

UNIVERSITAT DE GIRONA

DEPARTAMENT D'INFORMÀTICA, MATEMÀTICA APLICADA I ESTADÍSTICA  
UNIVERSITAT DE GIRONA, SPAIN

---

**Dissertation Abstract: SAT/SMT techniques for planning problems**

---

*Author:*  
Joan ESPASA

*Supervisors:*  
Mateu VILLARET

February 13, 2015

## Abstract

Although a lot of work has been devoted to the encoding of planning tasks to propositional logic, only a few works can be found in the literature on satisfiability based approaches to planning in domains that require numeric reasoning. This is probably due to the difficulty of efficiently handling at the same time numeric constraints and propositional formulas. Surprisingly, satisfiability modulo theories (SMT) has been scarcely considered in planning, despite being an active and growing area of research. Since SMT is the natural extension of SAT when propositional formulas need to be combined with numeric constraints, we think it is worth considering SMT for SAT-based planning with numeric domains. The purpose of this thesis is to adapt and take advantage of the expressivity of SMT technology for solving planning problems with numerical constraints.

## Introduction

The problem of planning, in its most basic form, consists in finding a sequence of actions that will allow to reach a goal state from a given initial state. Although initially considered a deduction problem, it was rapidly seen that it could be better addressed by looking at it as a satisfiability (model finding) problem (Kautz and Selman 1992). Many (incomplete) heuristic methods can be found in the literature to efficiently deal with this problem, most of them oriented towards finding models. Exact methods were ruled out at the beginning due to their inefficiency. However, in (Kautz, McAllester, and Selman 1996) it was shown that modern off-the-shelf SAT solvers could be effectively used to solve planning problems. In the last years, the power of SAT technology has been leveraged to planning (Rintanen 2012), making reduction into SAT state of the art for deterministic planning.

As we have stated, a lot of work has been devoted to the encoding of plans in propositional logic, but only a few works can be found in the literature on satisfiability based approaches to planning in domains that require numeric reasoning. However, the advances in satisfiability modulo theories (SMT) (Barrett et al. 2009) in the last years make worth considering this alternative.

The pioneering work of LPSAT (Wolfman and Weld 1999) on planning with resources can indeed be considered one of the precursors of SMT, as the basic ideas of SMT (Boolean abstraction, interaction of a SAT solver with a theory solver, etc.) were already present in it. A comparison between SAT and SMT based encodings for planning in numeric domains can be found in (Hoffmann et al. 2007).

Other approaches, related to SMT to some amount as well, have been developed more recently. In (Belouaer and Maris 2012), a set of encoding rules is defined for spatio-temporal planning, taking SMT as the target formalism. On the other hand, in (Gregory et al. 2012) a modular framework, inspired in the architecture of lazy SMT, is developed for planning with resources.

## Hypothesis

Methods and techniques for SAT have been adapted and extended successfully for dealing with problems modelled us-

ing more expressive logics than propositional. In the case of SAT Modulo Theories (SMT), its objective is to decide the satisfiability of a set of propositional formulas, in combination of theories like equality, linear or real integer arithmetic, bit vectors, . . . The application of SMT technology to combinatorial problems has given very good results. The purpose of this thesis is to adapt and take advantage of the expressivity of SMT technology for solving planning problems with numerical constraints. This objective can be seen as a two step plan.

## Efficient Encodings

The first step we have tackled is to efficiently encode PDDL problems into SMT problems.

As SMT is an expressive language, a first translation of planning problems expressed with PDDL was achieved, which resulted in a publication (Bofill, Espasa, and Villaret ) where it was shown that it has promising results.

Our approximation is competitive with other exact and complete methods for planning with resources on the tackled problem, and also with some incomplete (heuristic) ones. In particular, we obtained better results than NumReach (Hoffmann et al. 2007) and similar results to SGPlan (Hsu and Wah 2008).

But some of the instances were big or long enough to make this approach not feasible. As the number of variables, and hence the search space, rapidly grows with the number of time steps considered, a key idea to improve the performance of this approach is to consider the possibility of executing several actions at the same time, i.e., the notion of parallel plans. Parallel plans increase the efficiency not only because they allow to reduce the time horizon, but also because it is unnecessary to consider all total orderings of the actions that are performed in parallel.

Parallelization of actions relies in the notion of (non-)interference, which is usually determined at compile time, i.e., independently of any state. This method often overestimates the chances of interference, but guarantees the feasibility of the plan.

To ensure that a parallel plan is sound, it is necessary that all actions proposed to be executed at the same time do not interfere. Different notions of interference have been defined, some more restrictive, some more relaxed.

A generalization of the standard notion of interference between actions in SAT-based planning to the case of formulas over Boolean and linear arithmetic propositions, makes it suitable for planning with resources. In particular, we are developing novel ideas that can help to determine in a more fine-grained way interference between actions, as we think that it is a key ingredient for dealing with planning with resources efficiently.

To illustrate the situations where classic notions of interference (Fox and Long 2003) overestimate affectation between actions, consider the following example. The problem consists in transporting people between cities using cars. Each car has a limited number of seats and a given fuel capacity. The actions on this example are `drive` and `board`.

