valued or fuzzy logics.

2.2. Fuzzy Temporal Characteristic Functions (FTCF)

The idea of fuzzy characteristic functions consists in allowing the truth values to cover different degrees of truth from certainly false (0) to certainly true (1) passing through different scales of truth certainty. i.e. in terms of characteristic functions between 1 and 0.

Definition 3 *Let $T = [0, +\infty[$ be the time domain of interest and let $L = [0, 1]$ be the set (closed interval) of distinguished values. Any mapping $\psi : T \longrightarrow L$ will be called a* Fuzzy Temporal Characteristic Function. *Any mapping $\psi : T' \longrightarrow L$ is called a* weak *(or* partial*) FTCF, if $T' \subseteq T$.*

Thus any FTCF takes as its values some real numbers from the closed interval $[0, 1]$; for simplicity we assume that the function is "sufficiently regular". Note that any characteristic function satisfies the definition of FTCF (the opposite is not necessarily true). Note that an arbitrary FTCF cannot be represented using just a set of intervals (the α and β values): the discussed extension yields a concept significantly more general than a simple interval.

3

Furthermore, we a *filtering operation* for FTCF is an operation aimed at determining from a FTCF ψ defined over T and a condition C, a weak FTCF ψ^C defined on $T^{'} \subseteq T$ and such that $\psi^C(t)$ satisfies C. In a formal way:

Definition 4 *Let ψ be a FTCF defined over T and C a condition. filtering(ψ, T, C) is the weak fuzzy characteristic function ψ^C defined on $T^{'} \subseteq T$ and such that:*

$$\forall t \in T, \psi^C(t) =$$

$$\begin{cases} \psi(t), & \text{if } \psi(t) \text{ satisfies } C \\ \text{undefined}, & \text{otherwise} \end{cases}$$

Example 1 *Let ψ be a FTCF defined on T and $s \in [0,1]$:*

$$\text{filtering}(\psi, T, \psi(t) \le s) =$$

$$\begin{cases} \psi(t), & \text{if } \psi(t) \le s \\ \text{undefined}, & \text{otherwise} \end{cases}$$

For more details about this formalism, we invite the reader to consult [3].

3. Off-line transportation scheduling approach: a temporal reasoning algorithm

3.1 Stating the problem

The transportation scheduling problem consists of a set of shipping companies that receive orders of the form "*Load amount s of good g at location l_1 and transport it to location l_2 during a duration equal at most to d^w before a deadline D.* The system dedicated to solve this problem assigns each company's truck a specific time qualification. The time qualification is a FTCF ψ taking the value 1 when the truck is allocated and the value 0 when the truck is free. The FTCF allows the representation of the uncertainty on the availability of the truck during the intervals where the availability is not known in a precise way. The goal for each company is to optimize its service by executing the maximum number of orders and minimizing the allocation of truck-availability intervals.

3.2. A truck selection algorithm

Let us consider $\mathcal{O}$ the set of orders received by a given company where each order o_i is characterized by its deadlines D_i before which the order should be satisfied, a duration d_i^e representing an estimate of the time required to satisfy the order, and the worst-case duration d_i^w. These durations are determined from statistical data collected from previous execution of the truck. We use for the duration d_i^e the average duration over the collected data and the duration d_i^w the average duration increased with the standard deviation computed from the same collected data. Furthermore, we consider that each truck r has its FTCF ψ_r that indicates at time t to what degree the truck is allocated $\psi_r(t) = p$.

Given a set of trucks R and a set of orders $\mathcal{O}$, we need to generate a service schedule of the set $\mathcal{O}$. For this, the company, that we name in the following C_1, uses an algorithm based on the following steps:

- Compute expected utilities for all orders $o_i, Utility_{C_1}(o_i) = \text{Reward}(o_i) - \text{Cost}(d_i^w)$; where $cost$ is a function depending on the duration and charges to satisfy an order while $Reward$ is a function representing the rewarded value gained when the order is satisfied, it can for example represent the amount of money that the company is wanting to earn.

