

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

**4,800**

Open access books available

**122,000**

International authors and editors

**135M**

Downloads

Our authors are among the

**154**

Countries delivered to

**TOP 1%**

most cited scientists

**12.2%**

Contributors from top 500 universities



**WEB OF SCIENCE™**

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.

For more information visit [www.intechopen.com](http://www.intechopen.com)



# The Fusion of Fuzzy Temporal Plans: Managing Uncertainty and Time in Decentralized Command and Control Systems

Mohamad K. Allouche and Jean Berger

*Defence R&D Canada, Valcartier, 2459 Blvd. Pie XI North, Val Bélair, G3J 1X5  
Canada*

## 1. Introduction

Managing uncertainty in planning and plan execution activities is a key issue. This issue is even more critical in a network enabled environment where several tasks are distributed over the environment and can be carried out by different partners under different temporal constraints. In previous work (Allouche & Boukhtouta, 2009) a framework for distributed temporal plan fusion and monitoring has been proposed. A set of agents are tasked to coordinate the execution of different plans with different temporal constraints. Those plans are fused into one single plan, called coordinated plan. A coordinated plan can be executed and monitored by several agents while respecting the original temporal constraints of each agent's plan. The temporal constraints are set on tasks duration or/and between tasks. Each temporal constraint specifies the minimum and maximum authorized temporal distance between two events that typically represents the beginning or the end of execution of a task. In our opinion, the choice of such temporal constraints is not realistic in a distributed environment where different players must execute a common mission with limited and incomplete knowledge of their environment. The violation of a temporal constraint even with one unit of time will cause the plan execution to fail. For example, if a temporal constraint specifies that the duration of a task should last between 10 and 20 minutes, the fact that the task duration is 21 minutes is sufficient for the failure of any plan that contains this task. The fact that this situation is very likely to happen and that one minute late might be acceptable for a decision-maker, a new framework with degrading solutions for the problem is needed. In this work, we use fuzzy temporal constraints to maintain the execution of a plan with a degradation of its performance. In this context, the decision-maker decides whether the current execution is acceptable or not.

The following sections are organized as follows: Section 2 presents some related work. Section 3 presents a general framework for the fusion of fuzzy temporal plans. This framework is then applied to a Combat Search and Rescue (CSAR) mission in Section 4. Finally, Section 5 presents some conclusions and future work.

## 2. Related work

Classical requirements in supporting temporal reasoning for various application domains is the ability to handle and process quantitative information characterizing event duration,

and its intrinsic dependencies in handling multiple event or activity constraints. Temporal networks offered a suitable formalism to capture and handle constraint information and relationships in which variables denote event times, typically defined by start and end time points over a given timeline, and constraints reflect possible temporal relations between them. A well-known approach to properly represent constraint-based quantitative temporal networks lied within the general realm of Temporal Constraint Satisfaction Problems (TCSPs) (Dechter et al., 1991). TCSP is a technology to represent and support queries about events and its existing temporal relations. It mainly aims at determining constraint consistency, and answer scenario-based what-if constraint satisfaction queries. Early work mainly conducted in the 90's on TCSPs has been devoted on problem classification tractability, exact and polynomial-time approximation search algorithms (Schwalb, 1998).

Driven by problem requirements, our work primarily relates to *Simple Temporal Problems* (STPs) (Dechter et al., 1991) a subdomain of TCSPs. In the general setting, dependencies between temporal variables are captured in a constraint directed graph in which nodes represent an event time variables and arcs connecting nodes reflect binary constraints expressed as single time intervals exhibiting event duration. The formalism is used for basic temporal problem expressivity and support reasoning about temporal constraints. It provides an inference mechanism to verify properties, check consistency and handle queries. A solution to a STP problem prescribes values to event/activity time variables in order to satisfy all temporal constraints defined in the network. As STP focuses on non-disjunctive temporal constraint (single time interval), TCSP deals with general disjunctive temporal constraints (multiple intervals) (Schwalb & Dechter, 1997; Venable, 2005). Despite this apparent weakness, STP proves to be quite valuable in many practical application domains trading-off problem modeling complexity and tractability (polynomial time solution computation). Recent research on simple temporal problems has been increasingly directed to the development of approaches with augmented semantics and expressivity, and enhanced capability to efficiently handle uncertainty and preferences (Rossi, et al., 2006). Proposed frameworks and extensions include Simple Temporal network (crisp constraint, with no uncertainty), Simple Temporal Problems with Preferences, Simple Temporal Problems with Uncertainty which is closely related to our proposed approach and, Simple Temporal Problems with Preferences and Uncertainty. A recent survey may be found in (Rossi, et al., 2006).

Simple Temporal Problems (STPs) (Dechter et al., 1991) have traditionally been limited to hard crisp constraint network lacking expressivity and flexibility. Fuzzy temporal constraint networks were then introduced (Vila & Godo, 1994) proposing a propositional temporal language based on fuzzy temporal constraints to express knowledge and, imprecision as a single type of uncertainty. It provides an inference mechanism involving rules to reason on fuzzy temporal constraints, and ultimately specifying the tightest constraints possible on event duration. In (Godo & Vila, 1995), Godo and Vila proposed a STP-based *Fuzzy Temporal Constraint Networks* in which each constraint representing single time interval, is related to a possibility distribution. Temporal uncertainty is managed using possibility theory (Zadeh, 1975), mainly describing uncertainty with temporal information available in terms of vagueness or imprecision. It provides temporal information consistency-checking to identify potential contradictions and possible scenarios induced by constraints. Even if the framework strictly focuses on possibilities and differs from other kind of uncertainty such as probability, ignoring preferences or exploratory inference on the impact of time-point contingencies on possible/probable variable instantiations, it offers a simple and useful

framework to express basic knowledge with natural possibilistic semantics accounting for uncertainty induced by fuzzy temporal constraints. Fuzzy temporal constraints reasoning and time handling and emphasis on imprecision rather than ignorance to deal with uncertainty are naturally well-suited for our targeted application domain. This contrasts with subsequent frameworks described below, building on Simple Temporal Problems with Uncertainty and preferences, bringing unnecessary higher expressivity, query capability, semantic modeling or controllability checking for the problem at hand.

In the early 2000s, research efforts have been directed to further enrich existing frameworks in addressing limited expressiveness and flexibility to deal with preferences and uncertainty. Original TCSPs exclusively model hard temporal constraints emphasizing full constraint satisfaction and solution feasibility over partial constraint satisfaction and quality. Accordingly, Khatib et al. (Khatib et al., 2001; Rossi et al., 2002) proposed a generalized TCSP framework introducing a function mapping degree of temporal constraint satisfaction to preferences. Simple Temporal Problems with Preferences (STPPs) (Khatib et al., 2001), tackle the lack of expressiveness of hard temporal constraints by introducing preferences. This is due to the fact that in some real application domains, problem goals may be driven by biases for some solution classes, giving rise to preference-based constraint satisfaction optimality in which computed solution quality is determined in terms of specified preferences. In parallel, Badaloni and Giacomini (Badaloni & Giacomini, 2000) introduced the *Flexible Temporal Constraints* framework. The approach relies on soft constraints to represent preferences among feasible solutions, and, prioritized constraints to characterize constraint satisfaction suitability.

Other researchers refined temporal constraint networks to independently deal with uncertainty by taking into account the contingent nature of some constraints, whose effective duration is dictated by external world events under which the decision support system has no control. This departs from the TCSP framework which assumes that all activities have durations under the control of the agent. The notion of controllability (*strong*, *weak*, and *dynamic*) refers to the agent's (decision-maker) ability to control variables in assigning specific values (e.g. time point assignments) with respect to possible exogenous contingent events controlled by the external world (Vidal & Fargier, 1999). The Simple Temporal Problems with Uncertainty (STPUs) framework proposed by Vidal and Fargier, extends STPs incorporating *contingent* events controlled by "Nature", laying emphasis on controllability rather than traditional consistency. As there are no preferences stated explicitly, the focus is on controllability as opposed to optimality. As in STPs, activities durations in STPUs are modeled by intervals, whose start times (time-points) are determined by the agent. Recently, Venable (Venable, 2005) proposed Disjunctive temporal planning with uncertainty, an extension of the disjunctive temporal problem paradigm dealing with event contingency and similar controllability notions. The Probabilistic Simple Temporal Problems (PSTPs) framework has been introduced by Tsamardinos (Tsamardinos et al., 2003a) to handle temporal uncertainty. Similar ideas are presented in (Lau et al., 2005). In that setting, the occurrence of uncontrollable events is governed by a probability distribution rather than intervals. Alternatively, Dubois, HadjAli, and Prade (Dubois et al., 2003b) propose fuzziness in temporal reasoning to deal with uncertainty. They bring in *Fuzzy Allen Relations*, and as in Vila and Godo (Vila & Godo, 1994), consider on available information characterized by imprecision, and vagueness. Focusing on the notions of consistency and entailment, the authors ignore preferences and controllability. Similarly, Dubois, Fargier, and Prade (Dubois et al., 2003a) handle preferences and uncertainty using

the fuzzy framework for the classical job-shop scheduling problem characterized by ill-known activity durations. Using possibility theory they consider precedence constraints, capacity constraints and due dates, and release time constraints.

An alternate research direction consists in addressing Simple Temporal Problems with Preferences and Uncertainty (STPPU). Inspired from Simple Temporal Problems with Uncertainty (STPUs) (Vidal & Fargier, 1999), Rossi et al. (Rossi et al., 2006) introduced a new formalism handling both preferences and uncertainty in Simple Temporal Problems. They generalized controllability notions integrating optimality (preferences) and controllability allowing an agent to execute controllable events in a consistent way to meet preferences. The framework provides a way to handle preferences and compute the best solution (rather than a feasible one) through controllability property checking algorithms, in polynomial time. Conditional Temporal Problems (CTP) (Tsamardinos et al., 2003b) has been extended to include preferences (CTPP) (Falda et al., 2007). In CTP, a Boolean formula is attached to each temporal variable, representing preconditions enabling event occurrence. The uncertainty on temporal and conditional plans mainly lies on the selection of temporal variable to be executed. CTPP is a generalization of CTPs in which preferences on temporal constraints are explicitly introduced as well as fuzzy thresholds to govern the occurrence of some events.

### 3. General framework for fuzzy temporal plan fusion

In (Allouche & Boukhtouta, 2009) a temporal plan is defined as a graph of temporal constraints. It corresponds to a set of temporally constrained actions. Each action is defined by two events: *start* and *end*. Those events form the nodes of the graph. The temporal constraints between those nodes are defined by intervals specifying the minimum and maximum authorized delays between those events. In our new framework, temporal constraints are defined with fuzzy intervals. A temporal plan  $p_i$  is defined as the following:  $p_i \stackrel{def}{=} \{A_i, T_i\}$ , where  $A_i = \{e_1, \dots, e_n\}$  is a set of action start/end event nodes, and  $T_i$  defines fuzzy temporal constraints between nodes.

#### 3.1 Fuzzy temporal constraints

A fuzzy temporal constraint is set between actions start/end nodes. When a temporal constraint is set between the start and the end nodes of the same action, it specifies the minimum and maximum authorized duration for this action. A fuzzy temporal constraint can also synchronize different actions when it is set between their start/end nodes. A temporal constraint is represented by an interval of integers and a function  $\pi$  as shown in Figure 1. It is defined by the function  $T_i: A_i \times A_i \rightarrow (I, \pi)$ , where  $I$  is the set of all integer intervals and  $\pi: I \rightarrow [0, 1]$  is a *possibility distribution* over temporal distances.  $\pi$  associates a degree of possibility for each value in the interval, that is, it defines the degree of possibility to have a certain temporal distance between two nodes. Our previous definition of a temporal constraint becomes a particular instance of a fuzzy temporal constraint where all values (temporal distance) within the interval have the same degree of possibility equal to 1. A fuzzy temporal constraint can also be expressed by disjunction of several intervals (expressing alternative authorized durations), but this representation goes beyond the scope of this work.



