Mixed-Integer Linear Programming for Planning with Temporal Logic Tasks (Position Paper)

Vasumathi Raman
California Institute of Technology
1200 E California Blvd
Pasadena, CA 91125
vasu@caltech.edu

Eric M. Wolff
nuTonomy LLC
1 Broadway, 14th fl.
Cambridge, MA 02142
eric@nutonomy.com

Abstract

We are concerned with controlling dynamical systems, such as self-driving cars and smart buildings, in a manner that guarantees that they satisfy complex task specifications. Mixed integer linear programming has recently proven to be a powerful tool for such problems, enabling the computation of optimal plans that satisfy complex temporal constraints for high-dimensional, dynamical systems. These optimization-based approaches find solutions quickly for challenging (and previously unsolvable) planning problems. Framing temporal logic planning as constrained optimization also presents exciting new areas of research.

Motivation

The increasingly tight integration of computation and control in hybrid systems (e.g., self-driving cars, unmanned aerial vehicles, human-robot teams, the power grid, and smart buildings) leads to challenging planning problems. These problems are difficult due to the complex set of rules regulating acceptable behavior, the hybrid (discrete and continuous state) system dynamics, and the need for near-optimal plans. We propose a combination of techniques from mixed integer linear programming (MILP), model predictive control, and model checking to develop algorithms for *optimization-based temporal logic planning* that guarantee correct behavior for such systems.

Robots and other cyberphysical systems need to accomplish complex tasks such as following the rules of the road, helping a human assemble a part, or regulating the climate in a building. Temporal logics are a set of expressive languages that can specify these types of tasks and properties. Temporal logic planning generalizes classical robotic path planning by reasoning about intricate temporal sequencing of events.

Given a temporal logic task and a finite transition system, a variety of search algorithms can be applied to generate a plan guaranteed to satisfy the task. These include algorithms that leverage existing model checking tools, e.g., (Holzmann 2004; Cimatti et al. 2000), which convert the temporal logic formula into a corresponding automaton, compose it with the transition system, and search this product automaton for a solution. While such techniques are mature for discrete systems, there are significant challenges in extending them to robots and other cyberphysical systems.

A standard approach for extending the above algorithms to robotic systems is to replace the original continuous system with a finite transition system by computing a discrete abstraction (Alur et al. 2000). Unfortunately, discretizing the continuous state space limits practical use to systems with fewer than five continuous dimensions, rendering this approach infeasible for many robotic applications.

Recent results have shown that the expensive computations of a discrete abstraction and product automaton can be avoided by using MILP. By directly encoding temporal logic specifications as mixed integer constraints on the dynamical system, one can leverage the impressive empirical performance of state-of-the-art MILP solvers to solve previously unsolvable problems. Additionally, the direct encoding of temporal logic constraints nicely integrates into existing optimal control frameworks.

Related Work

Recent work in optimization-based temporal logic planning has helped connect the fields of bounded model checking and model predictive control. Optimization-based temporal logic planning extends the bounded model checking (BMC) paradigm for finite discrete systems (Biere et al. 1999), to deal with hybrid dynamical systems. The BMC approach systematically searches over fixed length discrete state sequences (plans) for one that satisfies the specification. The question of whether or not a state sequence satisfies the specification is reduced to the satisfaction of an appropriate Boolean satisfiability (SAT) instance, which can then be solved using standard tools. In this respect, these methods build on the view of planning as satisfiability (Kautz and Selman 1992), while generalizing the desired plan properties to temporal logic specifications, which may require plans that produce infinite executions.

While BMC has focused on discrete systems with complex tasks, model predictive control (Bemporad and Morari 1999) has focused on complex dynamical systems with relatively simple task specifications. While it is possible to encode temporal constraints with model predictive control, it is a difficult and error-prone process.

Using the power of MILPs to also reason about continuous variables, one can extend the SAT encoding used in

BMC to hybrid systems: the focus can now shift to dynamical systems executing complex tasks encoded in temporal logic. MILP encodings have been developed for various subsets of linear temporal logic (LTL) (Karaman and Frazzoli 2011; Kwon and Agha 2008), all of LTL (Wolff, Topcu, and Murray 2014), a restricted fragment of metric temporal logic (Karaman and Frazzoli 2008), and signal temporal logic (Raman et al. 2014). These approaches allow existing optimal control problems formulated as MILPs (see e.g. (Bemporad and Morari 1999)) to be augmented with additional constraints that enforce temporal logic specifications. These constraints would be tedious to encode manually; an automatic encoding also ensures soundness.

