Planning as Satisfiability for Cyber-Physical Systems

Francesco Leofante*

University of Genoa, Genoa, Italy
RWTH Aachen University, Aachen, Germany

Abstract. Planning as Satisfiability is one of the most well-known and effective techniques for classical planning. The basic idea is to encode the existence of a plan with n steps as a propositional satisfiability formula obtained by unfolding, $n + 1$ times, the symbolic transition relation of the automaton described by the planning problem. Planning for Cyber-Physical Systems, however, requires languages that are more expressive than propositional logic to model, *e.g.*, energy consumption, time durations. We study how first-order (arithmetic) theories can be used to this end, and propose to leverage recent advances in Satisfiability Modulo Theories solving to compute optimal plans for complex systems that require both propositional and numeric reasoning.

1 Introduction

In planning, the objective is to find a sequence of actions that leads a system from a given initial state to a goal state. As shown by Kautz and Selman for the first time in [6], classical planning problems can be naturally formulated as propositional satisfiability problems and solved efficiently by SAT solvers. The idea is to encode the existence of a plan of a fixed length n as the satisfiability of a propositional logic formula: the formula for a given n is satisfiable if and only if there is a plan of length n leading from the initial state to the goal state, and a model for the formula represents such plan.

Classical planning abstracts aways from numeric quantities, however these are of paramount importance when dealing with a Cyber-Physical System (CPS). Consider a robot managing orders in a *smart warehouse*. Once a new order request arrives, the robot would take the order, fetch the requested product and prepare it for delivery. With an increasing number of orders, large teams of robots would be needed to keep the business running. Each robot in the team would have to come up with an efficient plan to deliver its order, all while considering what other robots do so as to avoid interferences. On top of this, optimality targets such as minimizing overall energy consumption or time to delivery, should be taken into account to ensure efficiency.

The natural encoding of planning problems for domains like the one we just introduced requires an extension of propositional logic with arithmetic theories,

* Joint work with Erika Ábrahám and Armando Tacchella.

such as the theory of reals or integers. Advances in satisfiability checking led to powerful *Satisfiability Modulo Theories* (*SMT*) solvers such as [3, 12], which can be used to check the satisfiability of first-order logic formulas expressed over arithmetic theories and thus model numeric quantities such as the ones mentioned above.

For the synthesis of *optimal plans* we need to go beyond satisfiability and resort to optimization: an optimal plan is a satisfying solution for the logical formula encoding the planning problem, which ensures optimality of a desired objective function that defines a relevant cost metric. The importance of solving such optimization problems has been recognized by the SMT community [2, 17, 19] and led to the emergence of a new field called *Optimization Modulo Theories* (*OMT*) and the development of efficient OMT solvers [1, 21].

In our work we intend to leverage recent advances in satisfiability checking to extend the original Planning as Satisfiability framework to enable optimization over reward structures expressed in first-order arithmetic theories. More specifically, we propose to reduce optimal numeric planning to OMT: combining symbolic reachability techniques and optimization, OMT solvers can be leveraged to generate optimal plans for complex systems that require both propositional and numeric reasoning.

In the following we discuss our experience using OMT decision procedures for planning. After briefly presenting the preliminaries on which this work builds, we present our results on optimal planning in a *smart logistic* domain involving multi-robot systems. We then briefly sketch our ongoing work and then conclude with future directions we intend to explore to further the development of planning as OMT.

2 Preliminaries

2.1 Satisfiability Modulo Theories and optimization

Satisfiability Modulo Theories (SMT) is the problem of deciding the satisfiability of a first-order formula with respect to some decidable theory $\mathcal{T}$. In particular, SMT generalizes Boolean satisfiability (SAT) [4] by adding background theories such as the theory of real numbers, integers, and the theories of data structures.

To decide the satisfiability of an input formula φ in CNF, SMT solvers such as [12, 3] typically proceed as follows. First a *Boolean abstraction* $abs(\varphi)$ of φ is built by replacing each constraint by a fresh Boolean proposition. A SAT solver searches for a satisfying assignment S for $abs(\varphi)$. If no such assignment exists then the input formula φ is unsatisfiable. Otherwise, the consistency of the assignment in the underlying theory is checked by a *theory solver*. If the constraints are consistent then a satisfying solution (*model*) is found for φ. Otherwise, the theory solver returns a theory lemma φ_E giving an *explanation* for the conflict, *e.g.*, the negated conjunction of some inconsistent input constraints. The explanation is used to refine the Boolean abstraction $abs(\varphi)$ to $abs(\varphi) \wedge abs(\varphi_E)$. These steps are iteratively executed until either a theory-consistent Boolean assignment is found, or no more Boolean satisfying assignments exist.

