Robust Motion Planning with Timed Temporal Logic Tasks via Hybrid Feedback Control

Fernando S. Barbosa*, Lars Lindemann†, Dimos V. Dimarogonas† and Jana Tumova*
*Division of Robotics, Perception and Learning (RPL), KTH Royal Institute of Technology, Stockholm, Sweden
†Division of Decision and Control Systems, KTH Royal Institute of Technology, Stockholm, Sweden.
Email: {fdsb, llindem, dimos, tumova}@kth.se

I. Introduction

In recent years, temporal logics have become a popular task and motion planning formalism, enriching the traditional A-to-B planning with more complex, structured tasks. For instance, with the use of Linear Temporal Logic (LTL), one can formalize properties such as "Visit region A, then B, while avoiding a dangerous area C". In order to allow for time constraints in the task specification, such as "Visit A within 10 to 20 time units, then visit B no earlier than 30 time units after reaching A, while avoiding C", a timed temporal logic is required, such as Metric Interval Temporal Logic (MITL). A general approach to planning under temporal logic tasks builds on finding a suitable discrete abstraction of the system dynamics, such as a graph obtained from a sampling-based motion planning method, ensuring that a transition in the abstraction can be followed via an application of a certain control law [4]. When it comes to planning under timed temporal logic tasks, finding such a discrete abstraction becomes much more challenging.

Motion planning for dynamical systems is often posed as a problem to find a sequence of control inputs that drives the system from the initial to a goal configuration while avoiding obstacles. Since a simplified version of the system dynamics is used by the planner, the resulting plan is not robust to disturbance, noise and unmodeled dynamics. For instance, in [6] a synergistic combination of high-level discrete planning with sampling-based motion planning is proposed; and even though the proposed approach is efficient when compared to other methods, it relies on the model of the system to find a sequence of constant control inputs to take it from start to goal position. On the other hand, the motion planning algorithm can assume the existence of a low-level controller, capable of driving the system from one configuration to another, and plan a sequence of configurations that drives the system from start to goal. However, such low-level controller is not known by the planner, and the overall real-time performance and robustness of the system can not be guaranteed a priori.

We propose to integrate motion planning and feedback control into one framework, and subject a single-agent system to high-level, time-dependent task specifications written as a metric temporal logic formula. Our goal is to enhance the overall robustness of the system in relation to noise, disturbance and unmodeled dynamics. Furthermore, the planner is designed to be agnostic to the nonlinear, control-affine dynamics of the system, a feature made possible by the control law proposed.

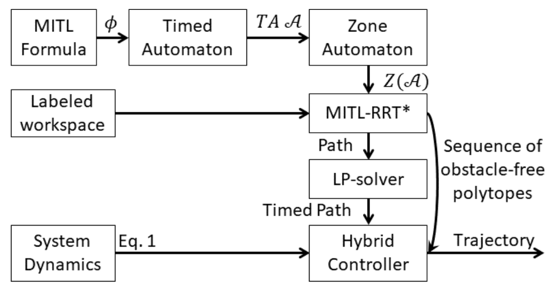


Fig. 1. The approach.

In our recent work [2] we propose an approach to find and follow a trajectory fulfilling a time-bounded MITL task that integrates sampling-based motion planning with low-level feedback controller design. A general overview is presented in Fig. 1, and can be summarized as follows: an RRT*-based algorithm is used to find an obstacle-free path – a sequence of waypoints – in the workspace that (i) satisfies the time-abstract version of the task specification (Zone Automaton - ZA), and (ii) is enclosed by a sequence of convex polytopes to be used by a Time-varying Control Barrier Function [5]. Time stamps to reach each waypoint are calculated by using clock zones of the ZA as constraints of a Linear Program (LP).

II. Problem Definition and Approach

Let $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{u} \in \mathbb{R}^m$, $\boldsymbol{d} \in \mathfrak{D} \subset \mathbb{R}^n$, where $\mathfrak{D} := \{\boldsymbol{d} \in \mathbb{R}^n | \|\boldsymbol{d}\| \leq D\}$ for some $D \geq 0$, be the state, input, and unknown disturbance, respectively, of a nonlinear system

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}) + g(\boldsymbol{x})\boldsymbol{u} + \boldsymbol{d}(t), \quad (1)$$

with locally Lipschitz continuous functions $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ such that $g(\boldsymbol{x})g(\boldsymbol{x})^T$ is positive definite for all $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{d} : \mathbb{R}_{\geq 0} \to \mathfrak{D}$ is piecewise continuous.

