Temporal Reasoning and Interactive Planning

An Application to Forest Fire Fighting

Anna Perini, Francesco Ricci, Paolo Avesani
Istituto per la Ricerca Scientifica e Tecnologica
38050 Povo (TN)
Italy
e.mail: perini@irst.it, tel ++39+461-314330

Giornata di lavoro su Ragionamento Temporale, IV Convegno AI*IA.

Abstract

Constraint Satisfaction Problem (CSP) for variables with real domains has been proposed as a framework for dealing with temporal reasoning on metric information [6]. We have developed a temporal reasoning system based on this framework inside a system aimed at supporting the planning of initial attack to forest fires. The system implements a constraint reasoning language through an extension of a Object Oriented language. More generally the interactive planning system integrates skeletal planning and case based reasoning techniques with constraint reasoning. Temporal constraints are used in two steps of the planning process: plan fitting and adaptation, and resource scheduling. This planning system is part of a decision support system aimed at supporting the user in the whole process of the forest fire management including both situation assessment and planning activities. A first prototype of the planning system is currently being integrated with the situation assessment subsystem. [1]

1 The Application Problem

In this paper we describe an application of CSP techniques for variables with continuous domain to a problem of temporal reasoning that characterizes the process of planning the first attack in fire fighting. It has been designed with the aid of an OO methodology [2] and implemented within a specialized module devoted to constraint reasoning that extends the object oriented language Spoke [2].

In this section we describe the planning problem faced during the decisional process that leads to a first intervention for forest fire fighting. In the next section we discuss the planning approach remarking the role of temporal reasoning. Then, in the last two sections we describe how the temporal reasoning problem has been faced and the current status of the implementation work.

Planning a first attack to forest fire require to alternate phases of planning and situation assessment following a precise operational work-flow that is typical of each fire fighting organization. Here we focus on the planning problem. It can be briefly structured in the following steps:

- Formulating a goal. For example extinguishing the fire, retarding the fire, arresting the fire, evacuating the zone, reporting about a situation.
- Choosing the intervention strategy. For instance performing fire recognition, attacking the fire simultaneously on different sectors of the fire front ("sectorization"), or combining basic strategies.

[1] This work has been partially supported by the EspritIII project CHARADE (#6095).

[2] Spoke is an object oriented programming environment which runs on Unix workstation. It has been developed by Alcatel.

- Finding tactics that allow a correct strategy implementation. A tactic is a partially ordered set of actions implementing specific fire fighting techniques. Typical actions are the construction of a fire line, the transportation of means, the recognition activity of a scout. The correctness of a tactic refers to the quantity of water, the resources to be employed and the temporal deadline associated to the corresponding strategy.

- Assigning the resources. The actions require certain type and quantity of resources. Resources have to be taken from bases or other fires and allocated to actions taking into account time constraints of the plan itself and constraints on resource allocation periods.

2 The Technical Approach

In defining the planning approach for our system we have been driven by domain requirements and constraints. In another document (CHARADE restricted report RR51, Aproaches and techniques for AI planning) we have discussed in detail the domain requirements and the implications of those requirements on the choice of AI techniques for planning. In the conclusions we suggested that the integration of different techniques, and in particular of Case-Based Reasoning [8] and Constraint Reasoning is an adequate baseline for implementing first intervention plans in a system able to support a strong user interaction (see also [10] for a similar approach). Combining different specialized AI techniques is common in dealing with real planning problems. A remarkable work in this direction is ongoing within the ARPA & ROME Laboratory Knowledge Based Planning Initiative (partially described in [15]).

The planning problem associated to a given fire alarm is described in terms of the emergency scenario which is defined by a set of fire parameters like fire location, meteo conditions, topography, available resources that are located in the bases close to the fire and a set of appropriate goal such as extinguishing the fire, evacuating a zone or performing a recognition within time deadlines. Finding a strategy means to find an association between the description of the current scenario and one of the preclassified possible strategies. The refinement of a selected strategy into appropriate tactics is performed through Case Based Reasoning (CBR) techniques. The retrieval of tactis is performed by a partial matching algorithm that looks for goals similar to the one associated to the strategical step to be expanded and to scenario descriptions similar to the current one.