- Sort the set $\mathcal{O}$ according to the utility of orders;

- Satisfy the orders one by one as follows:

  - Search among the set of FTCFs of trucks those which are defined in the interval [Now, D_i] and of which values are less than a threshold s. This step is performed through a *filtering operation* to find these trucks. Let S_{truck} be the set of selected trucks.

  - For each truck $j \in S_{truck}$, let $I_{j,d_i^e}^{o_i}$ be the intervals such that duration($I_{j,d_i^e}^{o_i}$) > d_i^w (if $I = [x, y]$, duration(I) = $y - x$)

4

- Let $I^{o_i}_{k,min}$ be the interval with the least duration for all trucks $j \in S_{truck}$: $\forall j \in S_{truck}$ $I^{o_i}_{k,min} = arg(MIN(duration(I^{o_i}_{j,d^e_i})))$. The intuition behind the selection of the smallest truck-availability interval is 1. to maximize the utility of the truck and 2. to reduce the allocation of the truck to free it for other tasks.

- Send to the selected truck, the order o_i, its estimated and worst-case durations d^e_i and d^w_i and the interval $I^{o_i}_{k,min}$.

3.3. An order allocation algorithm

The truck selected to carry out the order should allocate the interval during which it satisfies the order. This operation consists in updating its FTCF by changing its value for every $t \in I$, where I is the interval during which it satisfies the order. The truck r receives the order o_i, its interval I_i=[a,b] during which it satisfies the order, the duration d^e required to satisfy the order and the duration d^w as the worst-case duration to satisfy the order. To update the FTCF, the agent has to assess several situations. To optimize its service, the truck tries to allocate the interval where the probability to be free is the highest. Consequently, the truck agent performs a filtering operation over its FTCF in I_i such that: $filtering(\psi_r, I_i, \psi_r(t) = 0)$. This operation enables us to select a set S^c of intervals $\{I^c_r\}$. There are two possible outcomes for the set S^c:

- $S^c = \emptyset$: this situation means that there is no interval during which the truck is sure to be free. Then, we select intervals during which the probability that the truck is free is the highest. Because a FTCF during these intervals is approximated with a linear function, the most important intervals are the ones with the smallest $|tangent|$. Since the *tangent* allows us to measure the overall uncertainty over an interval (other measurements can be used such as the integral over the interval, but the tangent is sufficient to give us the required information). Therefore, the interval during which

an FCTF has a minimal tangent is the interval where the likelihood the truck be free is the highest.

- $S^c \neq \emptyset$: this situation implies that there are intervals $[x,y]$ in the interval $[a,b]$ during which ψ_r takes the value 0. We select the interval $[x,y]$ with the highest duration, $y - x = MAX_{[x_i,y_i] \in S^c} (y_i - x_i)$.

For all cases, in order to compute the value of FTCF ψ, we assume that ψ is linear between two points (with known values of ψ) and we then generate the linear equation representing ψ between these points. In some cases, the allocation of the interval is based on construction of an interval having the same middle as the one computed for the truck. This strategy is motivated by the fact that we guarantee the allocation of the interval where the degree of availability is highest regardless of the fact that the allocation is not necessarily optimal.

The reader interested on the technical details on this section can see [4].

4 On-line transportation scheduling : a spatio-temporal reasoning algorithm

In the previous section, we consider situations where an agent can take a new order into consideration only when it doesn't move toward a destination. In other words, an agent (truck) must reach destination to consider a new transportation order. In this section, we relax this assumption by extending the approach presented in the previous section in order to allow an agent to consider new orders in real time. An agent, with this extension, is able to analyze its current capacity to accept a new order on-line while moving toward its destination. To do that, we develop a new approach using simple space-time reasoning. This reasoning requires, in addition to the characteristic function, spatial information :

- Truck at a destination : the system needs to know its location.

- Truck moves toward a destination to achieve an order : in such situations, many

5

spatial information are necessary for the reasoning.

- the departure and arrival locations consist of the interval $[l_1, l_2]$ of locations where the truck loads goods to where it should unload them.

