2016 European Control Conference (ECC)
June 29 - July 1, 2016. Aalborg, Denmark

Receding Horizon Robot Control in Partially Unknown Environments with Temporal Logic Constraints

Vladislav Nenchev and Calin Belta

Abstract— **This paper addresses the control of a mobile robot that has to accomplish a finite task in a partially unknown static environment in minimum time. The task is expressed as a syntactically co-safe Linear Temporal Logic (scLTL) formula over a set of properties that can be satisfied at the regions of a partitioned environment. The robot discovers a-priori unknown properties upon covering the corresponding region by its limited sensing range. Instead of resorting to an abstraction of the hybrid system modeling the robot's motion in the environment, we propose an approach based on parameterizing the continuous motion of the vehicle and introduce a measure of violation that is used to enforce the satisfaction of the specification. Then, we formulate a parametric Optimal Control Problem (OCP), where the objective is a convex combination of the overall time and the measure of violation function. The OCP is solved in a receding horizon manner only upon detecting previously unknown properties of the environment. The approach is illustrated with a numerical case study.**

I. Introduction

Recently, there has been a growing interest in automated robot motion planning approaches that involve solving multiple tasks or visiting several locations in a complex environment. Linear Temporal Logic (LTL) has been identified as a specification language particularly suitable for numerous robotic assignments. The majority of the works on robot control with temporal logic constraints assume an a-priori given decomposition into high-level planning and low-level control, which allows for focusing on the purely discrete path planning and verification task, e.g., [1], [2], [3], potentially including optimizing objectives [4]. Optimizing an objective for a continuous system with a discrete specification can also be addressed as a hybrid Optimal Control Problem (OCP) by employing automata-based methods, e.g., [5], [6], [7]. Alternatively, an LTL specification can be encoded directly into a mixed-integer program to obtain the time-discretized optimal control for mixed logical dynamical [8] or differentially flat nonlinear systems [9].

Many robotic assignments involve uncertainties that have to be considered in the control design process. Assuming that a robot moving in an unknown environment is represented by a non-deterministic discrete system, the control synthesis problem with an LTL specification can be generally mapped to the solution of a Rabin game [10]. If the probabilities of the feasible transitions at each state are known, the problem reduces to finding a control policy for a Markov Decision Process (MDP), such that a temporal logic formula is satisfied [11]. Solutions can also be found by adapting methods from probabilistic model checking [12], [13], where a control policy is designed to maximize the probability of satisfying the LTL specification [14]. Continuous dynamics can be taken into account by employing discrete abstractions of the underlying hybrid system, such that the OCP with a discrete high-level specification can be addressed in a discrete optimization framework under the presence of uncertainty [15], [16]. To the best of the authors' knowledge, there are no approaches that optimize the continuous motion of a robot under the presence of uncertainty, while enforcing LTL satisfaction guarantees without the use of discrete abstractions.

In this paper we address the control of a robot with continuous dynamics that has to minimize the overall time for solving a task expressed as an scLTL formula over a set of properties satisfied at the regions of a partitioned environment. The robot can determine if an a-priori unknown property can be satisfied at a region upon covering it with its limited sensing range. Building on ideas from [17], [18], [19], [20], where related hybrid OCPs were solved without employing a discrete abstraction, we propose an approach based on parameterizing the continuous motion of the vehicle. To enforce the satisfaction of the specification and to allow for efficient model checking, we introduce a measure of violation function. Thus, the addressed OCP is approximately formulated as a parametric OCP, where the objective is a convex combination of the overall time and the measure of violation function. In contrast to standard model predictive control approaches, we propose a receding horizon control scheme that does not require time-discretization and where optimization is performed only upon detecting previously unknown properties of the environment, i.e., in an event-driven manner. Even though the approach is presented for a vehicle with forth order linear dynamics, its applicability is not restricted to such systems only, since many systems can be described by double integrators upon applying feedback linearization [21] or can be transformed into a suitable form that allows convex optimization along a parameteric curve [22]. Our solution is suboptimal, but tractable and scalable in the size of the environment, the proposition set and the length of the specification formula, while guaranteeing correctness.

The remaining paper is organized as follows. In Sec. II, we introduce the addressed problem and a running example. Then, we present our solution (Sec. III) and apply it to the running example (Sec. IV). Finally, we summarize the results

V. Nenchev is with the Control Systems Group, Technische Universität Berlin, Germany. nenchev@control.tu-berlin.de

C. Belta is with the Department of Mechanical Engineering and the Division of Systems Engineering, Boston University, MA, USA. cbelta@bu.edu

This work was partially supported at Boston University by NSF under grants CMMI-1400167 and NRI-1426907.

and discuss possible extensions in Sec. V.

Notation. *For a set S, $|S|$ and 2^S denote its cardinality and the set of all of its subsets (power set), respectively. $\mathbf{0}_{m,n}$ represents an $m \times n$ matrix with zero entries. If $m = 1$, we write $\mathbf{0}_n$. $\mathbf{I}_n$ is an identity matrix with dimension n.*

II. Problem statement

In this section we introduce the OCP for a vehicle that has to minimize the time for satisfying a temporal logic assignment in a partially unknown environment. We assume that the specification is given only in terms of the position of the robot and that a solution of the problem exists. For simplicity of presentation, the problem is formulated for a vehicle that moves deterministically with forth order dynamics. We will outline how the presented methods can be applied for more complex motion models later.

