Combined Constraint Models in Planning: Where to Search?

Roman Barták*, Daniel Toropila*†
{roman.bartak, daniel.toropila}@mff.cuni.cz

*Charles University, Faculty of Mathematics and Physics
Malostranské nám. 2/25, 118 00 Praha 1, Czech Republic

†Charles University, Computer Science Center
Ovocný trh 5, 116 36 Praha 1, Czech Republic

Abstract

One of the ways how to improve the efficiency of constraint models is combining several models of the problem, which is a form of adding redundant constraints to the model. Such a combined model contains redundant variables that are not necessary to define the solution but that should be instantiated either during search or via inference. This paper opens the question which variables in such a combined model for planning problems should participate in the search procedure, that is, where to search.

Introduction

Constraint satisfaction problem (CSP) is formulated as a finite set of decision variables, each having a finite of set of possible values (domain), and a set of constraints that restrict possible combinations of values to be assigned to the variables. A feasible solution to a CSP is a complete instantiation of the variables satisfying all the constraints. A typical method to solve CSPs is based on the combination of search and inference, where the inference is implemented as a form of maintaining some level of local constraint consistency and the search algorithm gradually splits the search space by exploring exclusive disjunctions of constraints (such as assigning a value to the variable and removing that value from the domain of the variable).

Constraint modeling, that is deciding how to describe a problem as a CSP, has a critical role in problem solving using constraints because the decision variables together with their domains define the search space and the constraints influence how much the search space is pruned via inference. Frequently, there exist several constraint models to describe the problem and it can be beneficial to combine these models rather than to select between them. The combination is done via so called *channeling constraints* that express the relation between the variables in both models, in particular transfer the instantiation of the variables in one model to the other model (and vice versa). The combined model theoretically increases the search space as there are more variables, but it is easy to realize that instantiation of variables in one model is enough to solve the problem as the other variables are instantiated via inference through the channeling constraints. The reason for combining constraint models is mutual exploitation of inference in both models. The effect is similar to adding so called implied constraints that are redundant for the description of the problem but help the inference mechanism to prune more the search space.

As mentioned above, it is enough to instantiate variables in one model to obtain a solution. It may be also possible to use both sets of variables as the search variables, especially, if all variables are of the same type and dynamic variable ordering is used. Hnich et al. (2004) compared such search strategies for permutation problems but we are not aware about any general study which variables in combined models should participate in search.

In this paper we would like to open the question where to search in combined constraint models for planning problems. We will first describe how planning problems are modeled using constraints, then we sketch our combined constraint model, and finally we discuss the possible search strategies for such a combined model.

Constraint Models in Planning

Recall first that classical planning deals with finding a (shortest) sequence of actions transferring the world from its initial state to a state satisfying the goal condition. Traditional planning systems explore either paths in the state space (state-space planning) or partial plans (plan-space planning). Formulating the planning problem as a CSP is not a new idea. There exist handcrafted models ala CPlan (van Beek and Chen, 1999) and general models GP-CSP (Do and Kambhampati, 2000) and CSP-PLAN (Lopez and Bacchus, 2003) based on the structure of planning graph. In (Barták and Toropila, 2008) we reformulated GP-CSP and CSP-PLAN for multi-valued state variables and sequential planning. Recent addition to sequential constraint-based planners is FDT (Grandcolas and Pain-Barre, 2007). All these approaches share the idea of modeling the planning problem as a sequence of CSPs, where each CSP encodes the problem of finding a plan of length *i* and *i* incrementally increases until the plan is found. Each such CSP consists of a set of action variables modeling the actions in the plan and state variables describing the values of the state variables in each state. These variables are connected via constraints modeling the state transitions by applying the actions (Figure 1). The models differ in using Boolean or multi-valued state variables and in the type of constraints describing the state transitions. In all these approaches, only the action variables participate in the search decisions. In our approach we use regression (backward) planning while FDT does forward planning. In both cases, the variable ordering is fixed.