A *time-bounded Metric Interval Temporal Logic* (MITL) formula over a set of atomic propositions (AP) and time intervals $[a, b]$, with $a, b \in \mathbb{N}$ and $a < b$ is defined as:

$$\phi := \top \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \mathcal{U}_{[a,b]} \phi_2 \mid \mathcal{F}_{[a,b]}\phi \mid \mathcal{G}_{[a,b]}\phi, \quad (2)$$

where the proposition $p \in$ AP and $\neg$, $\wedge$, are standard negation and conjunction, and $\mathcal{U}$, $\mathcal{F}$ and $\mathcal{G}$ correspond to temporal operators until, eventually and always, respectively. We state our problem as follows:

Problem 1. *Given a system (1), a labeled workspace W partitioned into obstacles (W_{obs}) and free-space (W_{free}), and a high-level task specification ϕ in time-bounded MITL, find a control law $\boldsymbol{u}$ producing a collision-free trajectory $\mathbf{x}$ that generates a timed word $w(\mathbf{x})$ satisfying ϕ.*

A. Approach

Our approach to Problem 1 is divided into three steps. First, the time-bounded MITL task specification ϕ is translated into a Timed Automaton $\mathcal{A}$ and thereafter to its time-abstract representation Zone automaton $Z(\mathcal{A})$. The intuition here is that if there exists a path of the system that satisfies $Z(\mathcal{A})$, then there must exist a sequence of time stamps that, together with the path, will satisfy the Timed Automaton $\mathcal{A}$ and, therefore, the task ϕ. The $Z(\mathcal{A})$ is used in a sampling-based, probabilistically complete motion planning algorithm, named MITL-RRT*, and returns (i) a sequence of $|\mathbf{p}|$ waypoints that give a path satisfying an untimed version of the specification, and (ii) assumptions to be met by the low-level controller in the form of a sequence of $|\mathbf{p}|-1$ convex, obstacle-free polytopes π_i within which the trajectory of the system has to stay.

Second, we find appropriate time stamps to the sequence of waypoints so that the corresponding timed path satisfies the task specification ϕ. To that end, we exploit the structure of a zone automaton and use its clock zones as constraints of a Linear Program (LP).

Third, we design a hybrid feedback control law that, applied to system (1), tracks the timed path returned by the motion planner while staying within the obstacle-free polytopes returned by MITL-RRT*. It consists of $|\mathbf{p}|-1$ continuous-time feedback control laws $\boldsymbol{u}_i(\boldsymbol{x},t)$, a switching mechanism

$$\boldsymbol{u}(\boldsymbol{x},t) = \boldsymbol{u}_i(\boldsymbol{x},t) \text{ for } t_{i-1} \leq t < t_i \tag{3}$$

with $i \in \{1,\dots,|\mathbf{p}|-1\}$ and where each $\boldsymbol{u}_i(\boldsymbol{x},t)$ is designed based on Time-varying Control Barrier Functions formulated to solve requirements stated in Signal Temporal Logic (STL) [5]. In order to drive the system starting in x_{i-1} at t_{i-1} to x_i at t_i time units while staying inside an obstacle-free convex polytope π_i, whose H-representation is $A_i\boldsymbol{x} \leq b_i$, the corresponding STL formula ψ_i is written as

$$\psi_i = \mathcal{F}_{[t_i,t_i]}(\|\boldsymbol{x}-x_i\| \leq \epsilon) \wedge \mathcal{G}_{[t_{i-1},t_i]}(A_i\boldsymbol{x} \leq B_i), \tag{4}$$

with $B_i = b_i + \epsilon$. We expand the polytopes by ϵ to ensure the ball of radius ϵ around each x_i is inside both π_i and π_{i+1}. Due to (1) and the sequence of convex polytopes, there always exists a controller $\boldsymbol{u}_i(\boldsymbol{x},t)$ for each ψ_i.