A. Environment

Consider the environment $Y \subset \mathbb{R}^2$ and its regular discretization with grid points w_k denoting the center points of the regions W_k, forming a set $W = \{W_1, \ldots, W_K\}$ with $\cup_k W_k \subseteq Y, k \in \{1, \ldots, K\}$. Let Π_s denote a finite set of assignments π that can be solved at known regions $W_\pi \subset W$ of the environment (e.g. "visit the base"). Further, let Π_d denote a finite set of requests π that can be serviced at a-priori unknown regions $W_\pi \subset W$ (e.g. "assist a person in danger" in a rescue assignment or "avoid", if it is occupied by an obstacle). Thus, $\Pi_s \cup \Pi_d$ will be referred to as the set of atomic propositions in the standard "logic jargon".

B. Vehicle model

Consider a vehicle with position $y \in Y$, velocity $v \in \mathbb{R}^2$ and state $x = [y\ v]^T \in X$ that evolves with continuous dynamics

$$\dot{x}(t) = f(x,u) = \begin{bmatrix} \mathbf{0}_{2,2} & \mathbf{I}_2 \\ \mathbf{0}_{2,2} & \mathbf{0}_{2,2} \end{bmatrix} x(t) + \begin{bmatrix} \mathbf{0}_{2,2} \\ \mathbf{I}_2 \end{bmatrix} u(t), \quad (1)$$

driven by the piecewise continuous control signal $u : [0, t_f] \to U := \{\tilde{u} \in \mathbb{R}^2 : \|\tilde{u}\| \leq 1\}$, where t_f is the free final time. The robot is equipped with an omni-directional sensor footprint of size $r > 0$ around its current position $y(t)$, hence covering the area

$$\mathcal{O}(y(t)) = \{\tilde{y} \in \mathbb{R}^2 : \|y(t) - \tilde{y}\| \leq r\}, \quad (2)$$

where for any k, $\forall \tilde{y} \in W_k, \|\tilde{y} - w_k\| < r$. Since the regions $W_\pi \subset W$, where $\pi \in \Pi_d$ is true, are a-priori unknown, satisfying π is assumed to be preceded by a corresponding detection proposition δ_π. Accordingly, let all detection propositions form the finite set Δ and define the augmented atomic proposition set

$$\Pi = \Delta \cup \Pi_s \cup \Pi_d. \quad (3)$$

The overall system is modeled by a hybrid automaton $\mathcal{H} = (Q, X, U, f, E, \mathrm{Inv}, G, R, \mathrm{Init})$. The discrete state $q(t) = (q_1, q_2)$ at time t consists of $q_1 \in 2^\Pi$ denoting the set of atomic propositions that are currently true, and $q_2 \subseteq 2^{(\Pi_s \cup \Pi_d) \times W}$ capturing the set of all currently known pairs (π, w_k) of atomic propositions $\pi \in (\Pi_s \cup \Pi_d)$ and regions $W_k \in W$. Thus, the overall discrete state set is given by $Q \subseteq 2^\Pi \times 2^{(\Pi_s \cup \Pi_d) \times W}$. As Q is finite, the set of discrete state transitions (or events) $E \subseteq Q \times Q$ is also finite. Let E be partitioned into $E_1 \cup E_2 \cup E_3$, where for $q = (q_1, q_2), q' = (q'_1, q'_2) \in Q$,

$$E_1 = \{(q, q') : q'_1 = q_1 \cup \{\delta_\pi\}, q'_2 = q_2 \cup \{(\pi, w_k)\}\}$$

is the set of events that capture detections of a-priori unknown request regions,

$$E_2 = \{(q, q') : q'_1 = q_1 \cup \{\pi\}, (\pi, w_k) \in q_2\}$$

is the set of events that correspond to entering a known request region, and

$$E_3 = \{(q, q') : q'_1 = q_1 \setminus \{\pi\}, (\pi, w_k) \in q_2\}$$

is the set of events that correspond to leaving a known request region. With the introduced sensor paradigm (2), assume that a detection of a proposition at W_k occurs, when the distance between the current robot position $y(t)$ and the center point w_k of W_k becomes r. Defining the set

$$\mathcal{X}_{(\pi,w_k)} = \begin{cases} \{x : \|y(t) - w_k\| = r\}, & \text{if } \pi \in \Delta, \\ \{x : \tilde{y} \in W_k\}, & \text{else,} \end{cases}$$

we obtain the invariant map $\mathrm{Inv} : Q \to 2^X$ with

$$\mathrm{Inv}(q) = X \setminus (\cup_q \mathcal{X}_{(\pi,w_k)}),$$

the guard map $G : E \to 2^X$ with

$$G((q,q')) = \cup_{q'} \mathcal{X}_{(\pi,w_k)},$$

and the (trivial) reset map $R : E \times X \to X$ with $R(e,x) = x, \forall (e,x) \in E \times G(e)$. The initial state set is $\mathrm{Init} \subset (Q \times X)$. Consider a time line τ partitioned into $N+1$ intervals, i.e.,

$$\tau := (\underbrace{[t_0, t_1]}_{\tau_0}, \ldots, \underbrace{[t_N, t_{N+1}]}_{\tau_N}), t_0 = 0, t_{N+1} = t_f, \quad (4)$$

where N is the finite number of events. The input is an ordered set of functions $u = (u_0, \ldots, u_N)$, where $u_n : \tau_n \to U$ are absolutely continuous functions for $n \in \{0, \ldots, N\}$. Thus, if $\zeta = (\tau, q, x)_u$ is an execution of the hybrid automaton $\mathcal{H}$ for an input signal u, $q = (q^0, \ldots, q^N)$ is a discrete state trajectory with $q^n : \tau_n \to Q, q^n = \mathrm{const}, \forall t \in \tau_n$, x is the continuous state trajectory with $x = (x_1, \ldots, x_N)$ and $x_n : \tau_n \to X$ are absolutely continuous functions. The cost of an execution is

$$t_f = \sum_{n=0}^{N} (t_{n+1} - t_n). \quad (5)$$

Let the labeling function $L : E \to 2^\Pi$ return the set of propositions satisfied upon a transition, i.e. $L(q, q') = q'_1$. Then, let a word be the infinite sequence of labels produced by an execution ζ of $\mathcal{H}$, i.e., $L(\zeta) = L((\tau, q, x)_u) = q_1(t_0) q_1(t_1) \ldots$ (with a slight abuse of notation).

