Automatica 77 (2017) 112–119

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

# Hierarchical trajectory optimization for a class of hybrid dynamical systems[☆]

Raghvendra V. Cowlagi

*Aerospace Engineering Program, Worcester Polytechnic Institute, Worcester, MA 01609, USA*

ARTICLE INFO

*Article history:*
Received 29 March 2016
Received in revised form
17 August 2016
Accepted 2 November 2016

*Keywords:*
Hierarchically intelligent control
Autonomous mobile robots
Optimal trajectory

ABSTRACT

We discuss trajectory optimization for a class of hybrid systems with a natural, hierarchical separation of discrete and continuous dynamics, where the continuous dynamics do not change with the discrete state. The trajectory optimization problem considered requires that a discrete state sequence and a continuous state trajectory must be both determined to minimize a single cost function, such that the discrete state sequence also solves a symbolic planning problem. We model this symbolic planning problem as a search on a planning graph, and we introduce a family of graphs called *lifted planning graphs* parametrized by an integer $H$. We define a family of continuous state trajectory optimization problems and associate them with edge costs in the lifted planning graphs. Next, we present an algorithm for finding an optimal solution to the hybrid trajectory optimization problem, which includes mapping paths in the lifted planning graphs to discrete state sequences and continuous state trajectories. We show that the cost of optimal hybrid trajectories is a nonincreasing function of $H$, and that there exists a finite $H$ for which this cost attains a minimum. We illustrate the proposed algorithm with numerical simulation results for two application examples: an autonomous mobile vehicle and an autonomous robotic manipulator.



## 1. Introduction

The typical topics addressed in the literature on the control of hybrid dynamical systems include mathematical modeling and existence of solutions (Goebel, Sanfelice, & Teel, 2012); Lyapunov-based robust stabilization and switching conditions for stability (Branicky, 1998; Haddad, Chellaboina, & Nersesov, 2006; Liberzon, 2003; Sanfelice, 2013); abstraction, verification, reachability, and dead-lock (Alur, Henzinger, Lafferriere, & Pappas, 2000; Tabuada, 2008); and hybrid optimal control, which is the subject of this paper.

The *mode switched system* model dominates the literature on hybrid optimal control. For problems where the mode switching sequence is fixed, first-order necessary conditions involving adjoint states, similar to the Pontryagin Minimum Principle, are developed (Garavello & Piccoli, 2005; Piccoli, 1998; Sussmann, 1999). Explicit algorithms and applications for control design based on such necessary conditions are presented (Pakniyat & Caines, 2015a; Shaikh & Caines, 2007). For controlled switching, the determination of derivatives of the value function with respect to switching times is addressed (Ding, Wardi, & Egerstedt, 2009; Kamgarpour & Tomlin, 2012; Xu & Antsaklis, 2002). Cassandras, Pepyne, and Wardi (2001) address the case where switching time instants are described by a controlled event-driven dynamical system. Extensions of dynamic programming to hybrid systems are addressed (Barles, Dharmatti, & Ramaswamy, 2010; Hedlund & Rantzer, 2002; Rungger & Stursberg, 2011), including connections to the adjoint variable in the necessary conditions (Pakniyat & Caines, 2015b), but the fundamental "curse of dimensionality" remains at least as severe as for continuous systems. The determination of optimal mode selection in systems with autonomous continuous dynamics is addressed using gradient descent-like algorithms (Axelsson, Wardi, Egerstedt, & Verriest, 2008; Sager, 2009). Zhu and Antsaklis (2013) provide an excellent survey of recent advances.

Hybrid systems involving a natural hierarchy of discrete and continuous subsystems are not directly addressed by the aforesaid methods. Such hierarchies arise, for instance, in autonomous robotics, where the high-level autonomy and intelligence are modeled by discrete state automata or state transition systems, and the low-level physical system and its control are modeled by continuous-time dynamical systems. Similar hierarchies arise in the chemical process industry (high-level production scheduling

[☆] The material in this paper was partially presented at the 5th Analysis and Design of Hybrid Systems, October 14–16, 2016, Atlanta, GA, USA. This paper was recommended for publication in revised form by Associate Editor Andrey V. Savkin under the direction of Editor Ian R. Petersen.

