Constraint Based Spatial and Temporal Reasoning

Position Paper - CoPiCAD-94

Francesco Ricci and **Anna Perini**

Istituto per la Ricerca Scientifica e Tecnologica
I38050 Povo (TN), Italy
e.mail: ricci@irst.it, tel ++39+461-314334

1 Research Interests

In this section we shall briefly summarize some research topics we are interested in. They are motivated by the desire to apply constraint reasoning to applications were spatial and temporal knowledge plays a key role. The results presented in this section constitute the technical baseline for developing the new ideas presented in Section 2.

1.1 Continuous Domains

Continuous constraints processing is a fruitful and intriguing area of research that is having now a new acceleration since the seminal work of E. Davis [3]. A number of problems are still open, but in particular many researchers are now trying to apply partial consistency techniques to different classes of constraints whose variables take values in real intervals [5, 8, 10].

We are pursuing a research project aimed at developing a constraint reasoning module on the top of an Object-Oriented Language. This work is motivated by the desire to have at disposal a tool, easily integrable into application packages, that can support problem definition and solving for Constraint Satisfaction Problems (CSPs) arising in real domains. This work is proceeding along a line introduced some time ago [1] and according to a set of standard goals for developing true Object-Oriented Constraint Languages [6].

In this module both continuous domain constraints and discrete domain constraints are considered. We use specific OO features, like polymorphism and inheritance, to build generic procedures that can be used in both cases. Having in mind the application of this module to really hard problems we have introduced:

- the management of a set of constraints defined on two variables with very similar motivations of that discussed in [5] (Total Constraint). We make a distinction between a constraint and a constraint arc. The first one is the usual constraint, the second one represents the dynamically defined collection of constraints between two variables. Furthermore, when one considers a particular class of constraints it is possible to take advantage of the restriction and develop ad hoc representations that enable efficient management during constraint reasoning. For instance, in the case of bounded difference constraints [4] ($x - y \leq a$) the constraint arc, if disjunctions are not allowed, can be defined using only two, possibly infinite, real values ($b \leq x - y \leq a$). General linear constraints would require a finite number of points in the real plane (vertices of a polyhedron).

- some support for dynamic control and propagation of constraints. We can interleave network construction with network propagation and we can support conditional propagation of new information (new constraints and new domains). This is done introducing functions for adding new variables, new constraints and modifying domains. Furthermore these functions can be called within two modes: *straight*, the changes are performed without any constraint check; *tentative*, a constraint check is performed before applying the required changes to the network. To develop these functions a method for storing a network state has been developed.

At the moment, for the continuous case, we have implemented a first part of the code that manages bounded differences constraints. The current focus on this constraints type is motivated by an application in planning and scheduling, that is briefly described in the following. We exploit both graph based algorithms (shortest-path) and k-consistency algorithms based on generic revise and composition procedures.

1.2 Case-Based Planning and Temporal Reasoning

We are developing an interactive planner to be used in an application of planning initial attack to forest fires [2, 13, 14]. This work is partially supported by the Esprit Project CHARADE (Combining Human Assessment and Reasoning Aids for Decision Making in environmental Emergencies). This system is based on a hybrid architecture for planning that integrates a Case-Based Planner [7] and a Constraint Reasoner [9].

- **The case based module** retrieves partial plans from memory using a similarity metric that takes into account not only a set of predictive features, as it is usually done in classical approaches, but also the plan structure itself. Plans are represented in a hierarchy of parts. Partial plans can be completed using the similarity metric by searching for expansions of the leaves in the plan tree structure.

- **The constraint module** adapts the retrieved plan attaching and propagating constraints. Constraints represent the temporal relations between actions in plans and participate to the definition of set of applicable actions. The temporal dimension of the plan is managed with a quantitative approach based on temporal variables on continuous domains [4].

The planner is also integrated with a Geographical Information System that provides support for spatial analysis of the fire zone. The GIS supports thematic mapping (roads, rivers, bases, etc.) and it is possible to draw on the map sector lines that represent the zones where some actions (water spraying, control line, back fire, etc.) have to be executed. This spatial information is used both in plan retrieval and temporal reasoning. For example the duration of an action is calculated taking into account the length of the sector line where it has to be executed, the type of work and the number of men in the involved squad.