TABLE I

ATOMIC PROPOSITIONS FOR THE EXAMPLE.

π	Proposition
π_b	Visit base
π_k	Observe W_k
δ_p	Detect person
π_p	Assist person
δ_o	Detect obstacle
$\neg\pi_o$	Avoid obstacle

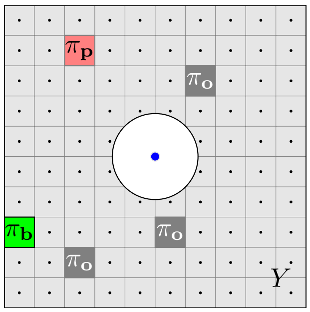


Fig. 1. Time-optimal rescue assignment for a mobile robot.

C. Specification

The robot has to satisfy a specification captured by an scLTL formula [23] over the set of atomic propositions Π. Roughly, an scLTL specification is build up from elements in Π, combined into a formula by Boolean operators $\neg$ (negation), $\wedge$ (conjunction), $\vee$ (disjunction), $\Rightarrow$ (implication), and temporal operators: $\mathbf{X}$ (next), $\mathbf{U}$ (until) and $\mathbf{F}$ (eventually). The semantics of scLTL are defined over infinite words $\sigma = \sigma_0\sigma_1\ldots$, where $\forall i, \sigma_i \in 2^{\Pi}$. A word σ satisfies the formula ϕ, if it holds at the first position of the word σ. Intuitively, $\mathbf{X}\phi$ states that the formula ϕ becomes true in the next position of a word, $\phi_1\mathbf{U}\phi_2$ expresses that ϕ_1 is true until ϕ_2 becomes true in a word, and $\mathbf{F}\phi$ requires that ϕ becomes true at some position in a word. While the semantics of scLTL are defined over infinite words, their satisfaction is guaranteed in finite time. A detailed description of the syntax and the semantics of scLTL is beyond the scope of this paper and we kindly refer the reader to [13], [23] for a more extensive treatment. With the introduced labeling function, an execution $\zeta|_u$ satisfies an scLTL formula ϕ, if its corresponding word $L(\zeta|_u)$ satisfies ϕ, denoted by $L(\zeta|_u) \vDash \phi$.

D. Problem formulation

The addressed OCP reads as follows.

Problem 1. *Given a hybrid system $\mathcal{H}$ and a specification as an scLTL formula ϕ over Π, find a control trajectory u that minimizes* (5) *subject to $L(\zeta|_u) \vDash \phi$.*

Example. *Consider an environment $Y = [0, 5]^2$ discretized by a regular grid with constant $d_g = 1$, such that $W = \{W_1, \ldots, W_{100}\}$. The environment contains the a-priori known $\Pi_s = \{\pi_b\} \cup \{\pi_k\}_{k\in\{0,\ldots,K\}}$, and a-priori unknown proposition sets $\Pi_d = \{\pi_o, \pi_p\}$ with corresponding detection proposition set $\Delta = \{\delta_o, \delta_p\}$ with statements as given in Table I, associated with the respective regions, depicted in Fig. 1. The dynamics of a quadrotor, maintaining a constant altitude above the ground, are reduced to* (1)*, with initial continuous state $x_0 = \mathbf{0}_4$. The size of the vehicle's sensing range is $r = \sqrt{2}d_g$.*

For a rescue assignment in Y, the vehicle has to observe regions $W_k \in W$ and avoid a priori unknown obstacles (e.g. trees) at W_k upon their detection, until a missing person is detected. Upon detection, it has to assist the person by visiting its location before returning to the base. This specification is represented by the scLTL formula

$$\phi := (expl\ \mathbf{U}\delta_p) \wedge assist \wedge (\delta_o \Rightarrow (\neg\pi_o)), \tag{6}$$

with assist$= (\delta_p \Rightarrow ((\neg\pi_b\mathbf{U}\pi_p) \wedge \mathbf{F}\pi_b))$, *expl*$= \bigwedge_{k=1}^{K}(\mathbf{F}\pi_k)$.

The above problem is challenging due to its inherent non-convexity and the lack of complete knowledge of the environment. In contrast to the majority of related approaches, the solution presented in the following does not require computing a finite discrete abstraction of $\mathcal{H}$ or discretizing time, and the OCP needs to be re-solved only upon detecting previously unknown propositional relations.

III. SOLUTION

The solution is based on the following key aspects: (i) restricting the motion of the robot to a family of curves, whose shape is determined by a finite number of parameters; (ii) introducing a function that represents a measure of violation of the specification by the trajectory of the vehicle that is used for model checking, and an approximate continuously differentiable version of it used for optimization; and (iii) solving an OCP w.r.t. the parameters of the curve by iterative gradient-based optimization and model checking.