A stored tactic contains also the recipe for building the temporal dimension of a plan listing the temporal relation between actions. These information are expressed as a set of temporal constraints defined on the temporal variables, such as the starting and ending times of actions. Fitting an old sector plan means to change some action parameter values according to the features of the current scenario and then to recompute the constraints involving them. These constraints represents, for instance, how the number of squads, the type of action they perform and the length of the fire front on which they will operate, define the action duration. At this level constraints are mainly used for value propagation. Plan Adaptation is performed by the operator by changing the plan structure, for instance deleting an action or adding a new one, by modifying an action feature or the temporal relations between actions of the plan. For instance the operator can decide to add the action of spraying fire retardant with an helicopter to a plan which didn't include it. The correctness of the resulting intervention plan in checked against constraints related the global quantity of water to be used, available resources and time deadline. The latter step of the planning process consists in finding a resource schedule suitable for the global plan. Temporal constraints on the allocation periods of the resources to be allocated come from the temporal constraints representing action duration and ordering relations in the intervention plan. The resulting constraint networks on the allocation periods of the resources may be enriched by additional constraints representing resource characteristics. For instance a 4 hours shift for volunteer squads or the fact that an helicopter cannot operate more than 2 hours. Solution of this constraint network will be computed taking into account heuristic criteria for resource selection such as that of avoiding to empty a base allocating all its resources or that of trying to reuse resources sent to a given location, assigning them to different, not simultaneous, activities.

3 The Temporal Reasoning Framework

In our planning system the structure of a fire intervention plan is represented as a network of plan steps (actions) whose start and end times are constrained by a set of temporal relations such as action A_1 starts at least 30 minutes before action A_2, action A_1 lasts at least 2h , the whole plan must be finished in 3h. The typical temporal reasoning tasks that are performed by the planner during plan fitting and adaptation can be stated as follows: recompute and propagate plan step durations, check if a plan can be performed before a given temporal deadline, check if a new action can be inserted in the plan network without violating the deadline, compute earliest and latest start (end) times for the action (the plan). This require to deal mainly with metric information.

Among the relevant approaches to temporal reasoning with metric information that can be found in the AI literature [14, 5, 6, 12] the Temporal CSP (TCSP) of Dechter et al. [6] seemed to be the most appropriate to our objectives for two main reasons: first this approach provides an adequate expressivness for the kind of temporal information that characterize our problem domain, second it can be implemented as a specialization of a set of CSP techniques that we implemented in our planning system in order to provide the possibility to take into account other domain constraints, such as those related to the resource usability, that can be dealt by CSP for variables with discrete domain. That is this choice goes towords the more general objective of facing all the constraint reasoning tasks related to the fire-fighting planning process within an homogeneous CSP framework.

The TCSP framework [6] extends a previuos work by Davis [4] and enriches the expressivness of temporal reasoning approaches like the TMM [5]. It allows the encoding of all temporal information in terms of distance constraints between time points that can represent starting and ending points of an interval as well as meaningfull time points, such as a deadline.

Qualitative relationships can be represented using infinity values as distance bounds. For example "x before or equal y" is expressed constraining the duration between x and y to be between zero and infinity.

Moreover it allows to represent also disjunctive information like "action A ends about 10-20 minutes or about 40-60 minutes before that the action B starts" or "action A ends about 10-20 minutes before that action C starts or about 30-40 minutes after C finishes". But this require to deal with an NP-hard problem. Recent applications of the TCSP formalism can be found in [1, 3].