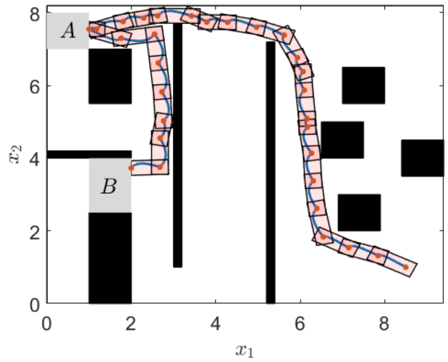
B. Results

Consider a dynamical system with coupled input given by

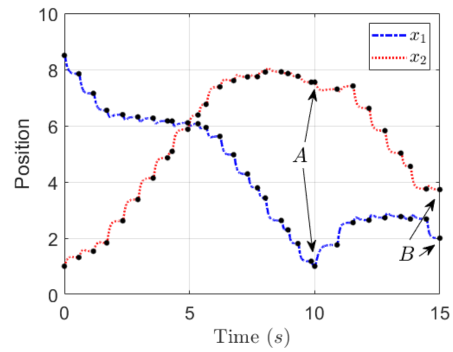
$$\dot{x}_1 = u_1 - 0.5u_2, \quad \dot{x}_2 = u_2$$

deployed in an office-like environment, with obstacles that resembles tables and walls and two goal regions A and B, that is subject to the following task specification

$$\phi_1 = \mathcal{F}_{[5,10]}A \wedge \mathcal{F}_{[15,20]}B.$$



(a)



(b)

Fig. 2. (a) presents the resulting trajectory, in blue, for $\phi_1 = \mathcal{F}_{[5,10]}A \wedge \mathcal{F}_{[15,20]}B$. The system starts from the bottom right, and the controller drives it throughout the waypoints marked in red dots without leaving the polytopes. (b) shows the system evolution in time when subject to ϕ_1. The black dots represent the projection of the timed path onto x_1 and x_2.

Fig. 2a shows the trajectory of the system on the workspace when subject to ϕ_1, with the corresponding waypoints and polytopes. Note that our approach finds a sequence of overlapping, convex, obstacle-free polytopes connecting every waypoint. Moreover, the proposed hybrid feedback control law solves an optimization problem in the attempt to minimize the control effort, and therefore two waypoints are not connected by straight lines in the resulting trajectory of the system. Fig. 2b presents the evolution of such trajectory in relation to time.

III. Conclusion and Future Work

This work is our first step into the integrating motion planning and control for high-level, time-dependent, temporal logic task specifications. We are currently working on replacing the current MITL task specification with Signal Temporal Logic in order to enable the use of its quantitative semantics into the sampling-based motion planner, as we have done in [1, 3], in which we introduce the concept of *spatial preference*, that weighs between finishing the task quickly and respecting the user-defined safety preference. Furthermore, we aim at making the framework applicable to a larger class of robotic systems, and to that end we are analyzing the possibility of (i) substituting the low-level controller by one that allows the control of systems with second-order dynamics, and (ii) extending the approach to multi-agent systems.

References

[1] F.S. Barbosa, D. Duberg, P. Jensfelt, and J. Tumova. Guiding autonomous exploration with signal temporal logic. *IEEE Robotics and Automation Letters (under review)*, 2019.

[2] F.S. Barbosa, L. Lindemann, D. Dimarogonas, and J. Tumova. Integrated motion planning and control under metric interval temporal logic specifications. In *Control Conference (ECC), 2019 European*. IEEE, 2019.

[3] J. Karlsson, F.S. Barbosa, and J. Tumova. Sampling-based motion planning with temporal logic missions and spatial preferences. In *2019 IEEE International Conference on Intelligent Robots and Systems (IROS) (under review)*, 2019.

[4] Marius Kloetzer and Calin Belta. A fully automated framework for control of linear systems from temporal logic specifications. *IEEE Transactions on Automatic Control*, 53(1):287–297, 2008.

[5] Lars Lindemann and Dimos V Dimarogonas. Control barrier functions for signal temporal logic tasks. *IEEE Control Systems Letters*, 3(1):96–101, 2019.

[6] Erion Plaku, Lydia E Kavraki, and Moshe Y Vardi. Motion planning with dynamics by a synergistic combination of layers of planning. *IEEE Transactions on Robotics*, 26(3): 469, 2010.