A. Motion parameterization

Let the robot's position in Y be described by the parametric equation $y(t) = c(s(t), \theta)$, where $s(t)$ denotes the position of the robot along the curve c, $\theta \in \mathbb{R}^p$ is a parameter vector that controls the shape and course of c, and c is twice continuously differentiable with respect to time and θ. The normed Euclidean arc-length variable $s(t) \in [0, 1]$ is monotonically increasing over $t \in [0, t_f]$, such that $s(0) = 0$ at the initial position, and $s(t_f) = 1$ at the final position. Due to their rich expressiveness in terms of motion behaviors, in this work we employ Fourier series of respective order Γ_1 and Γ_2, i.e.,

$$y(t) = c(s, \theta) = \begin{bmatrix} a_0^1 + \sum_{\gamma=1}^{\Gamma_1} a_\gamma^1 \sin(4\pi^2\gamma f_1 s + \phi_\gamma^1) \\ a_0^2 + \sum_{\gamma=1}^{\Gamma_2} a_\gamma^2 \sin(4\pi^2\gamma f_2 s + \phi_\gamma^2) \end{bmatrix}, \tag{7}$$

where f_1 and f_2 are base frequencies, a_0^1 and a_0^2 are zero frequency components, a_γ^1 and a_γ^2 are amplitudes for the sinusoid functions with frequency γf_1 and γf_2, and ϕ_γ^1 and ϕ_γ^2 are phase differences with respect to the $(\gamma+1)$-th term of y_1 or y_2. Since only the ratio of f_1 and f_2 (and not their absolute values) determines the shape of (7), f_1 is treated as a free parameter, while f_2 = const. Thus, the parameter vector of (7) is $\theta=[f_1, A_1, A_2, \Phi_1, \Phi_2]^T$, where $A_1 = \left[a_0^1, \dots, a_{\Gamma_1}^1\right]^T$, $A_2 = \left[a_0^2, \dots, a_{\Gamma_2}^2\right]^T$, $\Phi_1 = \left[\phi_1^1, \dots, \phi_{\Gamma_1}^1\right]^T$ and $\Phi_2 = \left[\phi_1^2, \dots, \phi_{\Gamma_2}^2\right]^T$. With $i \in \{1,2\}$, the derivative of (7) w.r.t. θ is $\nabla_\theta y_i = \left[\frac{\partial y_i}{\partial f_1} \quad \frac{\partial y_i}{\partial A_1} \quad \frac{\partial y_i}{\partial A_2} \quad \frac{\partial y_i}{\partial \Phi_1} \quad \frac{\partial y_i}{\partial \Phi_2}\right]^T$. For a robot that starts at $y(0) = y_0$, we obtain $a_0^i = y_{0,i} - \sum_{\gamma=1}^{\Gamma_i} a_\gamma^i \sin(\phi_\gamma^i)$ for $i \in \{1,2\}$. Note that our elaborations are not restricted to this type of parametric functions.

Since our goal is to solve an approximate OCP w.r.t. θ, the first question is how to obtain the optimal control u for $\mathcal{H}$ for moving along the curve for a given θ. Let the first and second derivatives of (7) w.r.t. s be given by $c'(\theta, s) = \partial c/\partial s$ and $c''(\theta, s) = \partial^2 c/\partial s^2$, respectively. Further, let $\dot{s} = ds/dt$ and $\ddot{s} = d^2s/dt^2$ denote the time derivatives. For the velocity and the acceleration along (7), we respectively obtain

$$\dot{y}_i = \frac{dc_i(s,\theta)}{dt} = c_i'(s,\theta)\dot{s},\; \ddot{y}_i = \frac{d^2c_i(s,\theta)}{dt^2} = c_i''(s,\theta)\dot{s}^2 + c_i'(s,\theta)\ddot{s}.$$

for $i \in \{1,2\}$, and (1) is equivalently restated as

$$c_i''(s,\theta)\dot{s}^2 + c_i'(s,\theta)\ddot{s} = u_i(s), i \in \{1,2\}. \tag{8}$$

The curve (7) can be traversed in time

$$J_t = \int_0^{t_f} dt = \int_{s(0)}^{s(t_f)} \frac{1}{\dot{s}} ds. \tag{9}$$

Introducing the nonlinear substitutions $\alpha(s) = \ddot{s}$ and $\beta(s) = \dot{s}^2$ such that $\beta'(s) = 2\alpha(s)$, for a fixed θ we obtain the OCP

$$\begin{aligned} &\min_{\alpha,\beta,u} J_t = \int_0^1 \frac{1}{\sqrt{\beta(s)}} ds, \\ &\quad \text{s.t. } (8), \|u(s)\| \le 1, s \in [0,1], \beta'(s) = 2a(s), \\ &\qquad \beta(0) = \dot{s}_0^2, \beta(1) = \dot{s}_{t_e}^2, \beta(s) \ge 0. \end{aligned} \tag{10}$$

Since (9) is convex in terms of $\alpha(s)$, $\beta(s)$ and $u(s)$, the above OCP is convex for convex input constraints and for particular nonlinear motion dynamics without viscous friction [22].

Next we introduce a measure of violation function that captures the specification and sensing constraints.

B. Measure of violation

In previous works, the satisfaction of an (sc)LTL formula by a discrete-time linear system was enforced by defining a real positive function that resembles a control Lyapunov function over the corresponding dual automaton [6]. Following this idea, we introduce the function $J_\phi : \zeta|_u \to \mathbb{R}_{\ge 0}$ that denotes a measure of violation of ϕ by the hybrid execution $\zeta|_u$ parameterized by (7). For an execution $\zeta|_u$, we require that $J_\phi(\zeta|_u) = 0$, if $L(\zeta|_u) \vDash \phi$, and $J_\phi(\zeta|_u) > 0$, if $L(\zeta|_u) \nvDash \phi$. Since we want to use this function for gradient-based optimization, we also introduce an approximate continuously differentiable version of the measure of violation, denoted by $\tilde{J}_\phi : \zeta|_u \to \mathbb{R}_{\ge 0}$ with $\tilde{J}_\phi(\zeta|_u) = 0$, if $L(\zeta|_u) \vDash \phi$, and $\tilde{J}_\phi(\zeta|_u) > 0$, if $L(\zeta|_u) \nvDash \phi$.