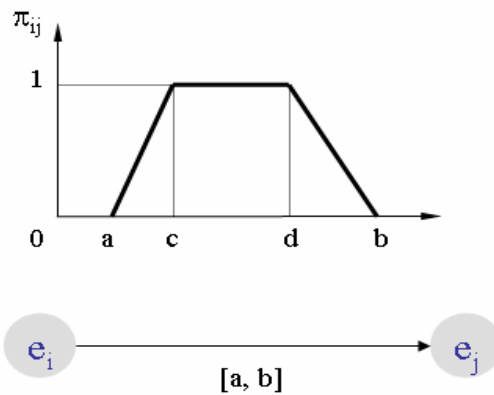


Fig. 1. Fuzzy temporal constraint set between two nodes  $e_i$  and  $e_j$

In Figure 1, the fuzzy temporal constraint specifies that  $e_j$  must occur at least “ $a$ ” units of time and at most “ $b$ ” units of time after  $e_i$ . This is represented by an arrow from  $e_i$  to  $e_j$ . The function  $\pi_{ij}$  represents the degree of possibility for each element of the interval.  $a$  and  $b$  have the minimum degree of possibility. This degree increases as we move toward the center of the interval. In our previous framework the user would have to define the temporal constraint by the interval  $[c d]$  as shown in Figure 1. In the fuzzy definition of this temporal constraint all values between  $a$  and  $c$ , and between  $d$  and  $b$  are still acceptable but with decreasing degree of possibility as we move farther from  $c$  and from  $d$ . The same fuzzy constraint can be expressed differently by saying that  $e_i$  must occur at least “ $a$ ” units of time and at most “ $b$ ” units of time before  $e_j$ . Schematically, this is represented by an arrow from  $e_j$  to  $e_i$ , labelled by the interval  $[-b, -a]$ .  $\pi_{ji}$  is the symmetric function of  $\pi_{ij}$ , that is,  $\pi_{ij}(x) = \pi_{ji}(-x)$  for all  $x$  in  $[a b]$ . The two constraints are equivalent, they coexist, and one is called the inverse of the other. It is possible to use  $+\infty$  and  $-\infty$  in order to define simple temporal precedence. For example, the interval  $[0, +\infty[$  may be used to specify that  $e_j$  must occur at the same time or after  $e_i$ . In this case the function  $\pi_{ij}$  should be defined over this interval. The inverse of this constraint is defined by the interval  $]-\infty, 0]$  and the symmetric function  $\pi_{ji}$ . The constraint defined by  $]-\infty, +\infty[$  is used to express temporal independence between  $e_i$  and  $e_j$ . It is called the *universal temporal constraint* and its function  $\pi_{ij} = 1$ . The null duration constraint is defined by the interval  $[0 0]$ . Its function  $\pi_{ij}$  is defined as follows:  $\pi_{ij} = \begin{cases} 1 & \text{if } d = 0 \\ 0 & \text{otherwise} \end{cases}$ . For

example, the null duration constraint will be set on atomic actions that have no duration and are represented by a single node, or on any node  $e_i$  to express that an event has no duration ( $\pi_{ii}$  is a null duration function).

The definition of fuzzy temporal actions allows the expression of all Allen’s temporal relations (Allen, 1983) where each pair of action nodes is associated to an interval and a possibility function  $\pi$ . These expressions<sup>1</sup> are given in Fig. 2. In this figure,  $I$ ,  $I^+$ ,  $J$  and  $J^+$  are action nodes. For the sake of clarity we didn’t include the  $\pi_{ij}$  functions associated to the fuzzy temporal constraints. While Allen’s relations are qualitative, the proposed representation allows qualitative as well as quantitative temporal relations by simply quantifying the interval of a fuzzy temporal constraint.

<sup>1</sup> In Fig. 2 only seven relations are represented. The converse of these relations (preceded-by, met-by, started-by, finished-by, overlapped-by and contains) are not represented.

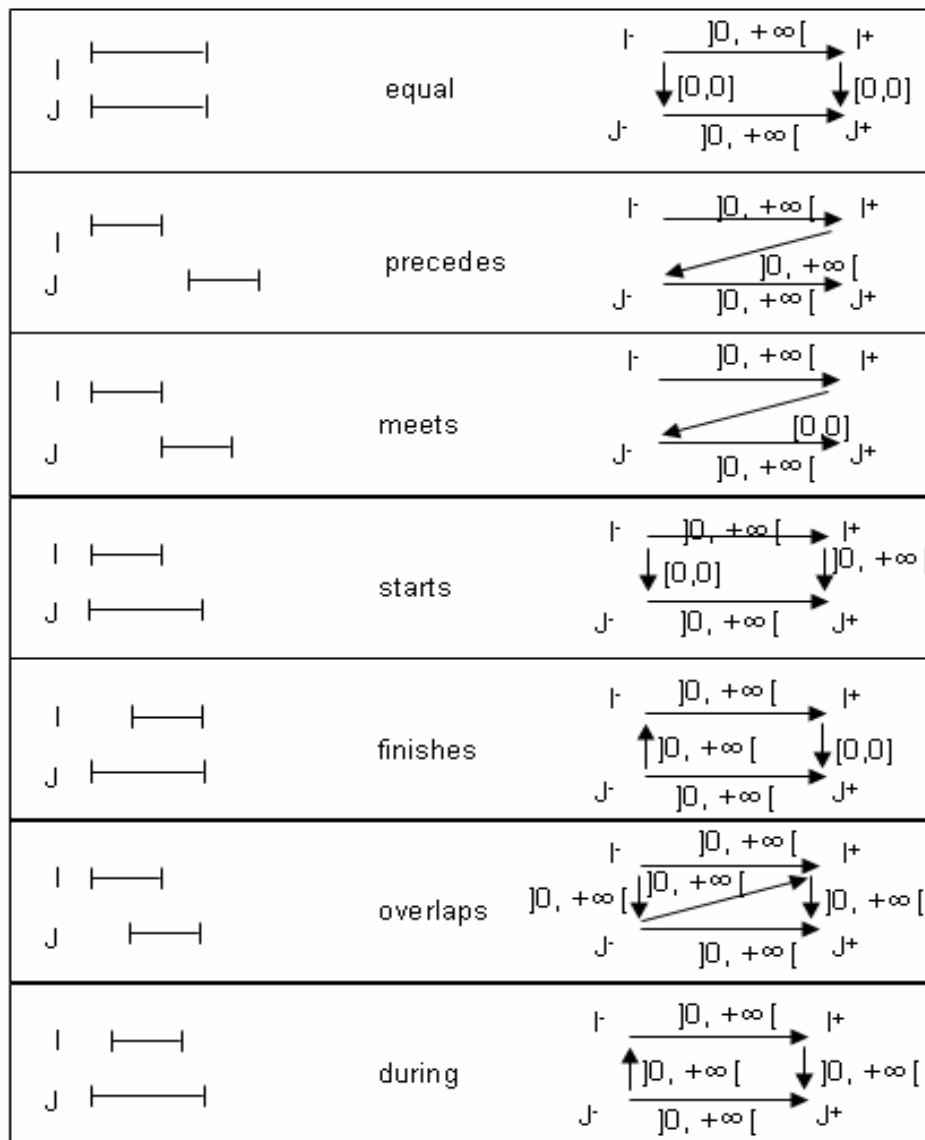


Fig. 2. Temporal constraints expressing the 13 temporal relations defined by Allen

### 3.2 Operations on fuzzy temporal constraints

As in the previous work, setting fuzzy temporal constraints between a set of events may lead to a temporally incoherent graph. An operation of graph minimization is necessary to check the coherence of the graph. This operation will be described in details in Section 3.3.1. To this end, two operations between fuzzy temporal constraints are needed.

#### 3.2.1 Intersection

The intersection of two fuzzy temporal constraints  $\{I_1, \pi_1\}$  and  $\{I_2, \pi_2\}$  is also a fuzzy temporal constraint  $\{I_1, \pi_1\} \cap \{I_2, \pi_2\} \stackrel{\text{def.}}{=} \{I_1 \cap I_2, \pi_1 \cap \pi_2\}$ , where  $I_1 \cap I_2$  is the intersection between

the two intervals  $I_1$  and  $I_2$ ,  $\pi_i$  is the restriction of  $\pi_i$  on the interval  $I_1 \cap I_2$ .

The intersection operation is illustrated in Figure 3.

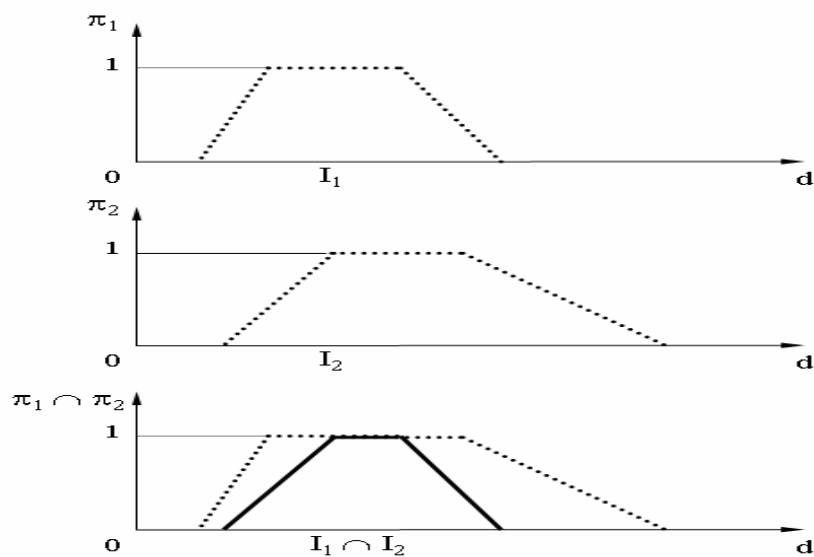


Fig. 3. Intersection of two fuzzy temporal constraints

### 3.2.2 Composition

The composition of two fuzzy temporal constraints is also a temporal constraint defined by the following:  $\{I_1, \pi_1\} \oplus \{I_2, \pi_2\} = \{I_1 + I_2, \text{Sup}_{d=d_1+d_2} \{\min\{\pi_1(d_1), \pi_2(d_2)\}\}\}, \forall d \in I_1 + I_2$ . The sum of two intervals is defined by the following:  $I_1 + I_2 = [a_1, b_1] + [a_2, b_2] = [a_1 + a_2, b_1 + b_2]$ . Figure 4 shows the composition of two fuzzy temporal constraints.