1.3 Learning Automata

Another research interest is the study of stochastic algorithms for constraint solving. In a recent paper [12] we present the model of a decision maker that exhibits adaptive behavior in highly uncertain stochastic environments (Learning automata [11]) and we exploit it in solving finite domain CSPs.

Within this approach the probabilities of different values assignments are updated according to a procedure that can be considered as hill climbing in probability space. At every stage the changes in probabilities are such that a performance criterion is improved monotonically in an expected sense. Under certain conditions this is shown to result in convergence to the optimal action with a probability arbitrarily close to one. All past Learning Automata approaches adopt slow learning and, the majority of them, absolutely expedient reinforcement schemes. That comes from the ϵ-optimality property proven for these algorithms in stationary environments. We investigate and exploit the use of a fast learning algorithm that relaxes the previously mentioned assumptions but takes advantage of the fact that stochasticity, in the CSP application, derives only from lack of knowledge of the other agents' choices. The optimality of the proposed learning model is proven, that is the learning process converges with probability 1 to a solution of the CSP. Moreover, a set of examples show that good performance can be achieved. In particular, this method achieves a higher level of performance than that presented in [15], whereas it shares with the same approach the property of being free from local minima.

Learning Automata have been already applied to line drawing interpretation [15]. We would like to develop more significant experiments in automatic feature extraction from digital maps.

2 Future Researches

We list here some points (very preliminary ideas) we wish to put up for discussion. These research themes mostly concern applications where spatial and temporal knowledge are both exploited and integrated.

Spatial Reasoning and Planning Our current architecture for planning uses spatial information only to a limited extent. Essentially the length of a sector line is calculated and some other data linked to a sector line are retrieved by the GIS (average slope, main vegetation, etc). In fact human planners use extensively data from the territory to effectively build intervention plans. The interaction between the temporal dimension of a plan and the spatial information contained in the map should be further investigated. For example, currently spatial data are used to evaluate temporal durations. But one can also search to constrain spatial data with temporal information: for instance, a delay in the deployment of a resource can cause a sector line to be redrawed (moved back with respect to the fire origin). In order to face this subject a real spatial and temporal model is required.

Interpretation of Line Drawings We are interested in applying a hybrid architecture to automatic feature extraction from digital maps. In particular we would like to face the problem of automatic classification of linear contours in a CAD design. In this application we guess that learning automata, or more in general anytime algorithms, can be used to produce partially interpreted drawings.

Case-Based Process Planning In the past we have developed a system for building a process plan from a CAD design. That system was based on generative techniques. The plan was generated with a search process where the search states were produced by simulating the effects of a milling machine. We would like to discuss the idea of applying a different approach, namely a hybrid architecture case-based/constraint-based where plans are stored in a case-base and constraints are used to adapt a partially correct plan.

References

[1] P. Avesani, A. Perini, and F. Ricci. COOL: An object system with constraints. In J. Bézivin, B. Meyer, and J.-M. Nerson, editors, *Technology of Object-Oriented Languages and Systems*, pages 221–228, 1990.

[2] P. Avesani, A. Perini, and F. Ricci. Combining CBR and constraint reasoning in planning forest fire fighting. In *Proceedings of the first european workshop on Case-Based reasoning*, pages 235–239, Kaiserslautern, 1993.

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

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

[5] B. Faltings. Arc-consistency for continuous variables. *Artificial Intelligence*, 65:363–376, 1994.

[6] 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.

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

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

[9] 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.

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

[11] K. S. Narendra and M. A. Thathachar. *Learning Automata.* Prentice-Hall, 1989.

[12] F. Ricci. Constraint reasoning with learning automata. *International Journal of Intelligent Systems*, 9, 1994. to be published.

[13] F. Ricci, S. Mam, P. Marti, V. Normand, and P. Olmo. CHARADE: a platform for emergencies management systems. Technical Report 9404-07, IRST, 1994.

[14] F. Ricci, A. Perini, and P. Avesani. Planning in a complex real domain. In *proceedings of the italian planning workshop*, pages 55–60, Rome, 1993.

[15] M. A. L. Thathachar and P. S. Sastry. Relaxation labeling with learning automata. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8:256–268, 1986.