For any scLTL formula ϕ over Π, there exists a Deterministic Finite Automaton (DFA) $\mathcal{A}_\phi$ with input alphabet 2^Π that accepts all good prefixes of ϕ [23]. Let $\mathcal{A}_\phi = (Q_\phi, \Sigma, \Delta_\phi, Q_{0,\phi}, Q_{f,\phi})$ be a DFA corresponding to the scLTL formula ϕ, where Q_ϕ is the finite set of states, $\Sigma = 2^\Pi$ the input alphabet, $\Delta_\phi : Q_\phi \times \Sigma \to Q_\phi$ the transition mapping, $Q_{0,\phi}$ the set of initial states and, $Q_{f,\phi} \subset Q_\phi$ the set of final or accepting states. At time $t \in [0, t_f)$, let $\mathcal{A}_{\tilde{\phi}} = (Q_{\tilde{\phi}}, \Sigma_{\tilde{\phi}}, \Delta_{\tilde{\phi}}, Q_{0,\tilde{\phi}}, Q_{f,\tilde{\phi}})$ denote the projection of $\mathcal{A}_\phi$ for the atomic proposition set $\tilde{\Pi} = \{\pi : \exists\{(\pi, w_k)\} \in q_2(t)\} \subset \Pi$. Intuitively, $\mathcal{A}_{\tilde{\phi}}$ describes the subset of the specification $\tilde{\phi} \subset \phi$ over the set of currently known propositional relations captured by the discrete state $q_2(t)$. Then, let the dual automaton of $\mathcal{A}_{\tilde{\phi}}$ be given by $\mathcal{A}_{\tilde{\phi}}^D = (Q_{\tilde{\phi}}^D, \Delta_{\tilde{\phi}}^D, Q_{0,\tilde{\phi}}^D, Q_{f,\tilde{\phi}}^D, \tau_{\tilde{\phi}}^D)$, where $Q_{\tilde{\phi}}^D = \{(q, \sigma, q') : (q, \sigma, q') \in \Delta_{\tilde{\phi}}\}$, $\Delta_{\tilde{\phi}}^D = \{((q, \sigma, q'), (q', \sigma, q'')) : (q, \sigma, q'), (q', \sigma, q'') \in \Delta_{\tilde{\phi}}\}$, $Q_{0,\tilde{\phi}}^D = \{(q, \sigma, q') : q \in Q_{0,\tilde{\phi}}\}$, $Q_{f,\tilde{\phi}}^D = \{(q, \sigma, q') : q' \in Q_{f,\tilde{\phi}}\}$ and $\tau_{\tilde{\phi}}^D : Q_{\tilde{\phi}}^D \to \Sigma_{\tilde{\phi}}, \tau_{\tilde{\phi}}^D((q, \sigma, q')) = \sigma$. Informally, the dual of an automaton is constructed by interchanging its states and its transitions. As the transitions of $\mathcal{A}_{\tilde{\phi}}$ become the states of $\mathcal{A}_{\tilde{\phi}}^D$, every state $(q, \sigma, q') \in Q_{\tilde{\phi}}^D$ can be associated with a measure of violation for σ, denoted by $d(\sigma, y(t))$. Assuming that $\sigma = \pi \in \tilde{\Pi}$ is satisfied upon observing W_k, let $d(\pi, y(t)) = 0$, if $\|y(t) - w_k\| \le r$, and $d(\pi, y(t)) = 1$, else. Let the corresponding approximate measure of violation $\tilde{d}(\pi, y(t))$ be monotonically increasing with the Euclidean distance $D_k = D(w_k, y(t)) = \|y(t) - w_k\|$ outside of the sensing range (2), which holds for example for

$$\tilde{d}(\sigma, y(t)) = \begin{cases} 1 - \exp(-(D_k - r)^2), & \text{if } D_k > r, \\ 0, & \text{else.} \end{cases} \tag{11}$$

Similarly, if $\sigma = \pi \in \tilde{\Pi}$ is satisfied upon visiting W_k, the measure of violation $d(\pi, y(t)) = 0$, if $y(t) \in W_k$, and $d(\pi, y(t)) = 1$, else. A corresponding (conservative) approximate measure of violation is obtained e.g. by

$$\tilde{d}(\sigma, y(t)) = \begin{cases} 1 - \exp(-(D_k - \frac{d_g}{2})^2), & \text{if } D_k > \frac{d_g}{2}, \\ 0, & \text{else,} \end{cases} \tag{12}$$

where d_g is the grid constant. If $\pi \in \tilde{\Pi}$ is satisfied at W_k, the measure of violation of $\sigma = \neg\pi$ is $d(\sigma, y(t)) = 1$, if $y(t) \in W_k$, and $d(\sigma, y(t)) = 0$, else. A corresponding (conservative) approximate measure of violation is e.g. given by

$$\tilde{d}(\sigma, y(t)) = \max\{0, 1 - 2D_k^2/d_g^2\}^2, \tag{13}$$

where d_g is the grid constant. Clearly, we can define a measure of violation for $\sigma = \pi_1 \wedge \pi_2$ by $d(\sigma, y) = (1 - (1 - d(\pi_1, y))(1 - d(\pi_2, y)))$, and an approximate measure of violation $\tilde{d}(\sigma, y)$ analogously.