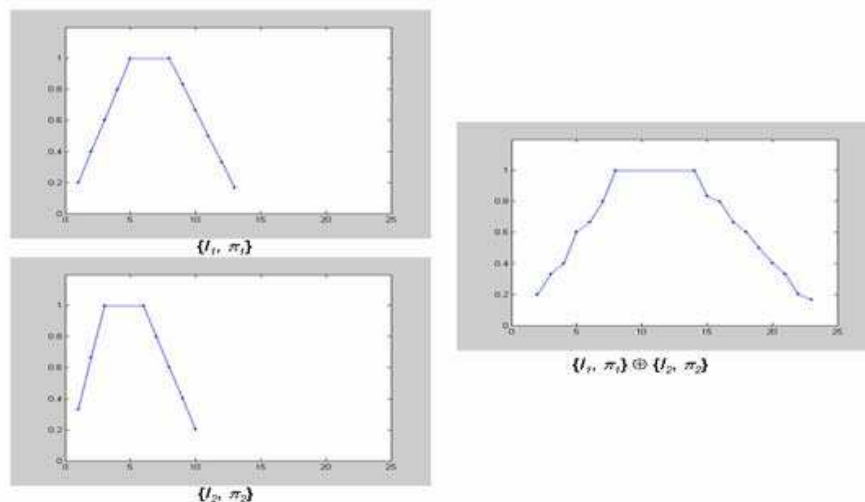


Fig. 4. Composition of two fuzzy temporal constraints

### 3.3 Operations on fuzzy temporal plans

This section focuses on different operations that may be performed on fuzzy temporal plans. The minimization operation is used to check the temporal consistence of a fuzzy temporal



plan. The intersection, augmentation and fusion operations are performed on two different plans and the result is a new fuzzy temporal plan. All these operations are graph-based since fuzzy temporal plans have graph structure.

### 3.3.1 Minimization

The building of a fuzzy temporal plan may entail temporal incoherencies. Usually, the user will build the graph of fuzzy temporal constraints without checking if there is a feasible solution that will respect all these constraints. To check these potential incoherencies, a minimization operation propagates the fuzzy temporal constraints within the graph in order to obtain its minimal version. When successful, this minimization will generate the minimal version of the graph. The failure of this operation means that the original graph contains temporal incoherencies. Two graphs express the same fuzzy temporal constraints if they have the same minimal graph (if we apply the minimization operation to those graphs). As a direct result, a minimal graph is equal to its minimized version. The minimization operation applies only to complete graphs. For this reason, any graph of fuzzy temporal constraints must be completed with universal fuzzy temporal constraints before minimization. We use the function  $comp(p_i)$  to complete the graph of the plan  $p_i$ .

The algorithm of minimization is the following:

```

for k=1 to n
  for i=1 to n
    for j=1 to n
       $\{I_{ij}, \pi_{ij}\} \leftarrow \{I_{ij}, \pi_{ij}\} \cap (\{I_{ik}, \pi_{ik}\} \oplus \{I_{kj}, \pi_{kj}\})$ 
    end
  end
end
end

```

In Figures 5 and 6, we show a graph of fuzzy temporal constraints and its minimized version.

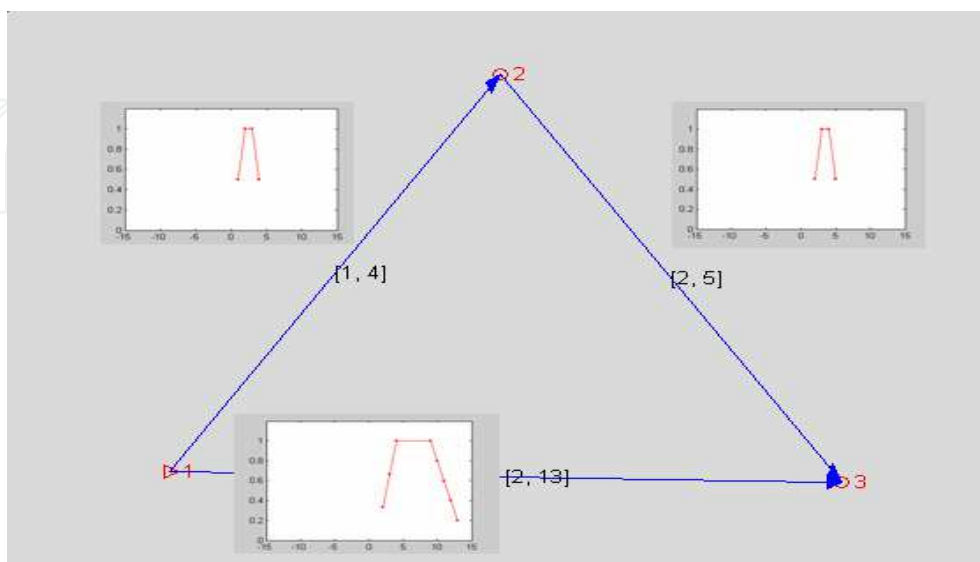


Fig. 5. Fuzzy temporal graph before minimization

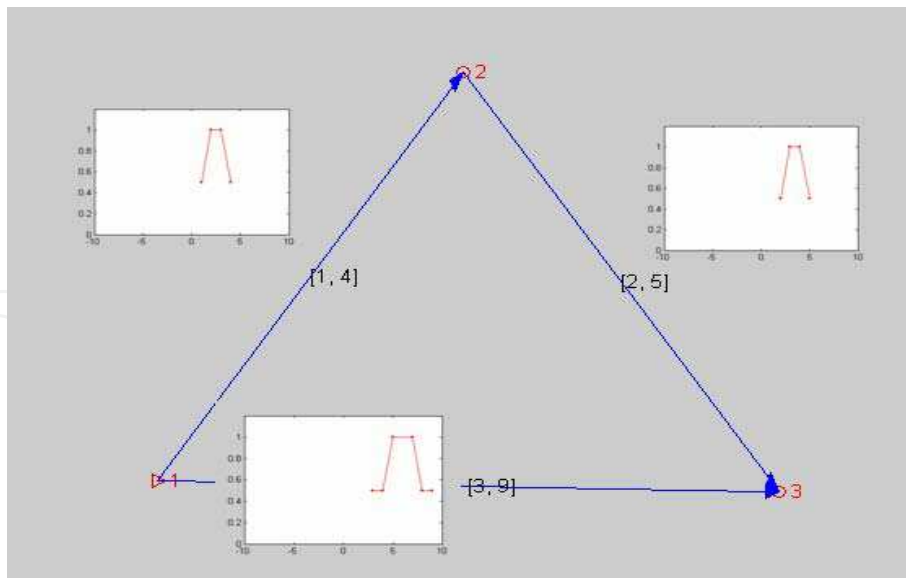


Fig. 6. Minimized version of the graph

Since a graph of fuzzy temporal constraints may be inconsistent, the minimization operation allows detecting a potential inconsistency as shown in Figure 7.

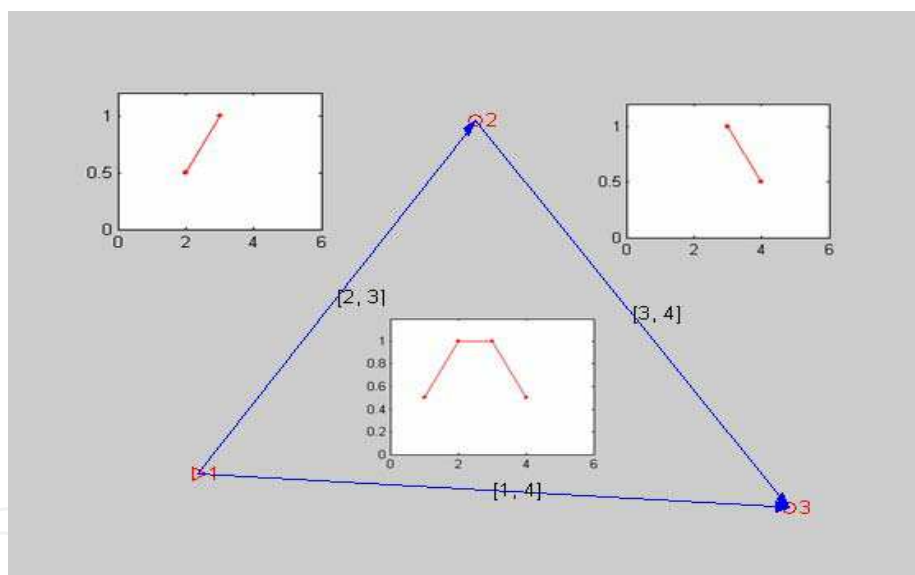


Fig. 7. Temporal inconsistency of a fuzzy temporal plan

In this figure, the composition of  $C_{12} = \{[2, 3], \{0.5, 1\}\}$  and  $C_{23} = \{[3, 4], \{1, .5\}\}$  is  $C_{12} \oplus C_{23} = \{[5, 7], \{.5, 1, .5\}\}$ . In the minimization algorithm, the expression:

$\{I_{13}, \pi_{13}\} \leftarrow \{I_{13}, \pi_{13}\} \cap (\{I_{12}, \pi_{12}\} \oplus \{I_{23}, \pi_{23}\}) = \{I_{13}, \pi_{13}\} \leftarrow \{[1, 4], \{.5, 1, 1, .5\}\} \cap (\{[2, 3], \{0.5, 1\}\} \oplus \{[3, 4], \{1, .5\}\}) = \{I_{13}, \pi_{13}\} \leftarrow \{[1, 4], \{.5, 1, 1, .5\}\} \cap \{[5, 7], \{.5, 1, .5\}\}$ . The intersection between  $\{[1, 4], \{.5, 1, 1, .5\}\}$  and  $\{[5, 7], \{.5, 1, .5\}\}$  is empty. This is due to the fact that the sum of the minimum distances between  $e_1e_2$  and  $e_2e_3$  is greater than the maximum distance  $e_1e_3$ :  $2+3 > 4$ .

The complexity of the minimization operation is  $O(n^3)$ , where  $n$  is the number of action nodes in the graph of the plan  $p_i$ . It is also important to mention that the modeling of a plan with several sub-plans will reduce the number of nodes in the graphs of the temporal plans, which can be executed and monitored in parallel by different agents.

### 3.3.2 Intersection

The intersection between two fuzzy temporal plans  $p_i = \{A_i, T_i\}$  and  $p_j = \{A_j, T_j\}$  is defined as follows:

$$p_i \cap p_j =_{\text{def.}} \{A_i \cap A_j, \underbrace{T_i \cap T_j}_{A_i \cap A_j}\}, \text{ where } \underbrace{T_i}_{A_i \cap A_j} \text{ is the restriction of } T_i \text{ on } A_i \cap A_j.$$

$T_i \cap T_j = \{C_{ab} \cap C_{cd} \mid C_{ab} = \{I_{ab}, \pi_{ab}\} \in T_i, C_{cd} = \{I_{cd}, \pi_{cd}\} \in T_j, e_{ar}, e_b \in A_i, e_c, e_d \in A_j, \{e_{ar}, e_b\} = \{e_c, e_d\}\}$ . In Figure 8, we illustrate the intersection between two fuzzy temporal plans. It shows the intersection between fuzzy temporal constraints belonging to both plans.