- This interval is divided into a set of sub-intervals $[[l_1, l_1^1], [l_1^1, l_1^2], \ldots, [l_1^k, l_2]]$, where l_1^i is a location that will be visited by the truck while heading toward l_2. The locations are obtained from a geographical database. The sub-intervals allow the system to determine in real time and in a precise way the current location of the truck during its moving . The system needs only to know in which subinterval $[l_1^i, l_1^{i+1}]$ the truck is moving.

- The deadline before which the truck should reach l_2

- Finally, the system uses the time needed for loading and unloading goods at a location that we call θ_c and θ_d respectively.

4.1. Accounting for a new order

Let a new order be characterized by the transportation of goods from location l_1' to l_2' and the deadline of the delivery is D'. The allocation technique consists in determining the truck to which the order should be allocated. The system takes all trucks into consideration including the ones moving towards a destination and that are not so far from the departure location l_1'. To determine trucks in movement that should be considered, the system determines trucks moving in the vicinity of the location l_1'. This vicinity can be determined as a circle around the location l_1' with a ray K suitable to the distance $l_2' - l_1'$. This approach needs first to determine all trucks able to satisfy the order (all truck moving inside the vicinity) and then select the most suitable for the order.

4.2. Spatial reasoning to determine the appropriate set of trucks in movement

The spatial reasoning we have used consists in looking for the trucks moving toward the neighborhood $\mathcal{N}$ of the location l_1'. For that, consider truck i that is moving in its subinterval $[l_i^k, l_i^{k+1}]$. Many situations need to be studied :

- $l_i^{k+1} \in \mathcal{N}$: truck i is considered.

- $l_i^k \in \mathcal{N}$ and $l_i^{k+1} \notin \mathcal{N}$: truck i should not be considered

- $l_i^k \notin \mathcal{N}$ and $l_i^{k+1} \notin \mathcal{N}$: truck i should not be considered.

4.3. Spatio-temporal reasoning to allocate the order to moving trucks

In the previous section, we introduced a spatial reasoning that determines the trucks in movement that can be considered for the new order allocation. Let this set be $\mathcal{TM}$. We need to locate each truck $e \in \mathcal{TM}$ according to departure and arrival locations of the new order. Many cases need to be addressed. Before that, let us define what we mean by two spatial relations that we use in the following :

1. *overlap* relation: we use it as an extension of the overlap relation defined by Allen. Therefore, in our case, $[l_1, l_2]$ overlaps $[l_1', l_2']$ if the projection of $[l_1', l_2']$ on the axis defined by $[l_1, l_2]$ and $[l_1, l_2]$ overlap.

2. *include* relation: we use it as an extension of the include relation of Allen. Therefore, $[l_1', l_2']$ is included in $[l_1, l_2]$ if the projection of $[l_1', l_2']$ on the axis defined by $[l_1, l_2]$ is included in $[l_1, l_2]$

1. $\underline{[l_1, l_2] \text{ and } [l_1', l_2'] \text{ overlap such that } l_1 < l_1'}$ $\underline{\text{and } l_2 < l_2'}$:
in this situation, we locate exactly the sub-interval in which the truck moves, let it be $[l_1^k, l_1^{k+1}]$. The truck can drive towards l_1', get goods (of the new order) then determine what is the best policy. Two policies need to be analyzed.

6

- *policy 1* : driving towards the first destination (l_2) and then driving towards the new destination l'_2.
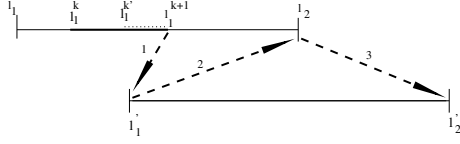

Figure 1. $[l_1, l_2]$ **and** $[l'_1, l'_2]$ **overlap**

- *policy 2* : driving towards the new destination l'_2 and then driving towards the first destination l_2 .
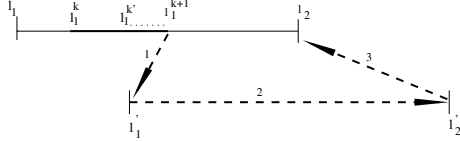