Upon a detection δ_π at position $y(t)$, which corresponds to a transition $(q, q') \in \Delta_\phi$ with $\delta_\pi \in \sigma$, we obtain the projected automaton $\mathcal{A}_{\tilde{\phi}}$ as described above, set its initial state set to $Q_{0,\tilde{\phi}} = \{q'\}$, and compute the corresponding dual automaton $\mathcal{A}_{\tilde{\phi}}^D$. Then, let the set of all finite paths from $\bar{q} = (q', \sigma, q'') \in Q_{\tilde{\phi}}^D$ to $Q_{f,\tilde{\phi}}^D$ in $\mathcal{A}_{\tilde{\phi}}^D$ be denoted by $\mathcal{Y}_{\bar{q}}$, i.e.,

$$\begin{aligned} \mathcal{Y}_{\bar{q}} = \{\bar{\mathbf{q}} = \bar{q}_0 \bar{q}_1 \dots \bar{q}_l : l \in \mathbb{N}, i = 0, \dots, l-1, \bar{q}_0 = \bar{q}, \\ (\bar{q}_i, \bar{q}_{i+1}) \in \Delta_{\tilde{\phi}}^D, \bar{q}_l \in Q_{f,\tilde{\phi}}^D\}. \end{aligned} \tag{14}$$

With (7), the measure of violation of ζ_u w.r.t. an automaton path $\bar{\mathbf{q}} \in \mathcal{Y}_{\bar{q}}$ is given by the sum of the minimal corresponding measures of violation and sequencing constraints, i.e.,

$$J_{\bar{\mathbf{q}}}(\zeta|_u) = \sum_{i=0}^{l-1} (\min_{s_i \in [0,1]} d(\sigma_i, c(s_i, \theta)) + \max\{0, s_i - s_{i+1}\}).$$

Thus, the measure of violation (and, analogously, the approximate measure of violation) of an execution ζ_u reads

$$J_\phi(\zeta|_u) = \min_{\bar{\mathbf{q}} \in \mathcal{Y}_{\bar{q}}} J_{\bar{\mathbf{q}}}(\zeta|_u). \tag{15}$$

Example. *At time $t = 0$, by projecting $\mathcal{A}_\phi$ corresponding to the scLTL formula* (6) *for* $\tilde{\Pi}$*, we obtain $\mathcal{A}_{\tilde{\phi}}$ denoting the subformula $\tilde{\phi}$ = expl. Since expl corresponds to visiting all K regions in an arbitrary order, the corresponding measure of violation (and analogously approximate measure of violation) is* $J_{\tilde{\phi}}(\zeta|_u) = \sum_{k=1}^{K} \min_{s_k \in [0,1]} d(\sigma_i, c(s_k, \theta))$.

Remark 1. *Note that the expressions used for the approximate measure of violation are not unique – any continuously differentiable function that fulfills the boundary conditions of the corresponding measure of violation can be employed.*

C. Optimization

Introducing the tuning parameter $\mu \in [0, 1)$ allows for an appropriate weighting of the potentially antagonistic cost portions for the overall time (9) and for violating the specification (15), leading to the approximate cost

$$\hat{J} = \mu J_t + (1 - \mu)\tilde{J}_\phi. \tag{16}$$

Minimizing (16) w.r.t. θ will be accomplished by a gradient-based algorithm. As the continuous dynamics of the vehicle are not changing over the time line τ, the gradient of (16) (omitting function arguments) is given by

$$\nabla_\theta \hat{J} = \int_0^1 \mu \nabla_\theta \frac{1}{\dot{s}} ds + (1 - \mu)\nabla_\theta \tilde{J}_\phi. \tag{17}$$

Since the only terms in (8) that depend on θ are $c_i'(s, \theta)$ and $c_i''(s, \theta)$, we solve (10) for the current θ to obtain $u(s)$ and J_t^*, and the first term in (17) is computed by

$$\nabla_\theta \frac{1}{\dot{s}} = \nabla_\theta \sqrt{\frac{c_1'(s,\theta)c_2''(s,\theta) - c_2'(s,\theta)c_1''(s,\theta)}{c_1'(s,\theta)u_2 - c_2'(s,\theta)u_1}}.$$

To obtain $\tilde{J}_\phi$ and $\nabla_\theta \tilde{J}_\phi$, we solve a set of optimization problems for each automaton path $\bar{\mathbf{q}} \in \mathcal{Y}_{\bar{q}}$. The latter is computed using the following derivatives. With $\frac{\partial D_k}{\partial \theta} = \frac{1}{D_k}((y_1 - w_{1,k})\nabla_\theta y_1 + (y_2 - w_{2,k})\nabla_\theta y_2)$ and omitting function arguments, for (11) (and analogously for (12)), we obtain

$$\nabla_\theta \tilde{d} = \begin{cases} 2(D_k - r)\exp(-(D_k - r)^2)\frac{\partial D_k}{\partial \theta}, & \text{if } D_k > r, \\ 0, & \text{else,} \end{cases}$$

and for (13),

$$\nabla_\theta \tilde{d} = -\frac{4}{d_g^2} \max\{0, 1 - \frac{2D_k^2}{d_g^2}\} \frac{\partial D_k}{\partial \theta}.$$

The above derivatives are evaluated for the corresponding optimal s_i, which denotes the position along the curve that yields the minimal value of $\tilde{d}$. The optimal s_i, solving the inner optimization problem, is computed by using a gradient-projection algorithm. Then, for a given μ, (16) is minimized by the gradient-based algorithm for the parameter θ

$$\theta_{z+1} = \theta_z - \eta_z \nabla_\theta \hat{J}(\theta_z)\Big|_Y \tag{18}$$

where $\{\eta_z\}, z = 0, 1, \ldots$ is a properly selected step-size sequence for the projection of the gradient onto the feasible position space Y. The algorithm terminates when $|\nabla_\theta \hat{J}(\theta_z)|_Y| < \epsilon$ for a pre-specified threshold ϵ.