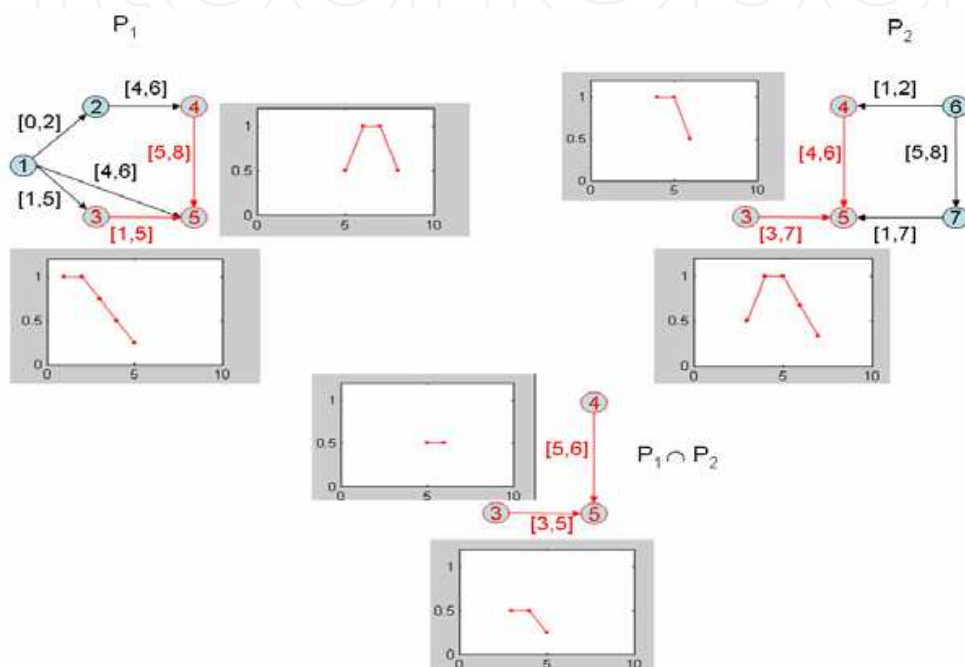


Fig. 8. Intersection of two fuzzy temporal plans

The computational complexity of the intersection operation is  $O(\max(n,m))$ , where  $n$  and  $m$  are the number of nodes in  $p_i$  and  $p_j$  respectively.

### 3.3.3 Augmentation

The augmentation operation adds to a plan, the nodes of another plan. This operation is necessary to perform binary operations on graphs of fuzzy temporal constraints:

$$\text{aug}(p_i, p_j) =_{\text{def.}} \text{comp}(\{A_i \cup A_j, T_i\})$$

This operation could also be defined as follows:

$$\text{aug}(p_i, p_j) =_{\text{def.}} \{A_i \cup A_j, \text{ext}(T_i, A_j)\}, \text{ where}$$

$\text{ext}(T_i, A_j)$  is the extension of  $T_i$  over  $A_j$ . It is defined by the following:

$$\text{ext}(T_i, A_j)(e_x, e_y) = \begin{cases} T_i(e_x, e_y) & \text{if } e_x, e_y \in A_i \\ \{[-\infty, +\infty], 1\} & \text{otherwise} \end{cases}$$

The computational complexity of the augmentation operation is  $O(n+m)$ .

### 3.3.4 Fusion of fuzzy temporal plans

The planning of a complex mission may require the participation of different planners (decision makers) from different backgrounds and with different perspectives. Very often this mission is decomposed into a set of sub-missions related with different types of relations. This decomposition draws links and dependencies between the different tasks that must be executed in order to fulfill the sub-missions. The planning of the different sub-missions is a process that can be carried out in parallel with different time constraints. The following example illustrates dependencies between two tasks. In the first task a truck must move from point A to point B. In the second task a tanker full of fuel must take another direction from C to D. If planned separately, the temporal constraints of these tasks should be independent. However, we add the following information: the truck does not have enough fuel to go from A to B and there is an intersection between the trajectory AB and CD. Based on this new information, it is clear that there must be a coordination between these two tasks and more specifically an adaptation of their temporal constraints to ensure their success. To this end, we introduce the fusion operation between two fuzzy temporal plans. The result of fusion is also a fuzzy temporal plan that can be executed by different players. It is called a coordinated plan. The fusion of several fuzzy temporal plans adapts their fuzzy temporal constraints in order to coordinate their execution. The fusion operation is a powerful tool that may be very useful when used in decentralized and distributed environments. It allows decentralized planning of activities and also a coordinated and distributed monitoring of their execution. The fusion of two temporal plans  $p_i$  and  $p_j$  is defined by the following:

$$p_i \oplus p_j \stackrel{\text{def.}}{=} \min(\text{aug}(p_i, p_j) \cap \text{aug}(p_j, p_i))$$

By developing this expression according to the definition of *aug* in Section 3.3.3 where  $p_i = \{A_i, T_i\}$  and  $p_j = \{A_j, T_j\}$ , it becomes:

$$\begin{aligned} p_i \oplus p_j &= \min(\{A_i \cup A_j, \text{ext}(T_i, A_j)\} \cap \{A_j \cup A_i, \text{ext}(T_j, A_i)\}) \\ &= \min(\{A_i \cup A_j, \text{ext}(T_i, A_j) \cap \text{ext}(T_j, A_i)\}). \end{aligned}$$

The intersection  $\text{ext}(T_i, A_j) \cap \text{ext}(T_j, A_i)$  is based on the intersection of fuzzy temporal constraints given in Section 3.2.1.

The fusion of two temporal plans is obtained by the union of their nodes and the merging of their fuzzy temporal constraints. Common fuzzy temporal constraints between  $p_i$  and  $p_j$  will be replaced by their intersection. In all other cases, the temporal constraints are unchanged. The coordinated plan must be minimized in order to detect a potential temporal inconsistency, which in this case means that it is not possible to coordinate the execution of the two fused plans and satisfy their corresponding temporal constraints. For this reason, the two temporal plans must be executed independently.

The computational complexity of the fusion operation is  $O((n+m)^3)$ .

The temporal plan fusion operation has two important properties that determine its context of use in a decentralized distributed environment.

#### Commutativity

The temporal fusion is commutative:

$$\begin{aligned} p_i \oplus p_j &= \min(\text{aug}(p_i, p_j) \cap \text{aug}(p_j, p_i)) \\ &= \min(\text{aug}(p_j, p_i) \cap \text{aug}(p_i, p_j)) \\ &= p_j \oplus p_i \end{aligned}$$

This property allows a group of players to fuse their plans regardless of the order in which the fusion is being performed.

### Associativity

The fusion of temporal plans is associative:

$$\begin{aligned} p_i \oplus (p_j \oplus p_k) &= p_i \oplus (\min(\{A_j \cup A_k, \text{ext}(T_j, A_k) \cap \text{ext}(T_k, A_j)\})) \\ &= \min(p_i \oplus \{A_j \cup A_k, \text{ext}(T_j, A_k) \cap \text{ext}(T_k, A_j)\}) \\ &= \min(\{A_i \cup A_j \cup A_k, \text{ext}(T_i, A_j \cup A_k) \cap \text{ext}(\text{ext}(T_j, A_k) \cap \text{ext}(T_k, A_j), A_i)\}) \\ &= \min(\{A_i \cup A_j \cup A_k, \text{ext}(T_i, A_j \cup A_k) \cap \text{ext}(T_j, A_k \cup A_i) \cap \text{ext}(T_k, A_i \cup A_j)\}). \end{aligned}$$

It is easy to demonstrate in the same way that  $(p_i \oplus p_j) \oplus p_k = \min(\{A_i \cup A_j \cup A_k, \text{ext}(T_i, A_j \cup A_k) \cap \text{ext}(T_j, A_k \cup A_i) \cap \text{ext}(T_k, A_i \cup A_j)\})$ .

This property allows a group of players to fuse any number of plans in any order to obtain the same coordinated plan.

## 3.4 Fuzzy temporal plan monitoring

The monitoring of a graph-structured fuzzy temporal plan differs from the monitoring of a threadlike plan. In the latter, each action is scheduled and executed according to its rank in the list. In graph-structured fuzzy temporal plans, each time an action is started/finished the fuzzy temporal constraints must be updated and then propagated throughout the graph by performing the minimization operation. This propagation tells if there still a solution for the execution of the plan while respecting the current fuzzy temporal constraints. It also gives the time left for an action to start or finish executing and the list of actions (candidates) that can be executed in the next step.

### 3.4.1 Timeout

The timeout is computed after each action start or end. Let  $e_i$  be the last occurred node in the graph, the timeout  $T$  is computed as the following: for all  $C_{ij} = \{[a_{ij}, b_{ij}], \pi_{ij}\} \mid b_{ij} \geq 0$ ,  $T = \min_j (b_{ij})$ .  $b_{ij}$  is noted  $\max(C_{ij})$ , hence,  $T = \min_j (\max(C_{ij}))$ . It corresponds to the minimum of maximum authorized times for the next node  $e_j$  to occur. In fact, past this time at least one action will be considered as too late to be carried out after  $e_i$ .

### 3.4.2 Candidate list

The candidate list  $L$  contains all the nodes that are authorized to occur after  $e_i$ . Obviously, an action from this list must be executed before the expiration of the timeout. Suppose that  $C_{is}$  is the fuzzy temporal constraint that allowed the computing of the timeout after the occurrence of  $e_i$ . All the nodes  $e_j$  such as  $C_{is} \cap C_{ij} \neq \emptyset$ , can occur (candidates) after  $e_i$ . This is true because it is possible to execute the corresponding actions before the timeout expiration and without violating the corresponding temporal constraints.

### 3.4.3 Propagation of fuzzy temporal constraints

The propagation of fuzzy temporal constraints must be performed each time an action starts or finishes executing. After propagation, the timeout and the candidate list are recomputed. When a new node in the graph occurs (the execution of an action starts or finishes), the fuzzy temporal constraint between this node and the previous occurred node is updated by



the new fuzzy temporal constraint  $\{[d, v]\}$  where  $d$  is the exact elapsed time between the two nodes and  $v \in [0, 1]$  is the possibility value associated to this distance. The choice of  $v$  can be user defined or based on different criteria. It is 1 when this distance is confirmed with certainty, and takes lower values otherwise. The constraint is propagated by minimizing the graph. The monitoring process is illustrated by the same example given in (Allouche & Boukhtouta, 2009) by adding fuzzy temporal constraints in the graph. The original minimal graph is illustrated in Figure 9. In this figure, the possibility functions of all fuzzy temporal constraints are represented in a 5x5 grid.

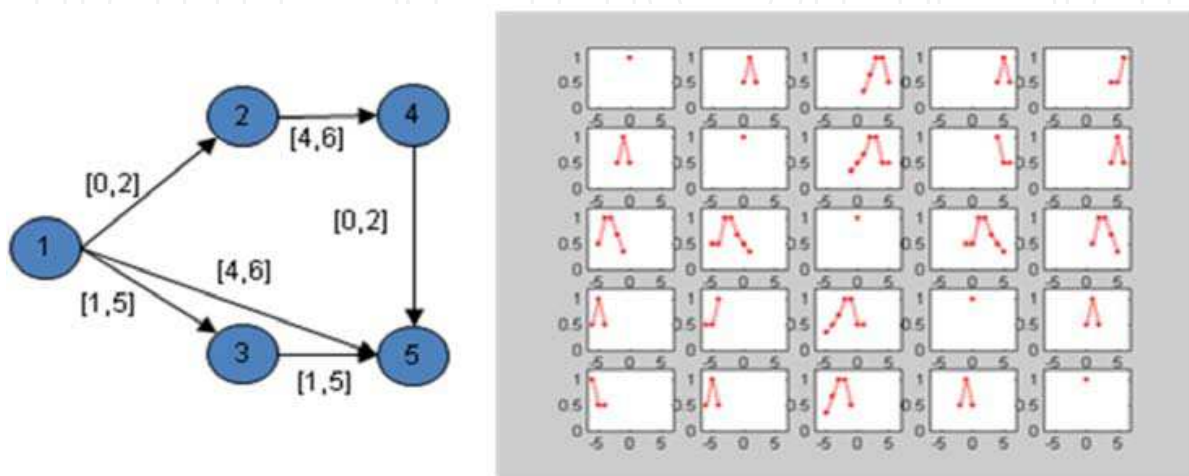


Fig. 9. Original minimal graph

In this example, we will not show the values of the timeout and candidate list since they are the same as in (Allouche & Boukhtouta, 2009). Only the change of fuzzy temporal constraints will be shown throughout the execution of the plan.