Previous work on search-based planning for temporal logic (Bacchus and Kabanza 1998; Patrizi et al. 2011) translates temporal goals into a classical planning framework to leverage state-of-the-art planners. Applying these techniques to cyberphysical systems requires computing a discrete abstraction, which scales poorly as discussed earlier. In contrast, we have successfully applied optimization-based synthesis to systems with over 30 continuous state variables.

Opportunities for Future Work

Framing planning with temporal logic as an optimization problem leads to interesting directions for future work. We propose the development of dedicated tools for solving optimization problems with constraints representing temporal logic formulas. Some important questions include:

1. **Automated logic-to-MILP parsing:** *Given the syntax and semantics of a language in a standard form, can one automatically extract the appropriate MILP constraints for the system?* An example of such a parser is in `LtlOpt`[1], a MATLAB tool for optimal control of high-dimensional, nonlinear systems with LTL specifications. It would be useful to be able to automatically generate a custom parser for a given domain-specific language. Beyond the soundness and completeness of such a parser, one might want to minimize the number of binary variables used, or maximize the quality of the continuous relaxations.
2. **Heuristics:** *What are natural heuristics and rounding rules for solving this class of MILPs?* Boolean satisfiability problems that come from planning have different computational characteristics from random formulas used to test SAT algorithms (Kautz and Selman 1992), and this is likely true of optimization problems that come from planning with temporal logic. A promising approach is to leverage heuristics stemming from relaxations of optimal control problems (e.g., ignoring obstacles or relaxing the dynamics) to guide the solution of the original problem.
3. **Incremental solutions:** *Can we use incremental methods to improve efficiency?* BMC techniques typically optimize over multiple fixed-length plans, so it is important to reuse as much information as possible between plans of different lengths. Warm starts, column-generation, and row-generation may have large benefits in this context.
4. **Stochastic planning:** *Can we plan in the presence of disturbances and uncertainty?* By leveraging techniques for model predictive control of stochastic systems, it may be possible to construct plans that have guaranteed performance despite exogenous disturbances.

[1]http://www.cds.caltech.edu/ ewolff/ltlopt.html

Impact

We argue that it is beneficial to view temporal logic planning as constrained optimization, instead of automata-based search. This optimization-based perspective allows simple integration with other planning, scheduling, and control frameworks, and has let us solve significantly larger problems in practice than standard techniques. While off-the-shelf solvers work reasonably well, there is still much work to be done, including automated parsing and domain-specific heuristics for faster solutions.

Acknowledgments

We thank Scott Livingston and the anonymous reviewers for their helpful comments. The first author is supported by TerraSwarm, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA.

References

Alur, R.; Henzinger, T.; Lafferriere, G.; George; and Pappas, G. J. 2000. Discrete abstractions of hybrid systems. In *Proc. of the IEEE*, 971–984.

Bacchus, F., and Kabanza, F. 1998. Planning for temporally extended goals. *Ann. Math. Artif. Intell.* 22(1-2):5–27.

Bemporad, A., and Morari, M. 1999. Control of systems integrating logic, dynamics, and constraints. *Automatica* 35(3):407–427.

Biere, A.; Cimatti, A.; Clarke, E. M.; and Zhu, Y. 1999. Symbolic model checking without BDDs. In *TACAS*, 193–207.

Cimatti, A.; Clarke, E. M.; Giunchiglia, F.; and Roveri, M. 2000. NUSMV: A new symbolic model checker. *STTT* 2(4):410–425.

Holzmann, G. J. 2004. *The SPIN Model Checker - Primer and Reference Manual*. Addison-Wesley.

Karaman, S., and Frazzoli, E. 2008. Vehicle routing problem with metric temporal logic specifications. In *Proc. of the IEEE Conf. on Decision and Control*, 3953–3958.

Karaman, S., and Frazzoli, E. 2011. Linear temporal logic vehicle routing with applications to multi-UAV mission planning. *Int. J. of Robust and Nonlinear Control* 21(12):1372–1395.

Kautz, H., and Selman, B. 1992. Planning as satisfiability. In *ECAI*, 359–363.

Kwon, Y., and Agha, G. 2008. LTLC: Linear temporal logic for control. In *HSCC*, 316–329.

Patrizi, F.; Lipovetzky, N.; De Giacomo, G.; and Geffner, H. 2011. Computing infinite plans for LTL goals using a classical planner. In *IJCAI*, 2003–2008.

Raman, V.; Maasoumy, M.; Donzé, A.; Murray, R. M.; Sangiovanni-Vincentelli, A.; and Seshia, S. 2014. Model predictive control with signal temporal logic specifications. In *Proc. of the IEEE Conf. on Decision and Control*.

Wolff, E. M.; Topcu, U.; and Murray, R. M. 2014. Optimization-based trajectory generation with linear temporal logic specifications. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 5319–5325.