Algorithm 1 Receding horizon control optimization

Input: Set of partitions $W = \{W_1, \ldots, W_K\}$ with center points w_k in environment Y; scLTL formula ϕ over $\Pi = \Pi_s \cup \Pi_d$; robot with sensor range r and dynamics described by the hybrid automaton $\mathcal{H}$; curve $y(t) = c(s, \theta)$ with finite parameter vector θ; optimization parameters $\epsilon > 0$, $\kappa \in (0, 1)$

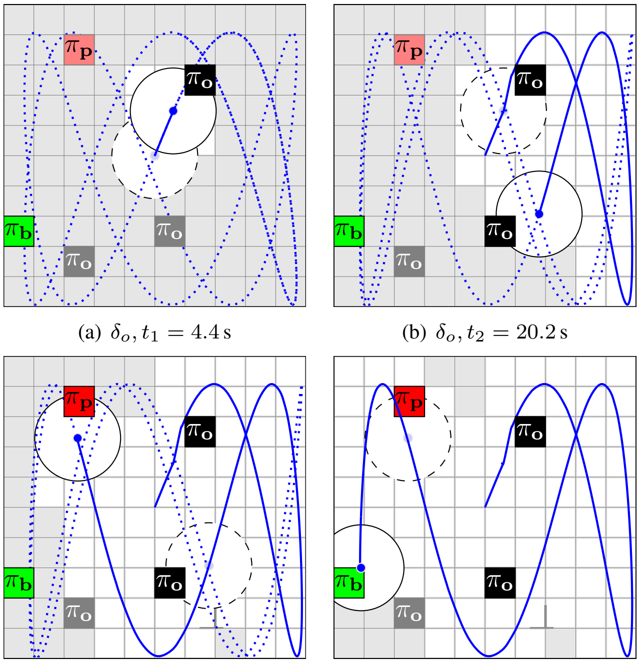
Output: The optimal control $u|^*_{[0,t_f]}$

1: **if** $(\vee_{\delta \in \Delta} y(t_n) \models \delta) \vee (t_0 = 0)$ **then**
2:   Set $\mu = 1$.
3:   Compute $u^*(s|_{[0,1]})$ and J_t^* through (10) for θ.
4:   Construct $J_{\tilde{\phi}}$ and $\tilde{J}_{\tilde{\phi}}$ for ϕ and $q(t_n)$.
5:   **while** $J_{\tilde{\phi}}(\zeta|_{u^*}) > 0$ **do**
6:     Set $\mu = \kappa\mu$.
7:     **repeat**
8:       Compute $u^*(s|_{[0,1]})$ and J_t^* through (10) for θ.
9:       Compute $\hat{J}(\theta)$ and $\nabla_\theta \hat{J}(\theta)|_Y$ with μ, and update θ through (18).
10:     **until** $|\nabla_\theta \hat{J}(\theta)|_Y| < \epsilon$
11:   **end while**
12:   **return** $u^*(s|_{[0,1]})$
13: **end if**

This leads to the proposed solution, summarized in Alg. 1. At t=0 or upon a detection of a previously unknown propositional relation, (10) is solved and the current measure of violation $J_{\tilde{\phi}}$ (used for model checking) and its approximation $\tilde{J}_{\tilde{\phi}}$ (used for optimization) are computed as described above. As long as the current solution violates the specification (line 5), (10) and (18) are solved iteratively, where μ decreases with each iteration, thus increasing the importance of $\tilde{J}_{\tilde{\phi}}$ over J_t in the optimization. The algorithm returns the optimal control input trajectory, when a non-violating execution is found. The soundness and completeness of the proposed procedure are summarized in the following proposition.

Proposition 1. *Alg. 1 yields a control u with $L(\zeta|_u) \models \phi$, if a solution to Problem 1 exists.*

Proof. Consider a hybrid execution ζ_u that violates ϕ, i.e. $L(\zeta|_u) \not\models \phi$, and u was obtained by Alg. 1. The corresponding measure of violation is $J_\phi(\zeta|_u) > 0$, by definition. However, Alg. 1 cannot return an input u that produces an execution with $J_\phi(\zeta|_u) > 0$ (line 5). Under the assumption that a solution exists and since the underlying OCPs are convex, the loop terminates in finite time. □



(a) $\delta_o, t_1 = 4.4\,\mathrm{s}$ (b) $\delta_o, t_2 = 20.2\,\mathrm{s}$ (c) $\delta_p, t_3 = 30.4\,\mathrm{s}$ (d) $t_f = 39.2\,\mathrm{s}$

Fig. 2. Snapshots of the robot's motion at detection instants and the final time t_f obtained with Alg. 1. Executed trajectories are denoted by a solid, and planned trajectories by dotted lines.

Alg. 1 relies on repetitively solving a finite set of OCPs by gradient projection and convex optimization, which both run in polynomial time.

IV. Case study

The methods were implemented in MATLAB and all computations were performed on an Intel® Core™ i7 2.20 GHz processor with 8 GB RAM. The DFA $\mathcal{A}_\phi$ corresponding to the scLTL specification ϕ was obtained with *scheck* [24]. The OCP (10) was solved with SeDuMi. A (pseudo-)random initial parameter vector θ_0 (that satisfies the initial condition $y(0)$) with $\Gamma_1 = \Gamma_2 = 2$ was chosen, as well as $\epsilon = 10^{-4}$ and $\kappa = 0.8$. Figure 2 shows snapshots of the robot's motion at detection instants and the final time, obtained by applying Alg. 1. On average, online re-computation took $10.6\,\mathrm{s}$. Note that the code was written without a particular emphasis on computational efficiency. Note that the convergence speed and the quality of the outcomes strongly depend on θ_0 and the step size selection method for gradient optimization.