*E-mail address:* rvcowlagi@wpi.edu.

http://dx.doi.org/10.1016/j.automatica.2016.11.040

and low-level process control) and the need for algorithms for their seamless integration to achieve overall optimality is emphasized (Engell & Hajunkoski, 2012; Engell, Kowalewski, Schulz, & Stursberg, 2000). In the optimal control of such hybrid systems, the discrete state sequence and the continuous state trajectory must be both determined with the objective of minimizing a single cost function. Furthermore, the discrete state sequence is often associated with a *symbolic* planning problem (Belta et al., 2007; Bemporad & Giorgetti, 2006). To this end, a *hybrid automaton* model (Alur et al., 2000) is more suitable, and the preservation of the natural hierarchy of the system of engineering importance (Engell & Hajunkoski, 2012).

We discuss a hierarchical trajectory optimization algorithm for hybrid control problems involving symbolic planning at the higher level and continuous state trajectory optimization at the lower level. We model the symbolic planning problem as a graph search on a *planning graph*. Next, we introduce a family of graphs called *lifted planning graphs* parametrized by an integer $H$. Next, we define a family of low-level trajectory optimization problems and associate them with edge costs in the lifted planning graphs. Next, we present an algorithm for finding an optimal solution to the hybrid trajectory optimization problem, which involves mapping paths in the lifted planning graphs to discrete-state sequences and continuous-state trajectories. We show that the cost of optimal hybrid trajectories is a nonincreasing function of $H$, and that there exists a finite $H$ for which this cost attains a minimum. We illustrate the proposed algorithm with two examples: an autonomous mobile vehicle, and an autonomous robotic manipulator.

*Contributions:* First, we propose a new hybrid trajectory optimization algorithm for the case where an optimal discrete state sequence and an optimal continuous state trajectory with free terminal time must be both simultaneously computed. The proposed algorithm is applicable to hybrid systems with a natural hierarchy of discrete and continuous subsystems, where the continuous dynamics do not change with the discrete state. The proposed algorithm introduces a loose coupling between the optimization problems at the two hierarchical levels, without entirely eliminating the hierarchical problem structure. This feature is beneficial because the optimization problems at the two hierarchical levels are typically solved using fundamentally different methods. Second, we provide important application examples from robotics that involve a symbolic planning problem at the higher level, and control for a dynamical system at the lower level. We demonstrate that the proposed approach produces plans with significantly lower cost compared to the ad hoc approach of hierarchical separation. Third, we provide a fundamental result for hierarchical hybrid systems on the relationship of sequences of discrete state transitions to costs defined on continuous state trajectories.

Preliminary results of this paper were previously presented (Cowlagi, 2015). This paper presents broader theoretical results, formal proofs, and an additional application example, which were not previously discussed.

## 2. Problem formulation

We consider a hybrid dynamical system $\mathscr{H}$ with discrete state in a finite set $V$, with $|V| = N^{\mathrm{V}} \in \mathbb{Z}_{>0}$, and continuous state in an open set $\mathcal{D} \subseteq \mathbb{R}^n$. The state of the system is denoted by $(v, \xi)$, where $v \in V$ and $\xi \in \mathcal{D}$. We label vertices in $V$ with superscripts, e.g. $v^1, v^2, \dots, v^{N^{\mathrm{V}}}$ whereas we use *sub*script to denote an index within a sequence of discrete states. The evolution of the discrete state is described by a labeled transition system $\mathcal{T}$ consisting of a finite set of labels $\Omega$, with $|\Omega| = N^{\mathrm{L}} \in \mathbb{Z}_{>0}$, and a transition map $\delta : V \times \Omega \to V$. We assume that each discrete state $v \in V$ is associated with a compact set $\Phi_v \subseteq \mathcal{D}$, such that for every pair $v^1, v^2 \in V$,

$$\left(v^k \neq v^\ell\right) \Leftrightarrow \left(\Phi_{v^k} \cap \Phi_{v^\ell} = \varnothing\right), \quad k, \ell \in \{1, \dots, N^{\mathrm{V}}\}.$$

The initial state of the system $\mathscr{H}$ is denoted by $(v^{\mathrm{s}}, \xi^{\mathrm{s}})$, and unless otherwise stated, we assume that $\xi^{\mathrm{s}} \in \Phi_{v^{\mathrm{s}}}$. The evolution of the continuous state is described by

$$\dot{\xi}(t) = f\left(\xi(t), u(t)\right), \tag{1}$$

where $u$ is the control input and $U \in \mathbb{R}^m$ is the set of admissible control input values. The initial time is assumed to be zero. We denote by $\mathcal{U}$ the set of all piecewise continuous functions of time taking values in $U$. We assume that $f : \mathbb{R}^{n+m} \to \mathbb{R}^n$ is globally Lipschitz continuous in $\mathbb{R}^{n+m}$ due to which the global existence and uniqueness of a solution to (1), denoted $\xi(t; \xi^{\mathrm{s}}, u)$, is guaranteed for all $(\xi, u) \in \mathcal{U}$ (Haddad & Chellaboina, 2008). Note that jumps in the continuous state are not permitted.