Figure 2. $[l_1, l_2]$ **and** $[l'_1, l'_2]$ **overlap**

In both policies, the time constraints (deadlines) should be respected such that:

- *policy 1* : The time constraints are :

$$\begin{cases} D \geq t_{current} + \tau_{l_1^{k+1}, l'_1} + \theta'_c + \tau_{l'_1, l_2} \\ D' \geq t_{current} + \tau_{l_1^{k+1}, l'_1} + \theta'_c + \\ \qquad \tau_{l'_1, l_2} + \tau_{l_2, l'_2} + \theta_d \end{cases}$$

- *policy 2* : the time constraints are then :

$$\begin{cases} D' \geq t_{current} + \tau_{l_1^{k+1}, l'_1} + \theta'_c + \tau_{l'_1, l'_2} \\ D \geq t_{current} + \tau_{l_1^{k+1}, l'_1} + \theta'_c + \\ \qquad \tau_{l'_1, l'_2} + \tau_{l'_2, l_2} + \theta'_d \end{cases}$$

Where $\tau_{l,l'}$ is a function measuring the time necessary to move from l to l', $t_{current}$ is the current time, θ_c time of loading goods and θ_d time of unloading goods. We assume that the time $\tau_{l_1^{k'}, l_1^{k+1}}$ we take from the current location $l_1^{k'}$ to l_1^{k+1} is negligible in comparison with $\tau_{l_1^{k+1}, l'_1}$. This assumption has no effect on our model.

2. $\underline{[l'_1, l'_2] \subset [l_1, l_2]}$:
   as in the previous case, we need to assess many policies 1. All of the time constraint are modeled as follows :

   - *policy 1* : drive towards l_2 and then towards l'_1. The time constraints for this policy are:
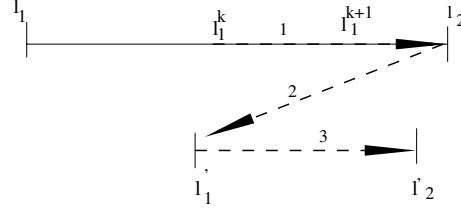

Figure 3. $[l'_1, l'_2] \subset [l_1, l_2]$

$$\begin{cases} D \geq t_{current} + \tau_{l_1^{k+1}, l_2} \\ D' \geq t_{current} + \tau_{l_1^{k+1}, l_2} + \theta_d + \\ \qquad \tau_{l_2, l'_1} + \tau_{l'_1, l'_2} + \theta'_c \end{cases}$$

- *policy 2* : drives towards l'_1, then towards l'_2 and finally towards l_2. The time constraints are:
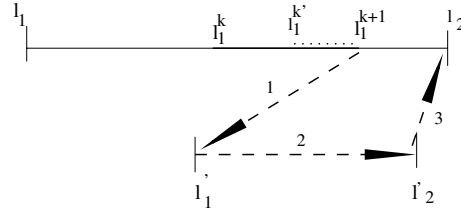

Figure 4. $[l'_1, l'_2] \subset [l_1, l_2]$

$$\begin{cases} D' \geq t_{current} + \tau_{l_1^{k+1}, l'_1} + \theta'_c + \tau_{l'_1, l'_2} \\ D \geq t_{current} + \tau_{l_1^{k+1}, l'_1} + \theta'_c + \\ \qquad \tau_{l'_1, l'_2} + \tau_{l'_2, l_2} + \theta'_d \end{cases}$$

- *policy 3* : drives towards l'_1, then towards l_2 and finally towards l'_2. The time constraints are:

$$\begin{cases} D \geq t_{current} + \tau_{l_1^{k+1}, l'_1} + \theta'_c + \tau_{l'_1, l_2} \\ D' \geq t_{current} + \tau_{l_1^{k+1}, l'_1} + \theta'_c + \\ \qquad \tau_{l'_1, l_2} + \tau_{l_2, l'_2} + \theta_d \end{cases}$$
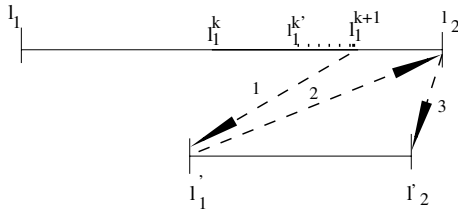
7

Figure 5. $[l'_1, l'_2] \subset [l_1, l_2]$

The policy selection is simply the policy with the shortest path strategy. We can introduce other criteria to select policies or use some multi-criteria decision approaches that we do not address in this paper.

