MBP: a Model Based Planner

Piergiorgio Bertoli[1]**, Alessandro Cimatti**[1]**, Marco Pistore**[1,2]**, Marco Roveri**[1]**, Paolo Traverso**[1]
[1] ITC-IRST, Via Sommarive 18, 38055 Povo, Trento, Italy
[2] DIT, University of Trento, Via Sommarive 14, 38050 Povo, Italy
{bertoli,cimatti,roveri,traverso}@irst.itc.it pistore@dit.unitn.it

Abstract

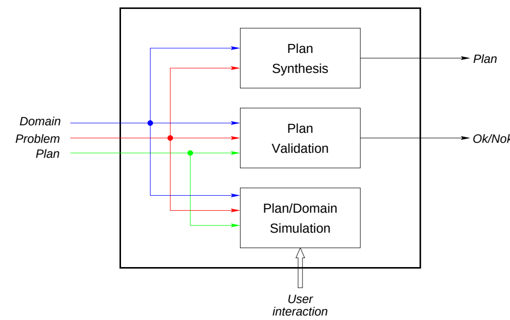
The Model Based Planner (MBP) is a system for planning in non-deterministic domains. It can generate plans automatically to solve various planning problems, like conformant planning, planning under partial observability, and planning for temporally extended goals. Moreover, MBP can validate plans, and offers a variety of simulation functionalities for plans and domains. MBP is based on Symbolic Model Checking techniques, and Binary Decision Diagrams (BDDs), that provide a practical solution to the problem of dealing with the large size of realistic planning problems. Experimental analysis in the course of the last years has shown MBP to be state-of-the-art in planning for nondeterministic domains. The demo aims at showing MBP's array of functionalities for plan generation, validation and simulation over an increasingly complex navigation problem.

Introduction

The Model Based Planner (MBP), available at `http://sra.itc.it/tools/mbp/`, is a system designed to do planning in non-deterministic domains. It provides:

- a general framework for dealing with different classes of planning problems in non-deterministic domains.
- planning algorithms that can deal effectively with large state spaces.
- plan validation and simulation functionalities to support the user in correctly modeling domains and plans.

MBP relies on a simple but general model of non-determinism, which encompasses uncertainty on the initial situation, on the action effects, and partial run-time observability of the domain state. In the model, domains are viewed as non-deterministic finite-state machines (FSMs), and plans as deterministic FSMs. MBP exploits the underlying symbolic model-checking machinery of the NuSMV tool to efficiently represent and handle large size FSMs. Given a non-deterministic planning domain, MBP can plan for different kinds of problems. Intuitively, these can be classified according to two dimensions. The first is the degree of observability, i.e. to which extent is the state of the domain observable at run-time: fully, not at all, partially. The second is the expressiveness of the goals. Planning problems can range from goal reachability (with different guarantees of achievement), to the more general case of *temporally extended goals*, i.e., where goals express conditions on whole sequences of states resulting from the execution of a plan.



MBP can tackle a wide spectrum of problems in this space: planning for temporally extended goals under conditions of total observability, reachability under every possible assumption on observability. Depending on the kind of problem being tackled, MBP uses specialized plan generation algorithms to generate plans of different form: sequential plans, conditional plans, iterative trial-and-error strategies, and plans that take into account the previous execution history. MBP represents generated plans by a single, powerful and user-friendly plan language. To complement its plan generation capabilities, MBP offers the possibility to validate a plan against a domain and a goal, and to simulate the behaviour of plans over domains, or to simulate domains in isolation. The figure represents MBP's set of functionalities. In the following, we detail the plan generation, validation and simulation capabilities of the system, and then describe the structure and contents of the ICAPS03 demo.

Plan generation

Reachability planning

In nondeterministic domains, a plan for a reachability goal is associated with many possible executions. This leads to classifying solutions as either strong, weak, or strong cyclic. A weak plan is required to admit some executions that achieve the goal; for a strong plan, all possible executions must achieve the goal. A strong cyclic plan implements a trial-and-error, such that all the associated executions always have a possibility of terminating and, when they do, they achieve the goal.

MBP's planning capabilities cover all the above cases in the case of full observability, implementing a variety of algorithms that rely on different forms of fixed-point computations over sets of states (Cimatti *et al.* 1998a; 2001). In all cases, MBP generates iterative and conditional plans that repeatedly sense the world, select an appropriate action, execute it, and iterate until the goal is reached.

For partial or null observability, MBP provides algorithms for generating strong plans. In both cases, since the knowledge about the run-time state is incomplete, search must be carried out in the space of belief states, i.e. the powerset of the space of states of the planning domain.