A *plan* $\overline{\omega} = (\omega_0, \omega_1, \dots, \omega_{P-1})$ is uniquely associated with a finite sequence of discrete states $\mathbf{v} = (v_0, v_1, \dots, v_P)$ such that $\delta(v_k, \omega_k) = v_{k+1}$ for each $k = 0, \dots, P-1$. We make this association explicit by denoting the plan as $\overline{\omega}(\mathbf{v})$, which is said to *transfer* the discrete state from $v_0$ to $v_P$. For a plan $\overline{\omega}$, an *executive control* is a pair $(t^{\mathrm{f}}, u)$ such that the control input $u$ drives the continuous state through each of the regions $\Phi_{v_k}$ in sequential order in a finite time $t^{\mathrm{f}}$. More precisely, $(t^{\mathrm{f}}, u)$ are such that there exists a strictly increasing sequence $\{t_1, t_2, \dots, t_P\} \in \left[0, t^{\mathrm{f}}\right]$ satisfying $t_P = t^{\mathrm{f}}$, and $\xi(t_k; \xi^{\mathrm{s}}, u) \in \Phi_{v_k}$, for each $k = 1, 2, \dots, P$. We denote $(t^{\mathrm{f}}, u) \vdash \overline{\omega}$ to indicate that $(t^{\mathrm{f}}, u)$ is an executive control for $\overline{\omega}$. The *cost* $\Lambda$ of an executive control $(t^{\mathrm{f}}, u)$ is

$$\Lambda(t^{\mathrm{f}}, u) := \int_0^{t^{\mathrm{f}}} L(\xi(t; \xi^{\mathrm{s}}, u),\ u(t))\,\mathrm{d}t, \tag{2}$$

where $L : \mathbb{R}^{n+m} \to \mathbb{R}_+$ is a bounded function.

The main problem of interest in this paper as follows.

**Problem 1.** Let $v^{\mathrm{s}}, v^{\mathrm{g}} \in V$ and $\xi^{\mathrm{s}} \in \mathcal{D}$ be prespecified. Find a plan $\overline{\omega}^*$ that transfers the discrete state from $v^{\mathrm{s}}$ to $v^{\mathrm{g}}$. Also find an executive control $(t^{\mathrm{f}*}, u^*) \vdash \overline{\omega}^*$ such that, for every plan $\overline{\omega}$ that achieves the same discrete state transfer, and every executive control $(t^{\mathrm{f}}, u) \vdash \overline{\omega}$,

$$\Lambda(t^{\mathrm{f}*}, u^*) \leq \Lambda(t^{\mathrm{f}}, u).$$

We assume that the discrete state transition system $\mathcal{T}$ models a high-level symbolic planning problem, a specific example of which is the *classical planning problem* (CPP) in the artificial intelligence literature (Russell & Norvig, 2003). A CPP consists of

(1) A finite set of *objects* $\{o^1, \dots, o^{N^{\mathrm{O}}}\}$.
(2) A finite set of *predicates* $\{p^1, \dots p^{N^{\mathrm{P}}}\}$. Each predicate accepts one or more arguments from the set of objects. A predicate evaluated for specific object(s) is a literal, which takes values in $\{\mathsf{true}, \mathsf{false}\}$.
(3) A finite set of *CPP states* $V = \{v^1, \dots v^{N^{\mathrm{V}}}\}$, where each state is a conjunction of literals, which can take values in $\{\mathsf{true}, \mathsf{false}\}$. The *current CPP state* is the unique state that is $\mathsf{true}$. An initial CPP state $v^{\mathrm{s}}$ and a goal CPP state $v^{\mathrm{g}}$ are prespecified.
(4) A finite set of *actions* $A = \{\omega^1, \dots \omega^{N^{\mathrm{L}}}\}$. Each action $\omega \in A$ is associated with a *precondition* $\mathsf{pre}(\omega)$ and an *effect* $\mathsf{eff}(\omega)$. The precondition $\mathsf{pre}(\omega)$ describes the conditions that must be true before the action $\omega$ can be executed, whereas the effect $\mathsf{eff}(\omega)$ describes the changes to the current CPP state, i.e., there is a unique $v \in V$ that is true when $\mathsf{eff}(\omega)$ is true, denoted by $v \equiv \mathsf{eff}(\omega)$.