The current version of our system implements a simplified version of the TCSP formalism in which all constraints are expressed by a single disjunct relation (it is called Simple TCSP: STP). For this case Dechter et al. [6] propose two approaches. The first one is based on the representation of the CSP network through directed weighted graphs on which all-shortest-path algorithms can be run to obtain solutions to CSP computations such as checking the network consistency, computing the minimal network and finding the feasible values of the network variables. The second approach rests on the exploitation of local consistency computations. Algorithms defined for CSP defined on discrete variables are proved to terminate and to produce the desired results. We have implemented the first approach and we are studing on the possibility to adapt to the continuous case some local consistency techniques developed for CSPs defined on variables with discrete domains following some recent work on this problem [13, 11, 9]. A second aspect we are interested in is that concerning the development of incremental alghorithms required by the fact that the planning system suports an interactive process. At the moment we can already support dynamic modifications of the plan structure, like inserting a new action or a new temporal relation. But this relies on a straightforward modification of the general (non dynamic) algorithm. We are reflecting on the improvement of this first naive approach.

4 The Constraint Language

We have extend the object hierarchy of the Spoke language with a new set of classes that enable problem solving in a broad set of constraint satisfaction problems. This integration of constraints and objects has been developped taking into account basic requirements such as that of maintaining all the flexibility of the object-oriented languages and that of constraints being objects themselves, as extensively discussed in [7]. Variables with real number domains and binary constraints representing distance bounds between two real variables are specialisations of the more general classes representing generic CSP variables and constraints. Moreover the class representing temporal constraints network is a specialisation of the constraint network class whose variables are restricted to be real variables and whose constraints are distance bound constraints. Methods are defined on the constraint network class for building and updating constraint network instance, for running propagation and solution algorithms on them. As mentioned above this propagation methods have been specialized for the temporal network class correspondingly to the graph based approach proposed for the STP framework. Work is in progress for the adaptation of arc consistency and path consistency algorithms defined for discrete CSP to the real domain case.

References

[1] M. Boddy. Temporal reasoning for planning and scheduling. *SIGART*, 1993.

[2] G. Booch. *Object oriented design with applications.* The Benjamin/Cummings Publishing Company, 1991.

[3] R. Cervoni, A. Cesta, and A. Oddi. Maintaining consistency in quantitative temporal constraint networks for planning and scheduling. In *Proceedings of the National Conference of the Italian Association for AI (AI*IA)*, pages 279–289, 1993.

[4] E. Davis. Constraint propagation with interval labels. *Artificial Intelligence*, 32:281–331, 1987.

[5] T. Dean and M. Boddy. Incremental causal reasoning. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, pages 196–201, 1987.

[6] R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence*, 49, 1991.

[7] B. N. Freeman-Benson and A. Borning. Integrating constraints with an object-oriented language. In *Proceedings of the 1992 european conference on object-oriented programming*, pages 268–286, 1992.

[8] K. J. Hammond. *Case-Based Planning: Viewing Planning as a Memory Task.* Academic Press, Boston, 1989.

[9] D. Haroud and B. Faltings. Global consistency for continuous constraints. In *Proceedings of the 11th European Conference on Artificial Intelligence*, 1994.

[10] T. R. Hinrichs. Towards an architecture for open world problem solving. In J. L. Kolodner, editor, *Proceedings of the 1988 DARPA Workshop on Case-Based Reasoning*, pages 182–189, 1988.

[11] O. Lhomme. Consistency techniques for numeric CSPs. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, Chambéry, France*, pages 232–238, 1993.

[12] J. Rit. Propagating temporal constraints for scheduling. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-86), Philadelphia*, 1986.

[13] G. Sidebottom and W. Havens. Hierarchical arc consistency for disjoint real intervals in constraint logic programming. *Computational Intelligence*, 8(4):601–623, 1993.

[14] S. Vere. Planning in time: Windows and durations for activities and goals. *IEEE*, 1983.

[15] D. E. Wilkins and R. V. Desimone. Applying an ai planner to military operation planning. In M. Zweben and M. Fox, editors, *Intelligent Scheduling.* Morgan-Kaufmann, San Mateo, 1992.