Standard decision procedures for SMT have been extended with optimization capabilities, leading to Optimization Modulo Theories (OMT). OMT extends SMT solving with optimization procedures to find a variable assignment that defines an optimal value for an objective function f (or a combination of multiple objective functions) under all models of a formula φ. As noted in [20], OMT solvers such as [21, 1] typically implement a *linear search* scheme, which can be summarized as follows. Let φ_S be the conjunction of all theory constraints that are true under S and the negation of those that are false under S. A local optimum μ for f is computed[1] under the side condition φ_S and φ is updated as

$$\varphi := \varphi \wedge (f \bowtie \mu) \wedge \neg \bigwedge \varphi_S \quad , \quad \bowtie \in \{<, >\}$$

Repeating this procedure until the formula becomes unsatisfiable will lead to an assignment minimizing f under all models of φ.

2.2 Planning Modulo Theories

Planning as Satisfiability frames the existence of a plan of a fixed length p as the satisfiability of a propositional logic formula: the formula for a given p is satisfiable if and only if there is a plan of length p leading from the initial state to the goal state, and a model for the formula represents such plan. Standard reductions of classical planning to SAT abstract away from numeric quantities, however this is not the case for SMT. More precisely, Planning Modulo Theories can be formalized as follows.

World states are described using an ordered set of real-valued *variables* $x = \{x_1, \ldots, x_n\}$. We also use the vector notation $x = (x_1, \ldots, x_n)$ and write x' and x_i for $(x'_1, \ldots, x'_n)$ and $(x_{1,i}, \ldots, x_{n,i})$ respectively. We use special variables $A \in x$ to encode the *action* to be executed at each step and $t \in x$ for the associated time stamp. A *state* $s = (v_1, \ldots, v_n) \in \mathbb{R}^n$ specifies a real value $v_i \in \mathbb{R}$ for each variable $x_i \in x$.

The planning domain can then be represented symbolically by mixed-integer arithmetic formulas defining the *initial states* $I(x)$, the *transition relation* $T(x, x')$ (where x describes the state before the transition and x' the state after it) and a set of *final* states $F(x)$. The transition relation is defined in terms of actions that can be performed at each step. A *plan* of length p is a sequence $s_0, \ldots, s_p$ of states such that $I(s_0)$ and $T(s_i, s_{i+1})$ hold for all $i = 0, \ldots, p-1$, and $F(s_p)$ holds. Thus, plans are models for the formula:

$$I(x_0) \wedge \left(\bigwedge_{0 \le i < p} T(x_i, x_{i+1}) \right) \wedge \left(\bigvee_{0 \le i \le p} F(x_i) \right) \tag{1}$$

In general the length of a plan is not known a priori and has to be determined empirically by increasing p until a satisfying assignment for Eq. 1 is found, or an upper bound on p is reached. In order to be able to support generation of

[1] For instance, if f and φ_S are expressed in `QF_LRA`, this can be done with Simplex

optimal plans with OMT, the bounded planning approach needs to be extended to enable optimization over cost structures expressed in first-order arithmetic theories. We introduce additional variables $c \in x$ to encode the *cost* of executing action A at time t. We define the total cost c_{tot} associated to a plan as:

$$c_{tot} = \sum_{0 \leq i < p} c_i \tag{2}$$

Optimal bounded planning is then defined as the problem to find a path of length at most p that reaches a target state and achieves thereby the smallest possible cost, *i.e.*, to minimize Eq. 2 under the side condition that Eq. 1 holds.

3 Current results: optimal planning for smart logistics

In a recent series of papers [7–11, 16] we proposed OMT as an approach to deliver task plans that can meet production requirements (optimally) and withstand deployment in the RoboCup Logistics League (RCLL). While the approach described in these papers is domain-specific, we expect that our solution can carry over to domains with similar structure and features, thus providing the basis for general, yet efficient, synthesis of optimal plans based on OMT.