V. Conclusions

We addressed the time-optimal control of a robot with limited sensing range that has to satisfy a specification given as a syntactically co-safe linear temporal logic formula in a partially unknown environment. Our approach was based on parameterizing the continuous motion of the vehicle and employing a measure of violation function to enforce the satisfaction of the specification. Then, we formulated a parametric optimization problem for minimizing the overall time and the measure of violation that has to be re-solved only upon detecting previously unknown properties of the environment. Further work will focus on extending the approach for full LTL specifications or a team of robots.

Acknowledgments

The authors would like to thank J. Raisch and the anonymous reviewers for their comments.

References

[1] G. E. Fainekos, A. Girard, H. Kress-Gazit, and G. J. Pappas, "Temporal logic motion planning for dynamic robots," *Automatica*, vol. 45, no. 2, pp. 343–352, 2009.
[2] E. Plaku, L. Kavraki, and M. Vardi, "Hybrid systems: From verification to falsification by combining motion planning and discrete search," *Formal Methods in Syst. Design*, vol. 34, pp. 157–182, 2009.
[3] T. Wongpiromsarn, U. Topcu, and R. M. Murray, "Receding horizon control for temporal logic specifications," in *Hybrid Systems: Computation and control (HSCC)*, 2010, pp. 101–110.
[4] S. L. Smith, J. Tumova, C. Belta, and D. Rus, "Optimal path planning for surveillance with temporal-logic constraints," *Int. Journal of Robotics Research (IJRR)*, vol. 30, no. 14, pp. 1695–1708, 2011.
[5] A. Bhatia, L. Kavraki, and M. Vardi, "Motion planning with hybrid dynamics and temporal goals," in *IEEE CDC*, 2010.
[6] E. A. Gol, M. Lazar, and C. Belta, "Language-guided controller synthesis for linear systems," *IEEE Trans. Automatic Control*, vol. 59, no. 5, pp. 1163–1176, 2014.
[7] V. Nenchev, C. Belta, and J. Raisch, "Optimal motion planning with temporal logic and switching constraints," in *14th European Control Conference (ECC'15)*, 2015, pp. 1135–1140.
[8] S. Karaman, R. Sanfelice, and E. Frazzoli, "Optimal control of mixed logical dynamical systems with linear temporal logic specifications," in *IEEE Conf. on Decision and Control (CDC)*, 2008, pp. 2117–2122.
[9] E. M. Wolff, U. Topcu, and R. M. Murray, "Automaton-guided controller synthesis for nonlinear systems with temporal logic," in *IEEE/RSJ IROS*, 2013, pp. 4332–4339.
[10] B. Yordanov, J. Tumova, I. Cerna, J. Barnat, and C. Belta, "Temporal logic control of discrete-time piecewise affine systems," *Automatic Control, IEEE Trans. on*, vol. 57, no. 6, pp. 1491–1504, 2012.
[11] M. Svorenova, I. Cerna, and C. Belta, "Optimal control of mdps with temporal logic constraints," in *IEEE Conf. on Decision and Control (CDC)*, 2013, pp. 3938–3943.
[12] M. Vardi, "Probabilistic linear-time model checking: An overview of the automata-theoretic approach," *Formal Methods for Real-Time and Probabilistic Systems*, pp. 265–276, 1999.
[13] C. Baier and J.-P. Katoen, *Principles of model checking*. MIT Press, 2008.
[14] A. Jones, M. Schwager, and C. Belta, "A receding horizon algorithm for informative path planning with temporal logic constraints," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2013.
[15] M. Maly, M. Lahijanian, L. Kavraki, H. Kress-Gazit, and M. Vardi, "Iterative temporal motion planning for hybrid systems in partially unknown environments," in *HSCC*, 2013.
[16] V. Nenchev and J. Raisch, "Towards time-optimal exploration and control by an autonomous robot," in *21st Mediterranean Conf. on Control and Automation (MED'13)*, 2013, pp. 1236–1241.
[17] G. Droge, P. Kingston, and M. Egerstedt, "Behavior-based switch-time mpc for mobile robots," in *IEEE/RSJ IROS*, 2012, pp. 408–413.
[18] V. Nenchev and C. G. Cassandras, "Optimal exploration and control for a robotic pick-up and delivery problem," in *IEEE Conf. on Decision and Control (CDC'14)*, 2014, pp. 7–12.
[19] X. Lin and C. G. Cassandras, "Trajectory optimization for multi-agent persistent monitoring in two-dimensional spaces," in *Proc. of 53rd Conf. on Decision and Control (CDC'14)*, 2014, pp. 3719–3724.
[20] V. Nenchev and C. G. Cassandras, "Optimal exploration and control for a robotic pick-up and delivery problem in two dimensions," in *54th Conf. on Decision and Control (CDC'15)*, 2015, pp. 258–263.
[21] B. d'Andrea Novel, G. Bastin, and G. Campion, "Dynamic feedback linearization of nonholonomic wheeled mobile robots," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 1992, pp. 2527–2532.
[22] D. Verscheure, B. Demeulenaere, J. Swevers, J. De Schutter, and M. Diehl, "Time-optimal path tracking for robots: A convex optimization approach," *Automatic Control, IEEE Trans.*, vol. 54, no. 10, pp. 2318–2327, 2009.
[23] O. Kupferman and M. Vardi, "Model checking of safety properties," *Formal Methods in System Design*, vol. 19, no. 3, pp. 291–314, 2001.
[24] T. Latvala, "Efficient model checking of safety properties," in *10th International SPIN Workshop*. Springer, 2003.