In the case of null observability, also called conformant planning, the search results in a plan consisting of a sequence of actions guaranteed to achieve the goal. MBP can carry out conformant plan generation either by a breadth-first search where the search frontier is represented symbolically (Cimatti and Roveri 2000), or by integrating the symbolic model checking techniques of the first approach with a heuristic-style search (Bertoli *et al.* 2001a).

For handling partial observability, MBP allows specifying *observation variables* that represent what is observable in the planning domain. In this case, MBP generates a tree-shaped plan that represent a conditional course of actions, where branching depends on the value of observation variables. MBP provides planning algorithms that perform a forward exploration of the and-or search space of belief states, induced by the domain (Bertoli *et al.* 2001b).

Planning for Temporally Extended Goals

MBP may plan for temporally extended goals, under the assumption of full observability of the domain. Temporally extended goals express conditions on the whole executions associated to the solution plan (rather than just on their final states). MBP allows for extended goals formulated as CTL formulae. CTL takes into account non-deterministic action outcomes; as such, it allows for requirements on either "all the possible executions" or "some executions" of a plan. MBP also allows for extended goals using EaGle, a goal language designed to also express intentionality.

In the general case of extended goals, the generated plans may have to execute different actions in a state, depending on the previous execution history. MBP generates plans that satisfy this requirement.

The planning algorithms for CTL and EaGLe (Pistore and Traverso 2001) rely on building control automata that corresponding to the extended goals, and using them to guide the search for a solution.

Reactive Planning

MBP allows for interleaving planning and execution for reachability in partially observable domains. The extension allow tackling large-size problems where dealing off-line with the huge number of contingencies is practically unfeasible. Several issues must be considered when performing interleaving, e.g. which is the balance between off-line planning and execution, how to avoid infinite plan-execution loops, how to deal with run-time failures. MBP's algorithm is guaranteed to terminate, either with a strong plan, or when no strong plan may ever be found from the belief state reached by execution.

Plan validation/simulation

MBP offers the possibility of validating plans, either produced by the planner or provided by the user in MBP's plan language, against a domain and a goal. Plan validation exploits the underlying model-checking machinery and results in validating the goal property agains a suitable combination of the FSMs describing the plan and the domain.

Simulating a plan over a domain, or the behaviour of a domain in isolation, is vital to allow users to effectively devise correct plans and domain descriptions. MBP offers a variety of simulation modes, where e.g. the choice of non-deterministic action outcomes is left to the user, or to the environment.

Demo

The demo at ICAPS03 aims at motivating and showing each of the functionalities above. For a more linear presentation, the demo focuses on a unique robot navigation planning domain, considered under several variants (e.g. with moving obstacles, unpredictable doors separating rooms, etc.). We will present a set of increasingly complex navigation problems, considering increasing complications in the domain. For each problem, we will generate one or more plans by exploiting MBP's algorithms. The plans will be compared by running pairs of parallel simulations. This will highlight the difference in behaviour between plans obtained for different goal classes, and under different observability assumptions. Finally, we will confront with plan generation for a complex problem instance, both via off-line and interleaved planning. This will stress the differences between the two approaches, and the advantages of the interleaved approach.

References

P. Bertoli, A. Cimatti, and M. Roveri. Heuristic Search + Symbolic Model Checking = Efficient Conformant Planning. In *Proc. IJCAI-01*. AAAI Press, August 2001.

P. Bertoli, A. Cimatti, M. Roveri, and P. Traverso. Planning in Nondeterministic Domains under Partial Observability via Symbolic Model Checking. In *Proc. IJCAI-01*. AAAI Press, August 2001.

A. Cimatti and M. Roveri. Conformant Planning via Symbolic Model Checking. *Journal of Artificial Intelligence Research (JAIR)*, 13:305–338, 2000.

A. Cimatti, M. Roveri, and P. Traverso. Automatic OBDD-based Generation of Universal Plans in Non-Deterministic Domains. In *Proceeding of the AAAI-98*, Madison, Wisconsin, 1998. AAAI-Press.

A. Cimatti, M. Roveri, and P. Traverso. Strong Planning in Non-Deterministic Domains via Model Checking. In *Proceeding of AIPS-98*, Carnegie Mellon University, Pittsburgh, USA, June 1998. AAAI-Press.

A. Cimatti, M. Pistore, M. Roveri, and P. Traverso. Weak, Strong, and Strong Cyclic Planning via Symbolic Model Checking. Technical report, IRST, Trento, Italy, 2001.

M. Pistore and P. Traverso. Planning as Model Checking for Extended Goals in Non-deterministic Domains. In *Proc. IJCAI-01*. AAAI Press, August 2001.