4.4. A truck selection algorithm

In the neighborhood $\mathcal{N}$, many trucks can be selected for the new order. In this section, we discuss the selection criterion to select the optimal truck. In general, we can use the length of the path that a truck will follow and select the truck that will travel toward the shortest path. This criterion should be considered by it is not sufficient because the truck moving towards the shortest path could be different from the cheapest. To introduce this notion, we need to take into account the cost of loading the new goods given a current load of the truck. For that, we introduce an opportunity cost OC measure of loading new goods (NO) such that :

$$OC(NO) = Cost(current\_load) -$$

$$Cost(current\_load + LOAD(NO))$$

where LOAD gives the weight of new goods and $Cost$ function measures the cost of moving with a load at a given level. The strategy simply consists in selecting the truck that minimizes this opportunity cost. The selected truck is then determined by :

$$truck = \min_{t \in \mathcal{TM}} OC_t(NO)$$

When we have more than one truck, we can at this moment selects the truck with the shortest path.

5. Conclusion

In this paper, we have presented an extension of an off-line transportation scheduling algorithm to give trucks the ability to take new order online. This approach combines the temporal reasoning developed in the previous approach based on the FTCF, with a spatial reasoning that allows to determine trucks to be considered. We have developed a spatial reasoning with different policies and how trucks can determine the best policy using the shortest path criterion. Finally we discuss the problem when many trucks can respond to the new order. For this problem, we introduce a new measure based on the opportunity cost that allows a strategy to prefer a truck minimizing this cost over another. Many other extensions could be studied in future works such as relaxing the assumption that a truck can be derived from its initial destination. One question is how to extend this approach to derive the truck, once, twice, and so on ... as much as possible. A stack of secondary orders need to be handled and some new criteria of selection can be merged. Another issue that appears in this paper is the multi-criteria decision, especially for truck selection. Finally, this work is an interesting step towards the development of time-situated Multi-Agents Systems. The design of such systems will motivate our next research.

References

[1] M. Andersson and T. Sandholm. Time-quality tradeoffs in reallocative negotiation with combinatorial contract types. In *Proceedings of the National Conference on Artificial Intelligence AAAI'99*, pages 3–10, 1999.

[2] M. Bouzid and P. Ladkin. Simple reasoning with time-dependent propositions. *International Journal of Interest Group in Pure and Applied Logic (IGPL)*, 10(4):1–21, 2002.

[3] M. Bouzid and A. Ligeza. Temporal logic based on characteristic functions. In

8

C. Rolliger I. Wachsmuth and W. Brauer, editors, *Advances in Atrificial Intelligence, 19th Annual German Conference on Artificial Intelligence, volume 981 of Lecture Notes in Artificial Intelligence*, pages 221–232. Speinger Verlag, 1995.

[4] M. Bouzid and A.I Mouaddib. Uncertain temporal reasoning for the distributed transportation. In *Proceedings of TIME-98, International Workshop on Temporal Representaion and Reasoning*, pages 45–56, 1998.

[5] K. Fischer, J.P. Muller, and M. Pischal. Cooperative transportation scheduling: an application domain for dai. *Technical Report RR-95-01*, 1995.

[6] A.I. Mouaddib. A study of a dynamic progressive reasoning system. *International Journal of Theoretic and Experimental Artificial Intelligence*, 12:101–122, 2000.

[7] Thierry Vidal. Dealing with temporal uncertainty and reactivity in a space mission plan. In *Proceedings of 2nd NASA International Workshop on Planning and Scheduling for Space*, 2000.