3.1 The RoboCup Logistics League

The RoboCup Logistics League provides a simplified smart factory scenario where two teams of three autonomous robots each compete to handle the logistics of materials to accommodate orders known only at run-time. Competitions take place yearly using a real robotic setup. However, for our experiments we made use of the simulated environment (Fig. 1) developed for the Planning and Execution Competition for Logistics Robots in Simulation[2] [13].

Products to be assembled have different complexities and usually require a base, mounting 0 to 3 rings, and a cap as a finishing touch. Bases are available in three different colors, four colors are admissible for rings and two for caps, leading to about 250 different possible combinations. Each order defines which colors are to be used, together with an ordering. An example of a possible configuration is shown in Fig. 1.

Several machines are scattered around the factory shop floor (random placement in each game, positions are announced to the robots). Each of them completes a particular production step such as providing bases (Base Station, BS), mounting colored rings (Ring Station, RS) or caps (Cap Station, CS). The objective for autonomous robots is then to transport intermediate products between processing machines and optimize a multistage production cycle of different product variants until delivery of final products.

Orders that denote the products which must be assembled are posted at run-time by an automated referee box and come with a delivery time window,

[2] http://www.robocup-logistics.org/sim-comp

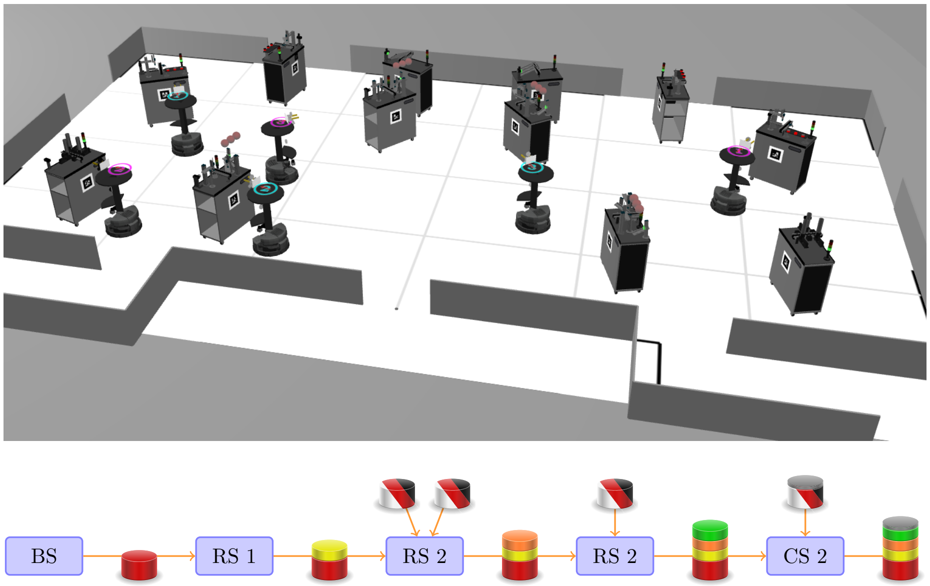

Fig. 1: Simulated RCLL factory environment [13] (top) and example of order configuration for the competition [15, 18] (bottom).

introducing a temporal component that requires quick planning and scheduling – see [11] for an account on the challenges presented by the RCLL.

3.2 Our results

To generate optimal plans, we extended standard Planning as Satisfiability to enable optimization over reward structures expressed in first-order arithmetic theories in OMT – see [7] for a brief overview of our approach. This idea was applied to solve multi-robot planning problems arising in the RCLL, such as factory shop-floor exploration [9] and planning for production [10].

To cater for the dynamics that occur when plans are executed on concrete systems, we also presented a system that integrates our planning approach into an online execution agent based on CLIPS [14], currently used by the RCLL world champion. A prototypical implementation of this system was presented in [16] and later extended in [8, 10]. Our approach proved to be competitive, gaining the first place in the Planning and Execution Competition for Logistics Robots in Simulation at ICAPS'18.

4 What's next?

The solutions presented in Sec. 3 are specifically tailored for the RCLL and, although promising, do not allow for a more general comparison within the broader field of AI planning. For this reason, we are currently implementing a domain-independent OMT planner. Our planner takes as input planning tasks defined in

PDDL 3 [5], creates an OMT representation and leverages νZ [1] as a planning engine. Once completed, this planner will allow to assess performances of OMT solvers on a wider range of planning problems chosen from, *e.g.*, the International Planning Competition.[3] In addition to this, such planner will also provide a platform to test novel encodings of planning as OMT. Indeed, our experiments with different encodings of planning problems indicate that considerable progress can be made by considering novel kinds of relaxations.