The execution of the plan starts with the occurrence of Node 1 at 15:06:12. Then Node 2 occurs at 15:06:13 with a possibility value = 1. The new graph is represented in Figure 10.

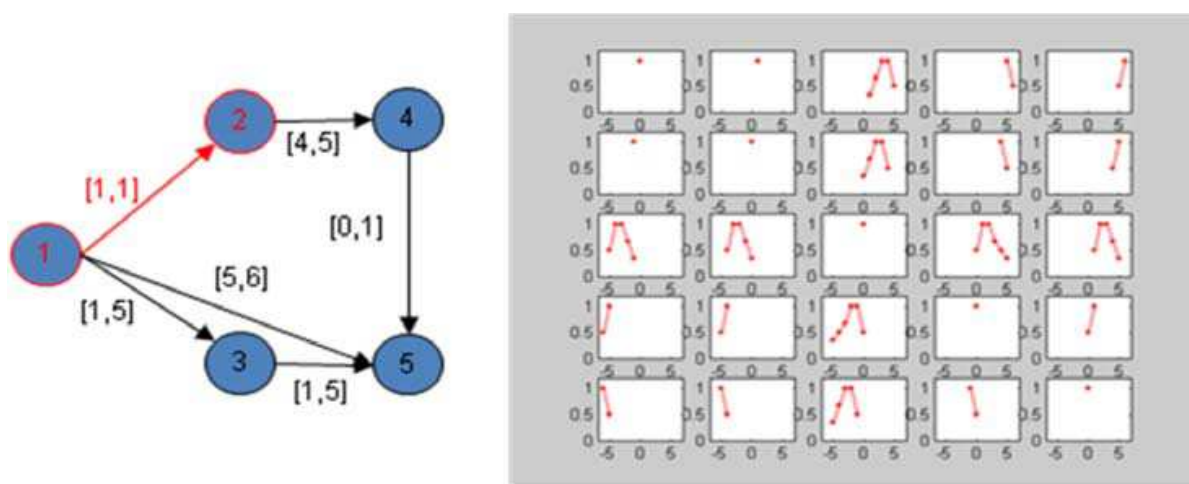


Fig. 10. Occurrence of the Node 2 at 15:06:13

Node 3 occurs at 15:06:15 with a possibility value = 0.5. The new graph is illustrated in Figure 11. It is important to see in this figure that all the fuzzy temporal constraints in the graph have now their possibility values at 0.5.



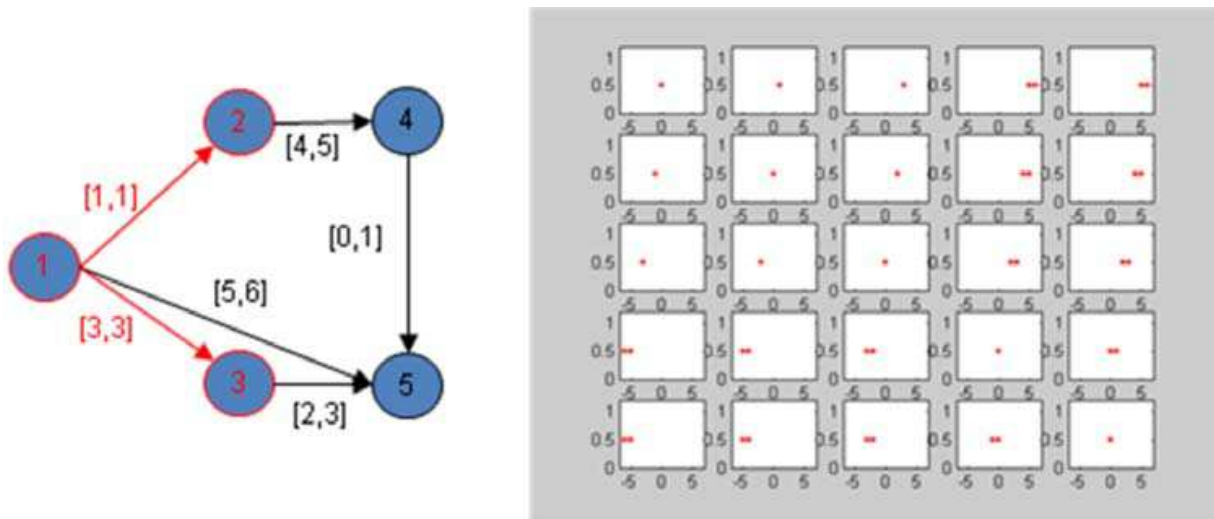


Fig. 11. Occurrence of Node 3 at 15:06:15

Finally Node 4 and Node 5 occur at 15:06:17 with possibility values .75 and 1 respectively. The result is depicted in Figure 12.

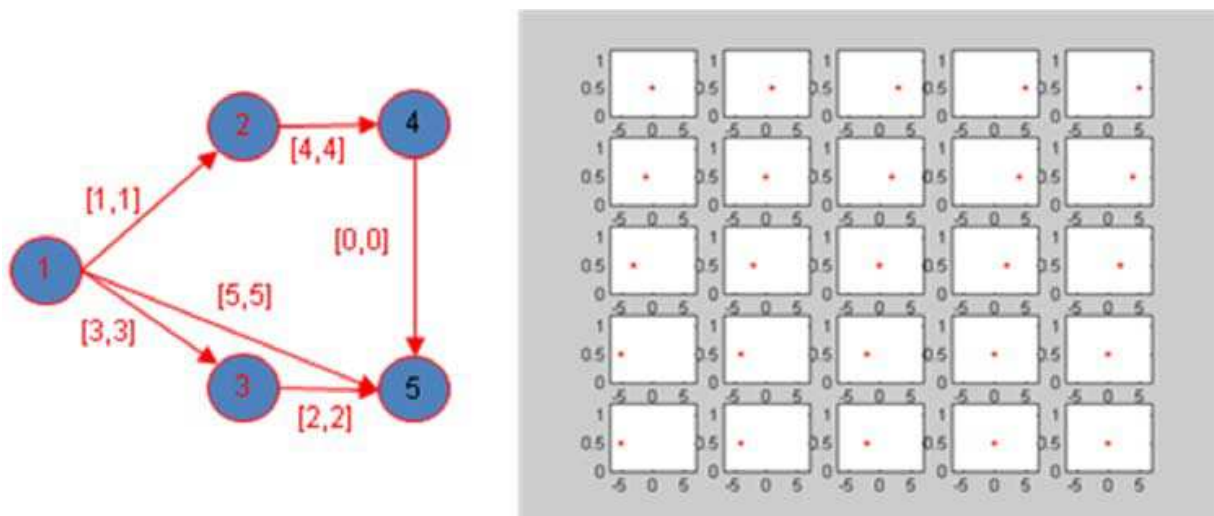


Fig. 12. Occurrence of Node 4 and Node 5 at 15:06:17

This example shows how the introduction of fuzzy information provides the monitoring process with more flexibility since the execution continued with possibility values less than 1. This flexibility should however be controlled by the decision-maker in order to give a meaning of a specific possibility value. In fact, the decision-maker is faced with two different problems. The first is to give a possibility value to the occurrence of a node. The second is to be able to qualify a possibility value in the graph that is less than 1. Mainly, the main question that the decision-maker would be eager to answer is: "should this specific possibility value be acceptable or not".

#### 4. Application to CSAR

The fuzzy plan fusion is applied to a CSAR (Combat Search and Rescue) mission in the context of the North Atlantis scenario. This fictitious scenario was used in 2000 as an

exercise by the Canadian Forces Command and Staff College (CFCSC) to teach the Canadian Forces Operational Planning Process and allow the sharing of operational knowledge and expertise among the CFC staff and students. The choice of this type of application is motivated by three main reasons: First, the temporal constraints are key elements in planning and the execution of CASR missions. Second, a CASR mission requires different types of assets distributed over the environment. Finally, a close coordination of the activities of the different involved assets is a key factor for the mission success.

A crisis has developed over the past 10 days on the continent of Atlantis. It is the result of years of growing tensions since the fall of 1999, and has now erupted into armed conflict. Individual country studies are provided as well as a document entitled "The Manghalour Peninsula Crisis," to provide the detailed background to the crisis.

As a result of the critical situation between ORANGELAND/REDLAND and BLUELAND, the UN requested the Alliance Council to consider a military response to help resolve the crisis.

On 12 June, the second day following the commencement of the Alliance joint operations to secure Blueland territory and expel any Coalition invasion forces, a UK Royal Air Force (RAF) Tornado call-sign HAWK27, conducting an electronic countermeasures and reconnaissance (ECR) mission, was shot down over the Celtic Straits by a surface-to-air missile (SAM) at 1608 hours. Shortly after the location of the downed crew was known, a CH-124 Sea King helicopter from Wahhabe Airbase, with a crew of five, was sent to recover and evacuate the Tornado aircrew. At approximately 1800 hours, in the process of extraction of the downed Tornado crew, the Sea King crashed.

A CSAR mission represents many dynamic challenges for the mission planners to locate and extract lost crew members in a hostile environment. Various elements must be taken into account, which may be predictable such as the friendly elements of detect and rescue, or unpredictable such as the enemy elements of detect and destroy. Usually, mission planners use air and ground picture inputs to make their decisions.

#### 4.1 Tasks

The PC (Package Commander) designed a plan to meet two critical mission requirements: air superiority and CSAR extraction. This plan should also allow SEAD (Suppression of Enemy Air Defenses) and air superiority established at least 15 minutes ahead, ABR (Airborne Regiment) delay/harassment occurring approximately 10 minutes ahead and CAS (Close Air Support) in place five (5) minutes ahead of the CSAR helicopter extraction area TOT (Time On Target). The plan includes tasks assignment to allocated assets to counter the enemy threat for "efficiency and safety" and to gain and maintain local air superiority:

- a. 4 x CF-18 - SIERRA 1-4 - sweep ingress and egress route and provide CAP (Combat Air Patrol) over CSAR pick-up area (above cloud);
- b. 4 x CF-18 - ECHO 1-4 - escort CSAR helicopters inbound and outbound to the pick-up area (below cloud with assets);
- c. 4 x CF-18 - BOMBER 1-4 - BAI (Battlefield Air Interdiction)(cluster munitions) pre-strike harass/delay of the LOC (Lines Of Communication) and ABR main body forward elements;
- d. 4 x ECR Tornado - JAMMER 1-4 - SEAD of Eaglevista SAMs from five (5) minutes before to five (5) minutes after mission aircraft enter AOO (Area Of Operations);

- e. 2 x ECR Tornado – ZAP 1-2 – SEAD of ABR SA 8 ahead of sweep aircraft and remain on station until all mission aircraft out of SA 8 range;
- f. 2 x CH 53 – RESCUE 1-2 – each with maximum JTF2 (Joint Task Force)(less seven (7) for downed crews) such that each can carry out mission if other helicopter aborts;
- g. 1 x AC-130 (Gunship) – GUNNER – for CAS in the target area;
- h. 1 x Predator UAV – PREDATOR 1 – to locate and monitor pick-up area;
- i. 1 x Predator UAV – PREDATOR 2 – to locate and assist in targeting ABR forward elements.

In this plan, the assets used such as the 4 x CF-18 in task c) for instance, correspond to agents that are responsible to execute this task. BOMBER 1-4 are the names of these agents. The organizational structure of agents and the means they have in order to form coalitions and help each other will not be discussed in this chapter.