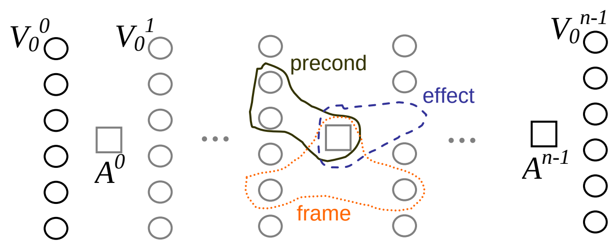

Figure 1. Base decision variables and constraints modeling plans.

CPT (Vidal and Geffner, 2004) is a representative of constraint-based Partial Order Causal Link planner. It uses existence and temporal variable for each possible action and the constraints ensure that for each actions that "exists" in the plan the actions giving its precondition also exist in the plan at the right (temporal) position. The search is based on resolving flaws, that is, finding a support for open preconditions (instantiating existence variables) and resolving causal threads by adding temporal ordering constraints. The disadvantage of this model is that it does not allow repetition of actions in the plan so it resembles more oversubscribed scheduling than general planning. Moreover, it does not contain explicit representation of states which disqualifies using state analysis techniques.

In (Barták and Toropila, 2009) we suggested an extension of the constraint-based sequential planner by ideas from partial order causal link planning to strengthen the inference. In particular, when the action variable is instantiated, we introduce the so called support action and level variables for each precondition of the action. These variables describe which action gives that precondition (a support action) and in which position in the plan this support action appears (a support level). All support action variables are connected using *nvalue* constraint ensuring that there are enough positions in the plan to contain all support actions. The channeling constraint is realized via *element* constraint which basically connects the support level and support action variables with the action variables (the support level describes the index in the sequence of the action variables where the support action is located). The dual model plays the role of redundant constraints only and search is done in the original model. The preliminary experiments showed that the overhead of the dual model is paid-off by the decrease of the search space and more planning problems can be solved.

Where to Search in Planning?

Though we sketched the search strategy for each constraint model, the open question is whether by extending the search to the dual model improves performance of the planner. The results in (Hnich et al., 2004) showed that it can be beneficial to search in both models. The difficulty of our approach is that the dual model is introduced incrementally when instantiating the action variables. Nevertheless, the search strategy from CPT can be applied to existing support action and support level variables which corresponds to instantiating existence and temporal variables in CPT. We are currently exploring this direction of research. Note also that so far we ignored the state variables that participate in the inference only. So another open question is whether the instantiation of the state variables during search is beneficial. This introduces another complexity to the search procedure as it needs to handle variables of three different types: actions, positions, and states.

Acknowledgments

The research is supported by the Czech Science Foundation under the projects 201/07/0205 and 201/09/H057.

References

Barták, R., Toropila D. 2008. Reformulating Constraint Models for Classical Planning. In: *21st International Florida AI Research Society Conference* (FLAIRS 2008), AAAI Press, 525-530

Barták, R., Toropila, D. 2009. Integrated Constraint Models for Sequential and Partial-Order Planning. To appear in SARA 2009.

Do, M.B. and Kambhampati, S. 2000. Solving planning-graph by compiling it into CSP. *Proceedings of the Fifth International Conference on Artificial Planning and Scheduling* (AIPS-2000), AAAI Press, 82-91.

Grandcolas, S., and Pain-Barre, C. 2007. Filtering, decomposition and search space reduction for optimal sequential planning. *Proceedings of the Twenty-Second Conference on Artificial Intelligence* (AAAI-07), 993–998.

Hnich, B., Smith, B.M, and Walsh, T. 2004. Dual Models of Permutation and Injection Problems. *Journal of Artificial Intelligence Research*, 21:357-391.

Lopez, A. and Bacchus, F. 2003. Generalizing GraphPlan by Formulating Planning as a CSP. *Proceedings of IJCAI*, 954-960.

van Beek, P. and Chen, X. 1999. CPlan: A Constraint Programming Approach to Planning. *Proceedings of AAAI-99*, 585-590.

Vidal, V. and Geffner, H. 2004. Branching and Pruning: An Optimal Temporal POCL Planner based on Constraint Programming. *Proceedings of AAAI-04*, 570-577.