A car can only drive if it is transporting somebody and

it has enough fuel to reach its destination, and boarding is limited by seat availability:

```
(:action fly
:parameters (?a - aircraft ?c1 ?c2 - city)
:precondition (and (at ?a ?c1)
(> (onboard ?a) 0)
(>= (fuel ?a)
(distance ?c1 ?c2)))
:effect (and (not (at ?a ?c1))
(at ?a ?c2)
(decrease (fuel ?a) (distance ?c1 ?c2)))

(:action board
:parameters (?p - person ?a - aircraft ?c - city)
:precondition (and (at ?p ?c)
(at ?a ?c)
(> (seats ?a) (onboard ?a)))
:effect (and (not (at ?p ?c))
(in ?p ?a)
(increase (onboard ?a) 1)))
```

The classic notion of interference would determine interference between drive and board, since board modifies the onboard function (number of passengers) and drive checks the value of this function in its precondition. We are developing techniques that can find out that there is no interference at all, since it is impossible that the preconditions of board and drive were true at the same time, and after executing board the precondition of drive becomes false. Note that the precondition of drive requires  $(> \text{ (onboard ?a) } 0)$  and the effect  $(\text{increase (onboard ?a) } 1)$  of board can never falsify  $(> \text{ (onboard ?a) } 0)$ .

As the previously introduced encodings grow considerably with time, to the point of getting unmanageable instances in some big domains, we are developing a more compact encoding, using the theory of uninterpreted functions to express predicates, functions and actions. This encoding is reminiscent of the lifted causal encodings in (Kautz, McAllester, and Selman 1996).

In the SMT-LIB standard (Barrett, Stump, and Tinelli 2010), QF\_UFLIA stands for the logic of Quantifier-Free Boolean formulas, with Linear Integer Arithmetic constraints and Uninterpreted Functions. Uninterpreted functions have no other property than its name and arity. In other words, they are only subject to the following axiom schema of consistency:  $x_1 = x'_1 \wedge \dots \wedge x_n = x'_n \implies f(x_1, \dots, x_n) = f(x'_1, \dots, x'_n)$ .

Every defined object (ship, port, cargo, ...) in the problem is mapped to an integer. For each function, predicate and action an uninterpreted function is declared. Uninterpreted functions corresponding to predicates and actions return a Boolean value, whilst the ones for functions return an integer value.

We observed promising performance using this sequential encoding with uninterpreted functions. This encoding is more compact, and it retains most of the problem original structure. It remains to be seen if a parallelized version of this encoding could lead to better results than the encoding without functions. To the best of our knowledge, there are no works using parallelized encodings with uninterpreted functions. It should hence be studied how to generalize the standard parallel encodings to this setting.

## Adapting SMT solvers

To this day, existing works to solve combinatorial problems using SMT are based in using SMT solvers as black boxes. This approximation has obvious limits, as much is left to the internal design decisions of the solver strategies. We aim to adapt SMT solvers for efficiently modelled planning problems, helping the solver search strategies and even adapting existing theories to our benefit. There is a lot of room for improvement in this area, and we expect to be able to continue doing meaningful contributions to the community.

## References

- Barrett, C.; Sebastiani, R.; Seshia, S.; and Tinelli, C. 2009. Satisfiability Modulo Theories. In *Handbook of Satisfiability*, volume 185. IOS Press. chapter 26, 825–885.
- Barrett, C.; Stump, A.; and Tinelli, C. 2010. The Satisfiability Modulo Theories Library (SMT-LIB). <http://www.SMT-LIB.org>.
- Belouaer, L., and Maris, F. 2012. SMT Spatio-Temporal Planning. In *ICAPS 2012 Workshop on Constraint Satisfaction Techniques for Planning and Scheduling Problems (COPLAS 2012)*, 6–15.
- Bofill, M.; Espasa, J.; and Villaret, M. Efficient smt encodings for the petrobras domain.
- Fox, M., and Long, D. 2003. Pddl2. 1: An extension to pddl for expressing temporal planning domains. *J. Artif. Intell. Res.(JAIR)* 20:61–124.
- Gregory, P.; Long, D.; Fox, M.; and Beck, J. C. 2012. Planning Modulo Theories: Extending the Planning Paradigm. In *Twenty-Second International Conference on Automated Planning and Scheduling (ICAPS 2012)*. AAAI.
- Hoffmann, J.; Gomes, C. P.; Selman, B.; and Kautz, H. A. 2007. SAT Encodings of State-Space Reachability Problems in Numeric Domains. In *20th International Joint Conference on Artificial Intelligence (IJCAI 2007)*, 1918–1923.
- Hsu, C.-W., and Wah, B. W. 2008. The SGPlan Planning System in IPC-6. <http://wah.cse.cuhk.edu.hk/wah/Wah/papers/C168/C168.pdf>.
- Kautz, H., and Selman, B. 1992. Planning as Satisfiability. In *10th European Conference on Artificial Intelligence (ECAI 92)*, 359–363. John Wiley & Sons, Inc.
- Kautz, H. A.; McAllester, D. A.; and Selman, B. 1996. Encoding Plans in Propositional Logic. In *Fifth International Conference on Principles of Knowledge Representation and Reasoning (KR 96)*, 374–384. Morgan Kaufmann.
- Rintanen, J. 2012. Planning as Satisfiability: Heuristics. *Artificial Intelligence* 193:45–86.
- Wolfman, S. A., and Weld, D. S. 1999. The LPSAT Engine & Its Application to Resource Planning. In *Sixteenth International Joint Conference on Artificial Intelligence (IJCAI 99)*, 310–317. Morgan Kaufmann.