#### 4.2 Temporal constraints

From the description of the plan and related tasks, it is possible to deduce different temporal constraints between the different tasks.

- f) during b);
- f) during a);
- g) starts at least 5 min. before f) starts;
- c) starts at least 10 min. before f) starts;
- d) starts at least 15 min. before f) starts;
- e) starts at least 15 min. before f) starts;
- f) during h);
- e) during i);
- f) during e);
- e) overlaps a);
- d) starts at least 5 min. before and continues at least 5 min. after e) starts;
- f) must be performed between 10 and 20 min.

#### 4.3 Temporal plans

The tasks described in Section 4.1 allow the definition of partial plans that must be executed by the different agents to insure the success of the CSAR mission. However, if the plans are executed separately, without taking into account the different temporal constraints, synchronisation problems may arise causing the whole CSAR mission to fail. For example, if f) is not executed during b), the extraction of the downed crew, at least in part, will be performed without escort, which will put the downed crew in danger. The idea is to define a plan for each agent or group of agents. Once the plans are defined, they must be fused to obtain a coordinated plan that can be executed by all the agents. Nine (9) plans can be defined from the tasks and related temporal constraints. They are shown in Figure 13. In these sub-plans the fuzzy constraints are those represented in figure 14. For each interval derived from the constraints defined in Section 4.2, we added 20% as a possibility distribution. The definition of this distribution is very important in order to cope with any unlikely delay that may alter the success of the mission if any of the temporal constraints defined in Section 4.2 is violated. In Figure 14, the fuzzy temporal constraint represented in (a) belongs to P3 and P7, (b) belongs to P4, (c) belongs to P6 and (d) belongs to P7 and P8.

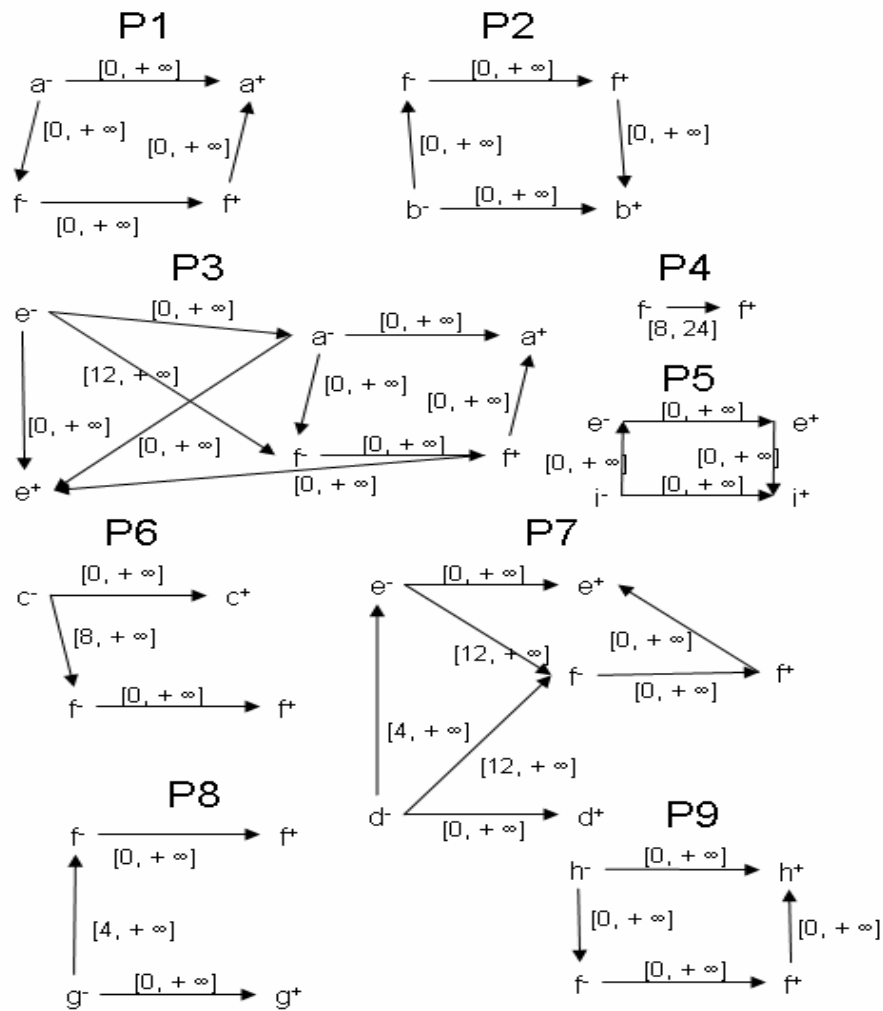


Fig. 13. Plans for CSAR mission

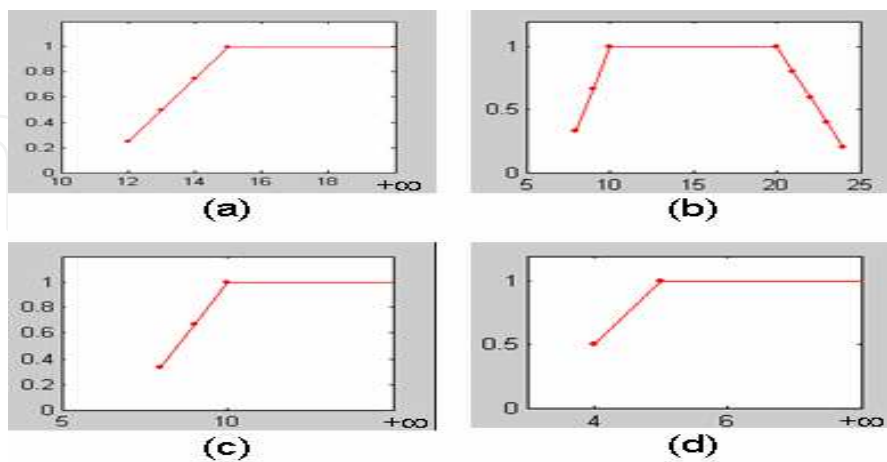


Fig. 14. Fuzzy temporal constraints in the sub-plans

#### 4.4 Temporal fusion

The fusion of the nine plans is shown in Figure 15.

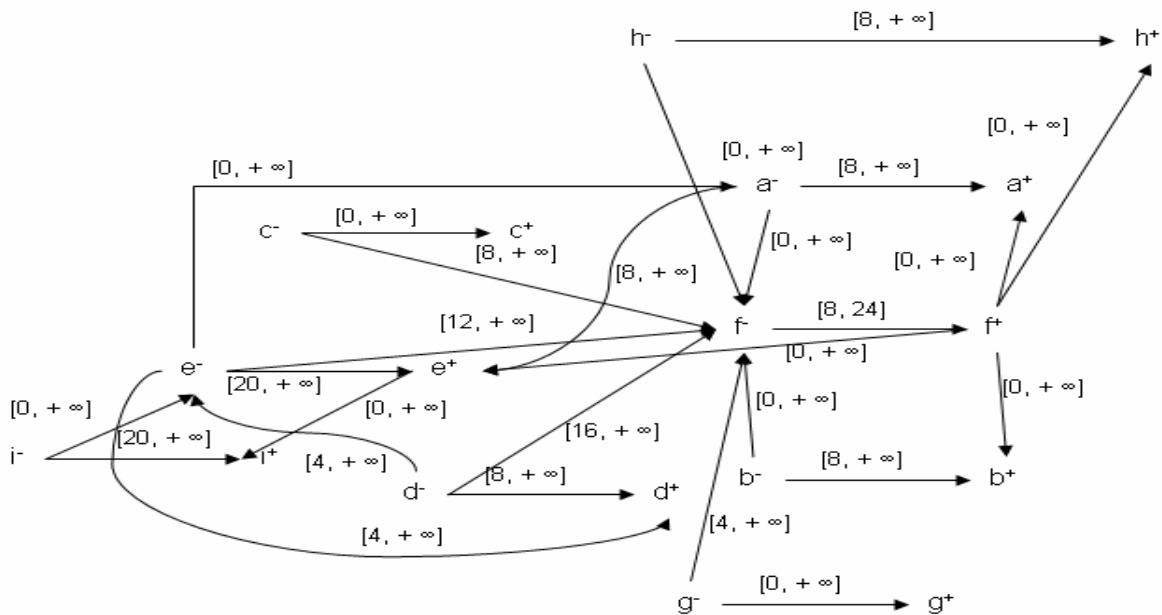


Fig. 15. Coordinated plan obtained by fuzzy temporal fusion of the nine plans

For the sake of clarity we show in Figure 16 only four fuzzy temporal constraints in the graph of the coordinated plan. Those constraints are the result of the propagation performed in the fusion process.

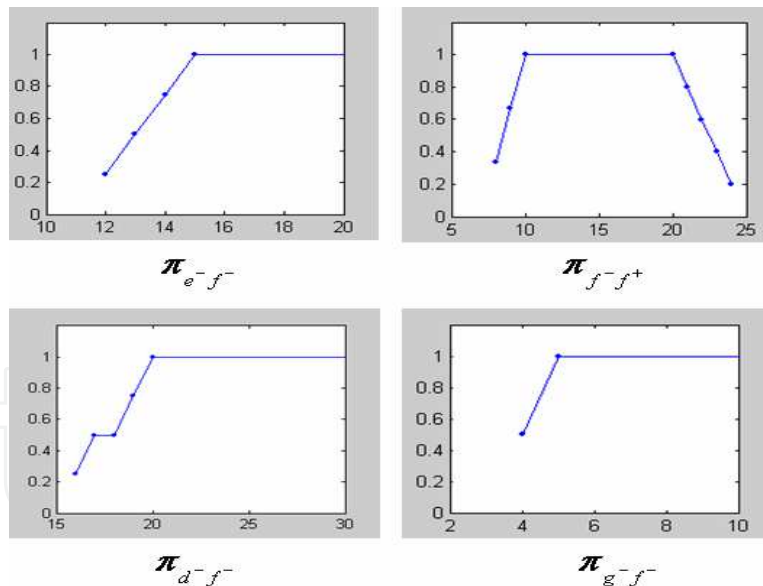


Fig. 16. Four fuzzy temporal constraints in the coordinated plan graph

It is also important to mention that for the sake of simplicity, details in the tasks such as take-offs, air-to-air refuelling, and landing were not taken into account in the plans. For example, the plan P4 is simply defined by the task f), which corresponds to the extraction of the downed crew. However, before the extraction, details such as the take-off of the rescue helicopters, the path followed before reaching the crash site, and finally the return to base, were not included in the plan. Also, all the communications between the PC, mission aircraft and other centres such as the Combined Air Operations Centre (CAOC) are not shown.



Some of the details will be shown in the timeline of the mission in Section 4.5. It is possible to use sub-plans to model such details. The use of sub-plans would allow different levels of abstraction of the problem.

#### 4.5 Mission execution

The following is a general timeline giving a global picture of the execution of the mission. Each element of this timeline includes the hour, the aircraft name and a summarized description of its activity. As mentioned before, even if this timeline is too detailed for the coordinated plan, it still contains little details compared to the original mission timeline. We think that this level of details is sufficient to illustrate the fuzzy temporal fusion and monitoring of the execution of a distributed CSAR mission.

The nodes in the coordinated plan are shown as they occur within the following timeline. We have chosen to show the execution of the coordinated plan at three time points within the timeline: at 11:15, then at the end of the extraction and finally at the end of the mission. At each time point, the propagation of the fuzzy temporal constraints is shown as well as the positions of the different aircraft in the area of operations. In this timeline, we suppose to know exactly when a node occurs. However, the formalism allows the use of fuzzy temporal constraints to express an uncertainty about the occurrence of nodes.