References

1. Bjørner, N., Phan, A., Fleckenstein, L.: νZ - An optimizing SMT solver. In: Proc. of TACAS. pp. 194–199 (2015)
2. Cimatti, A., Franzén, A., Griggio, A., Sebastiani, R., Stenico, C.: Satisfiability modulo the theory of costs: Foundations and applications. In: Proc. of TACAS. pp. 99–113 (2010)
3. Cimatti, A., Griggio, A., Schaafsma, B.J., Sebastiani, R.: The MathSAT5 SMT solver. In: Proc. of TACAS. pp. 93–107 (2013)
4. Franco, J., Martin, J.: Handbook of Satisfiability, chap. A history of satisfiability, pp. 3–74. IOS Press (2009)
5. Gerevini, A., Haslum, P., Long, D., Saetti, A., Dimopoulos, Y.: Deterministic planning in the fifth international planning competition: PDDL3 and experimental evaluation of the planners. Artificial Intelligence 173(5-6), 619–668 (2009)
6. Kautz, H.A., Selman, B.: Planning as satisfiability. In: Proc. of ECAI. pp. 359–363 (1992)
7. Leofante, F.: Guaranteed plans for multi-robot systems via optimization modulo theories. In: Proc. of AAAI (2018)
8. Leofante, F.: Optimal multi-robot task planning: from synthesis to execution (and back). In: Proc. of IJCAI. pp. 5771–5772 (2018)
9. Leofante, F., Ábrahám, E., Niemueller, T., Lakemeyer, G., Tacchella, A.: On the synthesis of guaranteed-quality plans for robot fleets in logistics scenarios via optimization modulo theories. In: Proc of IRI. pp. 403–410 (2017)
10. Leofante, F., Ábrahám, E., Niemueller, T., Lakemeyer, G., Tacchella, A.: Integrated synthesis and execution of optimal plans for multi-robot systems in logistics. Information Systems Frontiers (2018)
11. Leofante, F., Ábrahám, E., Tacchella, A.: Task planning with OMT: an application to production logistics. In: Proc. of IFM (to appear)
12. de Moura, L.M., Bjørner, N.: Z3: An efficient SMT solver. In: Proc. of TACAS. pp. 337–340 (2008)
13. Niemueller, T., Karpas, E., Vaquero, T., Timmons, E.: Planning competition for logistics robots in simulation. In: Proc. of PlanRob@ICAPS (2016)
14. Niemueller, T., Lakemeyer, G., Ferrein, A.: Incremental Task-level Reasoning in a Competitive Factory Automation Scenario. In: Proc. of AAAI Spring Symposium on Designing Intelligent Robots: Reintegrating AI (2013)
15. Niemueller, T., Lakemeyer, G., Ferrein, A.: The RoboCup Logistics League as a benchmark for planning in robotics. In: Proc. of PlanRob@ICAPS (2015)
16. Niemueller, T., Lakemeyer, G., Leofante, F., Ábrahám, E.: Towards CLIPS-based task execution and monitoring with SMT-based decision optimization. In: Proc. of PlanRob@ICAPS (2017)

[3] http://www.icaps-conference.org/index.php/Main/Competitions

17. Nieuwenhuis, R., Oliveras, A.: On SAT modulo theories and optimization problems. In: Proc. of SAT. pp. 156–169 (2006)
18. RCLL Technical Committee: RoboCup Logistics League – Rules and regulations 2017 (2017)
19. Sebastiani, R., Tomasi, S.: Optimization in SMT with $\mathcal{LA}(\mathbb{Q})$ cost functions. In: Proc. of IJCAR. pp. 484–498 (2012)
20. Sebastiani, R., Tomasi, S.: Optimization modulo theories with linear rational costs. ACM Trans. Comput. Log. 16(2), 12:1–12:43 (2015)
21. Sebastiani, R., Trentin, P.: OptiMathSAT: A tool for optimization modulo theories. In: Proc. of CAV. pp. 447–454 (2015)