06:00 Magic (1 x AWACS) on station north at 5800N/2600W

07:00 Predator 1 (1 x UAV) take-off from Bendeguz, Exxon 1 (1 x KC-135 AAR (Air-to-Air Refuel)) on station Track A 5830N 2400W

08:00

**h- Predator 1 on station 5700N/2700W, detect and track downed aircrew, detect and track enemy forces**

09:00 Predator 2 (1 x UAV) take-off from Bendeguz

09:45 Echo 1-2 (2 x CF-18) take-off from Bendeguz

**i- Predator 2 on station 5730N/2630W, airborne backup**

10:00 Spook (1 x JSTARS) on station 5720N/2400W, detect and track enemy forces (ABR and SA-8 TELs), Jammer 1-4 (4 x Tornado ECR) take-off from Bendeguz, Bomber 1-4 (4 x CF-18) take-off from Bendeguz, Exxon 2 (1 x KC-135 AAR) on station Track B 5830N 2400W

10:15 Echo join AAR Track B

10:30 Zap 1-2 (2 x Tornado ECR) take-off from Bendeguz, Jammer 1-4 AAR Track A, Echo 1-2 departs AAR Track B, Bomber joins AAR Track B, Gunner (1 x AC-130) take-off from Nitric

10:35 Rescue 1-2 (2 x CH-53) take-off from Nitric

10:45 Echo 3-4 (2 x CF-18) take-off Bendeguz

11:00 Jammer 1-4 departs AAR, Zap 1-2 joins AAR Track A, Sierra 1-4 (4 x CF-18) take-off from Bendeguz, Bomber 1-4 departs AAR Track B

11:10 Predator 2 departs north hold to reposition to 5640N/2700W, Predator 3 (1 x UAV) takes-off from Bendeguz

11:15

- Rescue 1-2 turn south along coast
- b- Echo 1-2 join CSAR for close escort**
- Echo 3-4 join AAR Track B
- Zap 1-2 departs AAR Track A





Fig. 17. Assets positions at 11:15

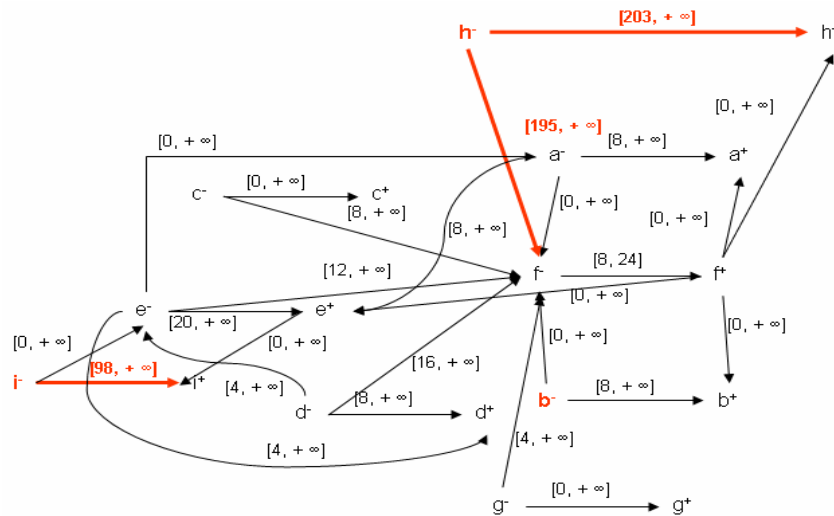


Fig. 18. Fuzzy temporal constraints of the coordinated plan at 11:15

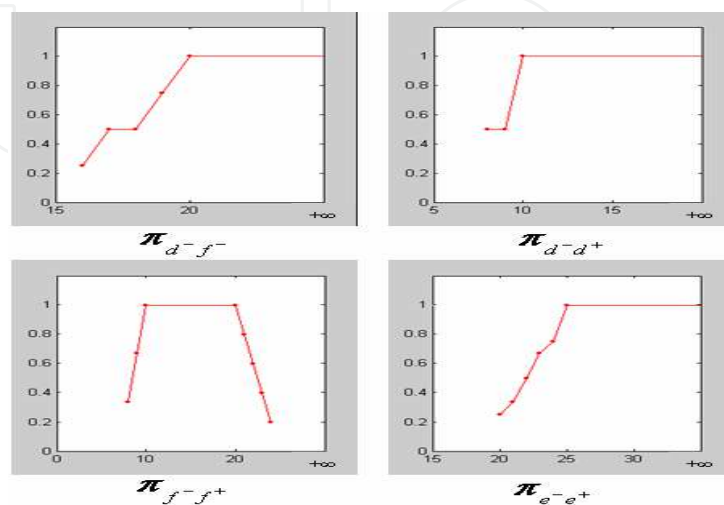


Fig. 19. Possibility functions of four fuzzy temporal constraints of the coordinated plan at 11:15

11:20

**d- Jammer 1-4 push from 5730N/2400W**

11:25 Jammer 1-4 (Low) ingress over Blueiland, SEAD at Eaglevista, Zap 1-2 push from 5750N/2500W

11:30 Sierra 1-4 Sweep, push from 5800N/2500W, Echo 3-4 depart AAR Track B, Predator 2 on station 5640N/2700W, Predator 1 repositions to 5700N/2730W

11:35 Bomber 1-4 push from 5800N/2900W

11:40 Gunner push from 5800N/2800W

11:45

**e- Zap 1-2 engaged SEAD SA-8**

**j- Predator 3 on station 5730N/2630W, airborne backup**

11:50

**a- Sierra 1-4 on CAP bullseye 5700N/2700W (southwest)**

**c- Bomber 1-4 TOT BAI LOC Cluster munitions**

11:55

**g- Gunner TOT CAS**

12:00

**f- Rescue 1-2 TOT extraction begins**

- Echo 1-2 provide top cover in target area
- Echo 3-4 arrive to provide top cover with Echo 1-2

**c+ Bomber 1-4 RTB (Return To Base) Bendeguz**

12:22

**f+ Rescue 1-2 extraction complete**

- Echo 1-2 RTB
- Echo 3-4 close escort CSAR egress

**g+ Gunner RTB Nitric**



Fig. 20. Assets positions at 12:22

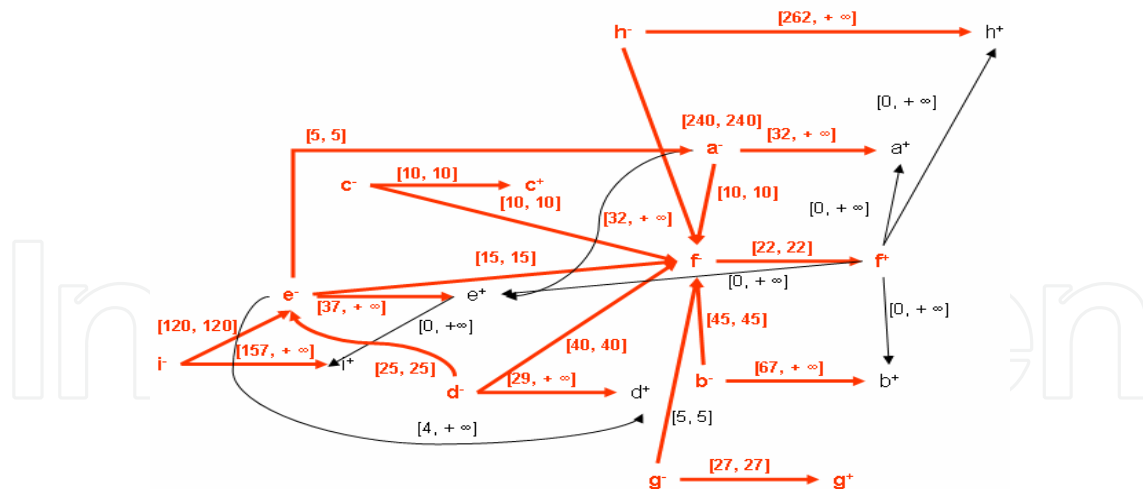


Fig. 21. Fuzzy temporal constraints of the coordinated plan at 12:22

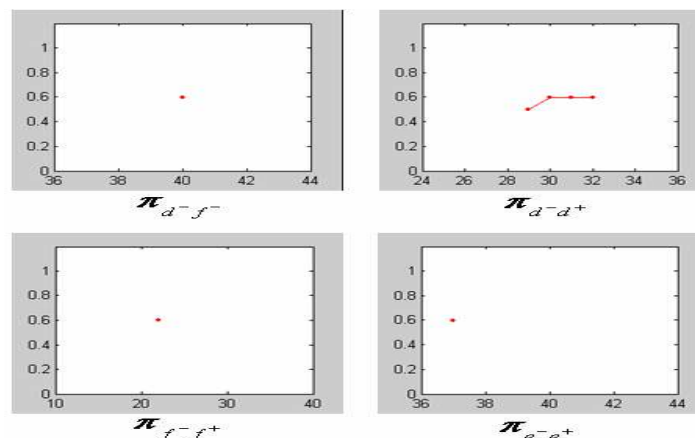


Fig. 22. Possibility functions of four fuzzy temporal constraints of the coordinated plan at 12:22

12:25

**d+** Jammer 1-4 RTB Bendeguz

12:30

**a+** Sierra 1-4 RTB Bendeguz

**e+** Zap 1-2 RTB Bendeguz

**h+i+** Predator 1-2 RTB Bendeguz

12:45

- Rescue 1-2 lands at Wahhabe

**b+** Echo 3-4 RTB Bendeguz

The propagation of the temporal constraints depicted in Figures 17-25, clearly show that the activity described in the timeline is compliant with the coordinated plan shown in Fig. 15. It is also important to mention that without the use of fuzzy temporal constraints the execution of the coordinated plan should fail. In fact, since the extraction task took 22 min, the possibility function of all the fuzzy temporal constraints has a value smaller than one as shown in Figure 22. For example, if the extraction of the downed crew had taken less than 20min, the propagation of temporal constraints would lead to possibility functions equal to 1. If the extraction had taken more than 24 min, it is then necessary either to change the plan

(re-planning) or to adapt the temporal constraints. In both cases the coordination of agents' activities is necessary.

### 5. Conclusion and perspectives

In this work, we propose a general framework for distributed fuzzy temporal plan modelling and monitoring. We believe that the explicit representation of time in plan modelling needs also to take into account the representation of uncertainty. This is due to the fact that in distributed environments where different activities may take place at the same time, it is sometimes difficult to manage the synchronisation of tasks with precision.

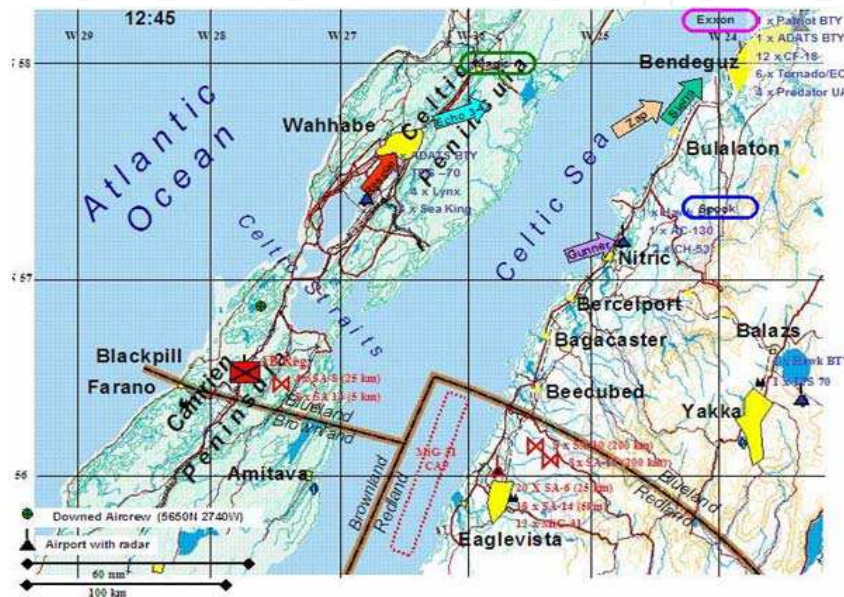


Fig. 23. Assets positions at the end of the mission

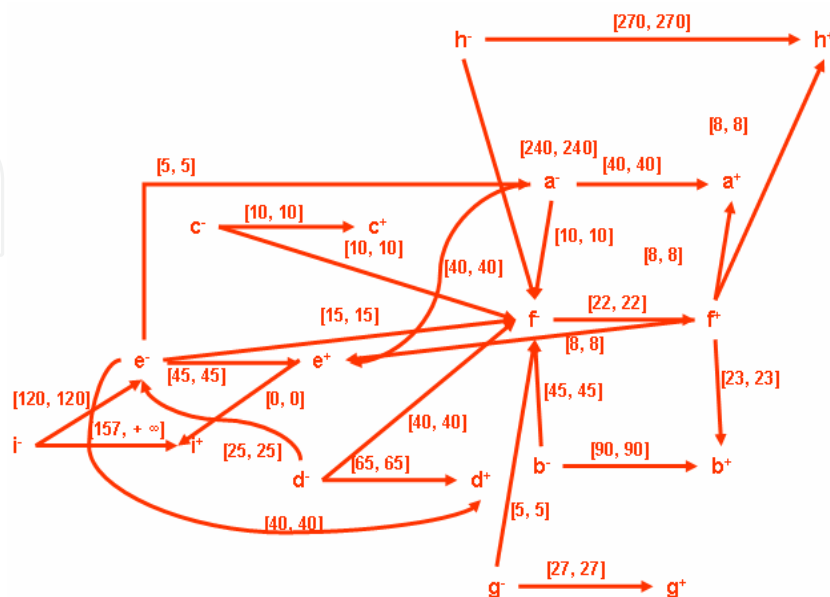


Fig. 24. Fuzzy temporal constraints of the coordinated plan at the end of the mission



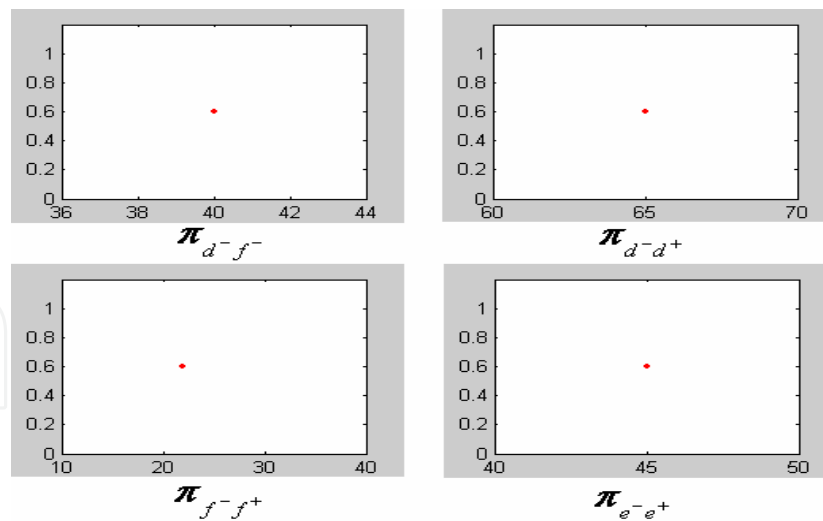


Fig. 25. Possibility functions of four fuzzy temporal constraints of the coordinated plan at the end of the mission

The fuzzy temporal plan model allows to model plans as a set of fuzzy and temporally constrained actions. Each action is modelled by two nodes (beginning and end). A fuzzy temporal constraint is defined by an interval where each value in this interval has a possibility value.

A propagation mechanism is defined in order to check the temporal consistence of the fuzzy temporal plan during execution.

In a distributed environment different activities are carried out simultaneously. This corresponds to the execution of different sub-plans by different players. A fusion operation is defined in order to fuse different sub-plans into a single plan called coordinated plan. A coordinated plan can be executed by different partners and the propagation mechanism is used to check its temporal consistence during the execution.

The approach only computes the possible solutions (coordinated plan) to execute distributed temporal sub-plans by different players. For instance, after each action execution, different actions may be candidate for execution. A decision must be made in order to choose the next action to be executed. In an ideal context, the proposed fusion and monitoring capability should be combined with a decision support capability and should keep the human operator in the loop to make decisions.

A CSAR mission was chosen to illustrate this work on a real-world scenario, where temporal aspects and uncertainty are key factors for the mission success.

One limitation of plan fusion is that each time a plan is fused with another plan, they become more temporally constrained. Hence, the fusion of large number of temporal plans tends to result in a temporally inconsistent coordinated plan. This indicates that plan fusion is useful in some coordination contexts but a re-planning activity may become unavoidable in some other cases. The introduction of fuzzy temporal constraints helps mitigate this problem since it is possible to extend the original temporal constraints with the appropriate possibility functions.

As future work, it would be interesting to give an interpretation of the possibility function values. For example, after the end of the extraction task, all the possibility functions have a maximum value of 0.6 as shown in Figure 22. This is due to the fact that the extraction task took 22 min, which exceeds the original definition of the corresponding temporal constraint given in Section 4.2. The question that needs to be answered is how a decision-maker should

interpret a value lower than 1 of the possibility function. This value could be used to define a measure of performance for the mission execution.

## 6. References

- Allen, J. F. Maintaining knowledge about temporal intervals. *Communications of the ACM* 26(11) pp.832-843, Nov. 1983.
- Allouche, M.K. and Boukhtouta A. (2009). Multi-Agent Coordination by Temporal Plan Fusion - Application to Combat Search and Rescue, *Information Fusion*, doi:10.1016/j.inffus.2009.09.005
- Badaloni, S., and Giacomini, M. (2000). Flexible temporal constraints. In *8th Conference on Information Processing and Management of Uncertainty in knowledge-Based System (IPMU 2000)*, pp. 1262-1269.
- Dechter, R., Meiri, I., and Pearl, J. (1991). Temporal constraint networks. *Artificial Intelligence*, 49(1-3), 61-95.
- Dubois, D., Fargier, H., and Prade, H. (2003a). Fuzzy scheduling: Modelling flexible constraints vs. coping with incomplete knowledge. *European Journal of Operational Research*, 147, 231-252.
- Dubois, D., HadjAli, A., and Prade, H. (2003b). Fuzziness and uncertainty in temporal reasoning. *Journal of Universal Computer Science*, 9(9).
- Falda, M., Rossi, F., Venable, K.B. (2007). Dynamic consistency in fuzzy conditional temporal problems, *PROC. COPLAS 2007, CP'07 WORKSHOP*.
- Godo L., Vila, L. (1995). Possibilistic Temporal Reasoning based on Fuzzy Temporal Constraints, *Proceedings of International Joint Conference on Artificial Intelligence IJCAI*.
- Khatib, L., Morris, P., Morris, R. A., and Rossi, F. (2001). Temporal constraint reasoning with preferences. In Nebel, B. (Ed.), *17th International Joint Conference on Artificial Intelligence*, (IJCAI 2001), pp. 322-327. Morgan Kaufmann.
- Lau, H. C., Ou, T., and Sim, M. (2005). Robust temporal constraint networks. In *Proc. of 17th IEEE Conf. on Tools with Artificial Intelligence (ICTAI'05)*, pp. 82-88 Hong Kong.
- Rossi, F., Sperduti, A., Venable, K., Khatib, L., Morris, P., and Morris, R. (2002). Learning and solving soft temporal constraints: An experimental study. In Van Hentenryck, P. (Ed.), *Principles and Practice of Constraint Programming, 8th International Conference (CP 2002)*, Vol. 2470 of LNCS, pp. 249-263. Springer.
- Rossi, F., and Venable, K.B., Yorke-Smith, N. (2006). Uncertainty in soft temporal constraint problems: a general framework and controllability algorithms for the fuzzy case, *Journal of Artificial Intelligence Research* 27.
- Schwalb, E. (1998). Temporal Constraints: A Survey. *Constraints*, Volume 3, Numbers 2-3.
- Schwalb, E., Dechter, R., (1997). Processing Disjunctions in Temporal Constraint Networks, *Artificial Intelligence*, 29-61.
- Tsamardinos, I., Pollack, M. E., and Ramakrishnan, S. (2003a). Assessing the probability of legal execution of plans with temporal uncertainty. In *Workshop on Planning Under Uncertainty and Incomplete Information of the Thirteenth International Conference on Automated Planning and Scheduling (ICAPS 2003)*.
- Tsamardinos, I., Vidal, T., Pollack, M. E. (2003b). CTP: A new constraint-based formalism for conditional, temporal planning. *Constraints*, 8(4), 365-388.
- Venable, K.B. (2005). Disjunctive temporal planning with uncertainty, *19th International Joint Conference on Artificial Intelligence*.



- Venable, K. B. (2005). Disjunctive temporal planning with uncertainty. *19th International Joint Conference on Artificial Intelligence IJCAI*.
- Vidal, T., Fargier, H. (1999). Handling contingency in temporal constraint networks: from consistency to controllabilities. *Journal of Experimental and Theoretical Artificial Intelligence*, 11(1), 23–45.
- Vila, L., and Godo, L. (1994). On fuzzy temporal constraint networks. *Mathware and Soft Computing*, 3, 315–334.
- Zadeh, L. A. (1975). Calculus of fuzzy restrictions. *Fuzzy Sets and their Applications to Cognitive and Decision Processes*, 1–40.

IntechOpen



## **Multi-Agent Systems - Modeling, Control, Programming, Simulations and Applications**

Edited by Dr. Faisal Alkhateeb

ISBN 978-953-307-174-9

Hard cover, 522 pages

**Publisher** InTech

**Published online** 01, April, 2011

**Published in print edition** April, 2011

A multi-agent system (MAS) is a system composed of multiple interacting intelligent agents. Multi-agent systems can be used to solve problems which are difficult or impossible for an individual agent or monolithic system to solve. Agent systems are open and extensible systems that allow for the deployment of autonomous and proactive software components. Multi-agent systems have been brought up and used in several application domains.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Mohamad K. Allouche and Jean Berger (2011). The Fusion of Fuzzy Temporal Plans: Managing Uncertainty and Time in Decentralized Command and Control Systems, Multi-Agent Systems - Modeling, Control, Programming, Simulations and Applications, Dr. Faisal Alkhateeb (Ed.), ISBN: 978-953-307-174-9, InTech, Available from: <http://www.intechopen.com/books/multi-agent-systems-modeling-control-programming-simulations-and-applications/the-fusion-of-fuzzy-temporal-plans-managing-uncertainty-and-time-in-decentralized-command-and-control>

# **INTECH**

open science | open minds

### **InTech Europe**

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia  
Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

### **InTech China**

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元  
Phone: +86-21-62489820  
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen