A Descent Algorithm for the Optimal Control of Constrained Nonlinear Switched Dynamical Systems

Humberto Gonzalez, Ram Vasudevan, Maryam Kamgarpour,
S. Shankar Sastry, Ruzena Bajcsy, and Claire J. Tomlin
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
Berkeley, CA 94720, USA
{hgonzale,ramv,maryamka,sastry,bajcsy,tomlin}@eecs.berkeley.edu

ABSTRACT

One of the oldest problems in the study of dynamical systems is the calculation of an optimal control. Though the determination of a numerical solution for the general nonconvex optimal control problem for hybrid systems has been pursued relentlessly to date, it has proven difficult, since it demands nominal mode scheduling. In this paper, we calculate a numerical solution to the optimal control problem for a constrained switched nonlinear dynamical system with a running and final cost. The control parameter has a discrete component, the sequence of modes, and two continuous components, the duration of each mode and the continuous input while in each mode. To overcome the complexity posed by the discrete optimization problem, we propose a bi-level hierarchical optimization algorithm: at the higher level, the algorithm updates the mode sequence by using a single-mode variation technique, and at the lower level, the algorithm considers a fixed mode sequence and minimizes the cost functional over the continuous components. Numerical examples detail the potential of our proposed methodology.

Categories and Subject Descriptors

G.1.6 [**Numerical Analysis**]: Optimization—*Constrained optimization, Nonlinear programming*

General Terms

Algorithms

Keywords

Constrained optimal control, Switched hybrid systems

1. INTRODUCTION

The determination of an optimal control is one of the oldest problems in the study of dynamical systems. In particular, the calculation of a numerical solution to the general nonconvex optimal control problem for hybrid systems has proven difficult for numerous reasons. Most importantly, hybrid systems comprise a far too general set of systems (i.e. a system under consideration may not even satisfy the conditions required for existence and uniqueness). In this paper, we consider a subset of hybrid systems with great utility: constrained nonlinear switched dynamical systems; that is, systems which consist of a finite number of constrained nonlinear subsystems and a switching law that describes which of these subsystems is active at a given time. The control parameter for such systems has both a discrete component, the sequence of modes, and two continuous components, the duration of each mode and the continuous input. Switched systems arise in a variety of applications including the modeling of the dynamics of automobiles and locomotives in different gears [13, 21], the modeling of the dynamics of biological systems [10], situations where a control module has to switch its attention among a number of subsystems [15, 20, 24], and situations where a control module has to collect data sequentially from a number of sensory sources [6, 7].

Recently, there has been growing interest in the optimal control of such hybrid systems, stemming from Branicky et al.'s seminal work that established a necessary condition for the optimal trajectory under a general cost function in terms of quasi-variational inequalities [5]. Unfortunately, they provide no method for the calculation of the desired control. Several researchers have attempted to address the special case of piecewise-linear or affine systems [2, 4, 11, 19, 27]. Most of these methods employ variants of dynamic programming; however, since after each iteration of their algorithm the number of possible switches grows exponentially, the representation of the optimal value function becomes increasingly complex. These papers focus on addressing this particular shortcoming by considering a variety of possible relaxations of the optimal value function.

More pertinently, Xu and Antsaklis consider the optimal control of a nonlinear switched system under a fixed, prespecified modal sequence and develop a bi-level hierarchical optimization algorithm: at the higher level, a conventional optimal control algorithm finds the optimal continuous input given the sequence of active subsystems and the switching instants and at the lower level, a nonlinear optimization algorithm finds the locally optimal switching instants [25, 26]. Though we employ a similar division of labor, our approach also optimizes over switching sequences while considering constraints. Shaikh and Caines consider the same problem and utilize an identical bi-level hierarchical optimization al-

HSCC'10, April 12–15, 2010, Stockholm, Sweden.

gorithm under a prespecified sequence [22]. However, rather than maintain the same prespecified sequence, they search through all possible sequences within a fixed distance (using the Hamming distance) of the prespecified sequence to find a sequence with a lower optimal cost after performing the original optimization. Instead of resorting to this type of brute force search, we employ a descent technique to find an optimal switching sequence similar to Axelsson and Egerstedt et al. who consider the special case of nonlinear, autonomous switched systems (i.e. systems wherein the control input is absent) [1, 8, 9]. They employ a similar bi-level hierarchical algorithm: at the higher level, the algorithm updates the mode sequence by considering a single mode insertion technique, and at the lower level, the algorithm considers a fixed mode sequence and minimizes the cost functional over the switching times. This method falls short since it only considers a restrictive subclass of switched systems, but it provides the starting point for the results presented in this paper.

In this paper, we construct an optimal control algorithm for constrained nonlinear switched dynamical systems. We develop a bi-level hierarchical algorithm that divides the problem into two nonlinear constrained optimization problems. At the lower level, we keep the modal sequence fixed and determine the optimal mode duration and optimal continuous input. At the higher level, we employ a single mode insertion technique to construct a new lower cost sequence. The result of this approach is an algorithm that provides a sufficient condition to guarantee the local optimality of the mode duration and continuous input while decreasing the overall cost via mode insertion, which is a powerful outcome given the generality of the problem under consideration. This paper is organized as follows: Section 2 provides the mathematical formulation of the problem under consideration, Section 3 describes the optimal control algorithm which is the primary result of this paper, Section 4 details how we prove the convergence of our algorithm, Section 5 considers an efficient numerical implementation of the optimal control scheme, Section 6 presents numerical experiments, and Section 7 concludes the paper.

2. PROBLEM FORMULATION

In this section, we present the mathematical formalism and define the problem wo solve in the remainder of this paper. W begin by definining a space, $\mathcal{X}$, by:

$$\mathcal{X} = \Sigma \times \mathcal{S} \times \mathcal{U}, \tag{1}$$

where Σ denotes the discrete mode sequence space, $\mathcal{S}$ denotes the transition time space, $\mathcal{U}$ denotes the continuous input space. We consider each of these spaces in more detail below.

First, we define the discrete mode sequence space. The continuous dynamics for each discrete mode q in $\mathcal{Q} = \{1, 2, \dots, R\}$ is given by the vector field $f_q : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$. We also include an additional mode, *NF*, for notational convenience, which denotes the discrete mode in which the trajectories stop evolving (i.e. the dynamics of mode *NF* is defined as $f_{NF} : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$, $f_{NF}(x, u) = 0$). Since at each iteration our optimization algorithm, which we describe in the next section, may provide a mode sequence with a varying number of total modes, the mode sequence space is an infinite dimensional space; however, in order to avoid pathological examples (i.e. Zeno behavior), we only allow a finite number of non-zero vector fields:

$$\Sigma = \bigcup_{N=1}^{\infty} \Sigma_N,$$
$$\Sigma_N = \left\{ \sigma \in \tilde{\mathcal{Q}}^{\mathbb{N}} \,\middle|\, \sigma(j) \in \mathcal{Q} \quad j \le N,\ \sigma(j) = NF \quad j > N \right\},$$

where $\tilde{\mathcal{Q}} = \mathcal{Q} \cup \{NF\}$.

Second, we define the amount of time spent in each discrete mode, using an absolutely summable sequence s in S:

$$\mathcal{S} = \bigcup_{N=1}^{\infty} \mathcal{S}_N,$$
$$\mathcal{S}_N = \left\{ s \in l^1 \,\middle|\, s(j) \ge 0 \ \ \forall j \le N,\ s(j) = 0\ \forall j > N \right\},$$

where l^1 denotes the space of absolutely summable sequences. Observe that though we require this mode duration sequence to only have a finite number of non-zero values, it may have zero entries interspersed between non-zero entries. Also note that this notion of relative times stands in contrast to the conventional notion in the hybrid system literature of the absolute time at which a certain discrete mode begins. However, maintaining the notion of absolute times, which we call the jump time sequence, is also convenient. We define the jump time sequence $\mu : \mathbb{N} \times \mathcal{S} \to [0, \infty)$ by:

$$\mu(i; s) = \begin{cases} 0 & \text{if } i = 0 \\ \sum_{k=1}^{i} s(k) & \text{if } i \neq 0. \end{cases} \tag{2}$$

We also define $\mu_f(s) = \|s\|_{l^1} = \sum_{k=1}^{\infty} s(k)$ which is well defined for each $s \in \mathcal{S}$. We can also take a time t in $[0, \infty)$ and determine to which index t belongs by considering $\kappa : [0, \infty) \times \mathcal{S} \to (\mathbb{N} \cup \infty)$:

$$\kappa(t; s) = \begin{cases} 1 & \text{if } t = 0 \\ \max\{i \in \mathbb{N} \cup \infty : \mu(i, s) < t\} + 1 & \text{if } t \neq 0. \end{cases} \tag{3}$$

Finally we define, $\pi : [0, \infty) \times \mathcal{S} \to \{\text{NF}, 1, \dots, R\}$ as a function that returns the mode at a time t:

$$\pi(t; s) = \begin{cases} \sigma(\kappa(t; s)) & \text{if } \kappa(t; s) < \infty \\ \text{NF} & \text{if } \kappa(t; s) = \infty. \end{cases} \tag{4}$$

Since we allow zero entries to be interspersed between non-zero entries for elements in $\mathcal{S}$, we may have $\mu(i, s) = \mu(i + 1, s)$. Capturing these zeros is critical in the definition of our optimization algorithm. Therefore, we define functions $m, n : \mathbb{N} \times \mathcal{S} \to \mathbb{N}$ as follows:

$$m(i; s) = \min\{m \le i : \mu(i; s) = \mu(m; s)\} \tag{5}$$
$$n(i; s) = \max\{n \ge i : \mu(i; s) = \mu(n; s)\}. \tag{6}$$

Since the choice of $s \in \mathcal{S}$ is clear in context, we supress the dependence on it in μ_f and Equations (2), (3), (4), (5), and (6). We illustrate these various definitions by considering the example presented in Figure 1 and Table 1 for a σ and s defined as follows:

$$\sigma = \{1, 2, 3, 4, 5, 3, 4, NF, \dots\} \in \Sigma_7 \tag{7}$$
$$s = \{1, 0, 0, 0, 1, 0, 1, 0, \dots\} \in \mathcal{S}_7. \tag{8}$$

Third, we require the continuous input space, $\mathcal{U}$, to be bounded for all time:

$$\mathcal{U} = \left\{ u \in L^2([0, \infty), \mathbb{R}^m) \,\middle|\, \|u(t)\| \le M,\ \forall t \in [0, \infty) \right\}. \tag{9}$$

An element of our space $\mathcal{X} = \Sigma \times \mathcal{S} \times \mathcal{U}$ is then denoted by a 3−tuple $\xi = (\sigma, s, u)$, where $\sigma \in \Sigma, s \in \mathcal{S}, u \in \mathcal{U}$. We define a metric on our space $d : \mathcal{X} \times \mathcal{X} \to [0, \infty)$:

$$d(x, y) = \mathbb{1}\{\sigma_x \neq \sigma_y\} + \|s_x - s_y\|_{l^1} + \|u_x - u_y\|_2, \quad (10)$$

which is quickly verified as a well-defined metric.

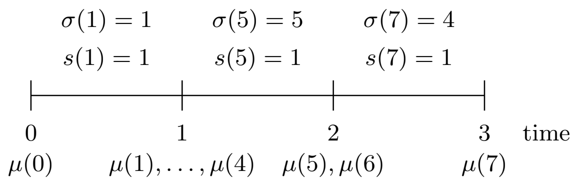

Figure 1: Given σ and s as defined in equations (7) and (8), the diagram illustrates the modes assigned to each interval via $\sigma(i)$ and their time length via $s(i)$.

i	μ	m	n
0	0	0	0
1	1	1	4
2	1	1	4
3	1	1	4
4	1	1	4
5	2	5	6
6	2	5	6
7	3	7	7

t	κ	π
0	1	1
0.5	1	1
1	1	1
1.5	5	5
2	5	5
2.5	7	4
3	7	4
3.5	∞	*NF*

Table 1: Given σ and s as defined in equations (7) and (8), the values of the functions μ, m, and n is illustrated in the table on the left and the values of the functions κ and π at various instants is illustrated in the table on the right.

Finally, given a $\xi \in \mathcal{X}$ and an initial condition, x_0, the corresponding trajectory, $x^{(\xi)}(t)$, is defined by:

$$\begin{aligned} \dot{x}(t) &= f_{\pi(t)}\big(x(t), u(t)\big) \quad \forall t \in (0, \mu_f] \\ x(0) &= x_0. \end{aligned} \quad (11)$$

Note, that zero entries interspersed between nonzero entries in s have no affect on the solution of the differential equation.

We define the cost $J : \mathcal{X} \to \mathbb{R}$ on the state trajectory and the continuous input as:

$$J(\xi) = \int_0^{\mu_f} L\big(x^{(\xi)}(t), u(t)\big)dt + \phi\big(x^{(\xi)}(\mu_f)\big). \quad (12)$$

We also require that for all time the state is constrained to a set described as:

$$x(t) \in \left\{x \in \mathbb{R}^n \,\middle|\, h_j(x) \leq 0,\ j = 1, \dots, N_c\right\}. \quad (13)$$

Let $\mathcal{J} = \{1, \dots, N_c\}$ denote the set of constraints. Using a standard reduction technique, we compactly describe all the constraints by defining a new function ψ:

$$\psi(\xi) = \max_{j \in \mathcal{J}} \max_{t \in [0, \mu_f]} h_j\big(x^{(\xi)}(t)\big). \quad (14)$$

Note that $\psi(\xi) \leq 0$ if and only if for all time t, the constraint, equation (13), is satisfied. With these definitions, we can state the hybrid optimal control problem.

Switched Hybrid Optimal Control Problem:

$$\begin{aligned} &\min_{\xi \in \mathcal{X}} J(\xi) \\ \text{s.t.} \quad &\psi(\xi) \leq 0 \end{aligned} \quad (15)$$

We make the following assumptions on the dynamics, cost, and constraints:

Assumption 1. *The functions L and f_q are Lipschitz and differentiable in x and u for all q in $\mathcal{Q}$. In addition, the derivatives of these functions with respective to x and u are also Lipschitz. Since this set of functions is finite, we define $K_1 \in (0, \infty)$ large enough to be the Lipschitz constant for these functions and their derivatives.*

Assumption 2. *The functions ϕ and h_j are Lipschitz and differentiable in x for all j in $\mathcal{J}$. In addition, the derivatives of these functions with respect to x are also Lipschitz. Since this set of functions is finite, we define $K_2 \in (0, \infty)$ large enough to be the Lipschitz constant for these functions and their derivatives.*

Assumption 3. *The input $u \in \mathcal{U}$ is continuous from the right.*

Observe that the cost function defined in equation (12) is general enough to capture both a running and a final cost. This definition captures most interesting cost functions. Assumption 1 is sufficient to ensure the existence, uniqueness, and boundedness of the solution to our differential equation (11). Assumption 2 is a standard assumption on the final cost and constraints and is used to prove the convergence properties of the algorithm defined in the next section. Though Assumption 3 may seem exacting, under reasonable conditions on the cost function one can in fact guarantee that the optimal input is Lipschitz continuous for all time [23]. Next, we develop an algorithm to calculate a hybrid optimal control.

3. OPTIMIZATION ALGORITHM

In this section, we present our optimization algorithm to determine a numerical solution to the Switched Hybrid Optimal Control Problem. We leave the calculations and proof of the convergence of our algorithm to the next section. Before we consider the algorithm explicitly, we describe a property that any optimization algorithm should satisfy.

An algorithm $a : \mathcal{X} \to \mathcal{X}$ takes an initial point $\xi_0 \in \mathcal{X}$ and generates a sequence of feasible points by letting $\xi_{j+1} = a(\xi_j)$ for $j = 0, 1, \dots$. We want to find suitable conditions under which the sequence of points generated by algorithm a converge to a local minimum of our optimization problem. It is important to note that simply requiring that an algorithm a has a descent property, i.e. $J(a(\xi)) < J(\xi)$, is not sufficient to ensure the convergence of the sequence to a local minimum. However, if the algorithm has the *sufficient descent* property, important convergence properties follow.

Before we define the sufficient descent property, we must first define a non-positive function, $\theta : \mathcal{X} \to (-\infty, 0]$, called the optimality function. Also denote the set of points at which θ vanishes by:

$$QS = \left\{\xi \in \mathcal{X} \,\middle|\, \theta(\xi) = 0\right\}. \quad (16)$$

The elements of QS, which we refer to as *points that satisfy our optimality condition*, are points of interest.

DEFINITION 1 (SUFFICIENT DESCENT). *An algorithm $a : \mathcal{X} \to \mathcal{X}$ is said to have the sufficient descent property with respect to θ if for all ξ in $\mathcal{X}$ with $\theta(\xi) < 0$, there exists a $\delta_\xi > 0$ and a neighborhood of ξ, $U_\xi \subset \mathcal{X}$, such that given the cost function J and the feasible set $\mathcal{F}$ the following inequality is satisfied:*

$$J\big(a(\xi')\big) - J(\xi') \leq -\delta_\xi, \quad \forall \xi' \in U_\xi \cap \mathcal{F}.$$

To make the utility of this property explicit, suppose we defined $\theta(\xi)$ to be zero whenever $J(\xi)$ was at a local minimum. If we used the function θ as a stopping criteria for an algorithm, a, that satisfies the sufficient descent property with respect to θ, then we generate a sequence of points that progressively approach an element of QS, or a local minima of J, as follows: given $j = 0$ and ξ_j in $\mathcal{X}$, if $\theta(\xi_j) = 0$ stop, or else let $\xi_{j+1} = a(\xi_j)$, and repeat. In fact, we can prove that this sequence approaches an element of QS.

THEOREM 1 (POLAK [17] THEOREM 1.2.8). *Suppose $J : \mathcal{X} \to \mathbb{R}$ is continuous and the constraint set is closed. If an algorithm a satisfies the sufficient descent property with respect to an optimality function θ, then, either the sequence $\{\xi_j\}$ constructed by algorithm a is finite and its last element belongs to QS and is feasible or else it is infinite and every accumulation point of $\{\xi_j\}_{j=1}^\infty$ belongs to QS and is feasible.*

Remark: Theorem 1 as originally stated is for an upper semi-continuous θ and a convex constraint set. However, the result as stated here can be proved without requiring either of these properties.

Returning from this digression, observe that our cost function is continuous (proved in Proposition 3 below) and our constraint set is closed since ψ is continuous (proved in Proposition 4 below). Our goal is to apply this previous theorem to show the convergence of our soon to be constructed algorithm. In particular, we must design an algorithm with the sufficient descent property with respect to an optimality function whose vanishing points include solutions to our desired optimal control problem.

We propose a bi-level hierarchical algorithm that divides the problem into two nonlinear constrained optimization problems one continuous and the other discrete:

Bi-Level Optimization Scheme

Stage 1: Given a jump sequence, σ, calculate the optimal jump time sequence, s, and the optimal continuous control u.

Stage 2: Calculate a new sequence, $\tilde{\sigma}$, that is the result of the insertion of a new jump into the original sequence σ. Repeat Stage 1 using $\tilde{\sigma}$.

Given this procedure, a point $\xi = (\sigma, s, u) \in \mathcal{X}$ is considered optimal if (s, u) is a locally optimal solution for Stage 1 and if there exists no feasible mode insertion which reduces the cost. Observe that Stage 1 can be transformed into a standard optimal control problem where both the control and initial condition are optimization variables (Section 5.1 describes this transformation and see Section 4.1.2 of [17], for more details). Let $\hat{a} : \mathcal{S}\times\mathcal{U} \to \mathcal{S}\times\mathcal{U}$ be an algorithm that solves Stage 1. We make an additional assumption about $\hat{a}$ which we justify in the next section:

ASSUMPTION 4. *The algorithm, $\hat{a}$ has the sufficient descent property with respect to θ, for a fixed σ, as described in Definition 1.*

This assumption states that our inner algorithm has the sufficient descent property with respect to the same optimality function which we define for our entire algorithm, a. Though this may seem like a rigid assumption, we describe its necessity and reasonableness in the next section.

Finally, we can construct our optimality function $\theta : \mathcal{X} \to (-\infty, 0]$. Since we would like the vanishing points of our optimality functions to include solutions to our desired optimal control problem, we require that if there are no feasible mode insertions which lower the cost then $\theta(\xi) = 0$. To define this function explicitly, we must describe how a feasible mode insertion looks. Given $\xi \in \mathcal{X}$, consider the insertion of a mode, $\hat{\alpha}$ and control, $\hat{u}$, at time $\hat{t}$. This insertion is characterized by $\eta = (\hat{\alpha}, \hat{t}, \hat{u}) \in \mathcal{Q} \times [0, \mu_f] \times \mathcal{B}^m(0, M)$, where $\mathcal{B}^m(0, M) = \{u \in \mathbb{R}^m \mid \|u\| \leq M\}$. Suppose that the insertion is for a duration of length $\lambda \geq 0$ as illustrated in Figure 2.

Let $\rho^{(\eta)} : [0, \infty) \to \mathcal{X}$ denote the function that describes this type of insertion ($\rho^{(\eta)}$ is defined more explicitly in Definition 2 and its argument denotes the length of the insertion). In order to determine if the cost decreases due to this type of insertion, we want to evaluate a first order approximation of J with respect to the variation ρ:

$$\left.\frac{dJ\big(\rho^{(\eta)}(\lambda)\big)}{d\lambda}\right|_{\lambda=0} = \lim_{\lambda \downarrow 0} \frac{J\big(\rho^{(\eta)}(\lambda)\big) - J(\xi)}{\lambda} \tag{17}$$

Observe that if this derivative is negative and the mode insertion leaves the trajectory feasible, then it is possible to decrease the overall cost (this is proved in Theorem 2). In order to ensure that the mode insertion results in a feasible trajectory, we must also consider:

$$\left.\frac{d\psi\big(\rho^{(\eta)}(\lambda)\big)}{d\lambda}\right|_{\lambda=0} = \lim_{\lambda \downarrow 0} \frac{\psi\big(\rho^{(\eta)}(\lambda)\big) - \psi(\xi)}{\lambda} \tag{18}$$

Using these results, we define the optimality function:

$$\theta(\xi) = \min_{\eta \in Q \times [0,\mu_f] \times \mathcal{B}^m(0,M)} \max\left\{ \left.\frac{dJ(\rho^{(\eta)}(\lambda))}{d\lambda}\right|_{\lambda=0}, \psi(\xi) + \left.\frac{d\psi(\rho^{(\eta)}(\lambda))}{d\lambda}\right|_{\lambda=0} \right\}. \tag{19}$$

Note that $\theta(\xi) \leq 0$ since at any time $t \in [0, \mu_f]$ inserting the same mode and the same continuous input is a feasible solution and the derivatives of the cost and constraint are zero for this choice. Observe that as required if there are no feasible mode insertions which reduce the cost then $\theta(\xi) = 0$ (this is proved in Theorem 2). We construct an algorithm, $a : \mathcal{X} \to \mathcal{X}$, to solve the Switched Hybrid Optimal Control Problem:

The main result of our paper is that Algorithm 1 converges to a point that satisfies our optimality condition, which is proved in Theorem 3.

4. ALGORITHM ANALYSIS

In this section, we describe in detail the pieces that are required to show that Algorithm 1 converges to a point that satisfies our optimality condition. The section is divided into a piece where we prove the continuity of the cost and constraint and a piece which proves the convergence of our algorithm. The proofs of these various propositions and theorems can be found in the technical report [12] due to lack of space.

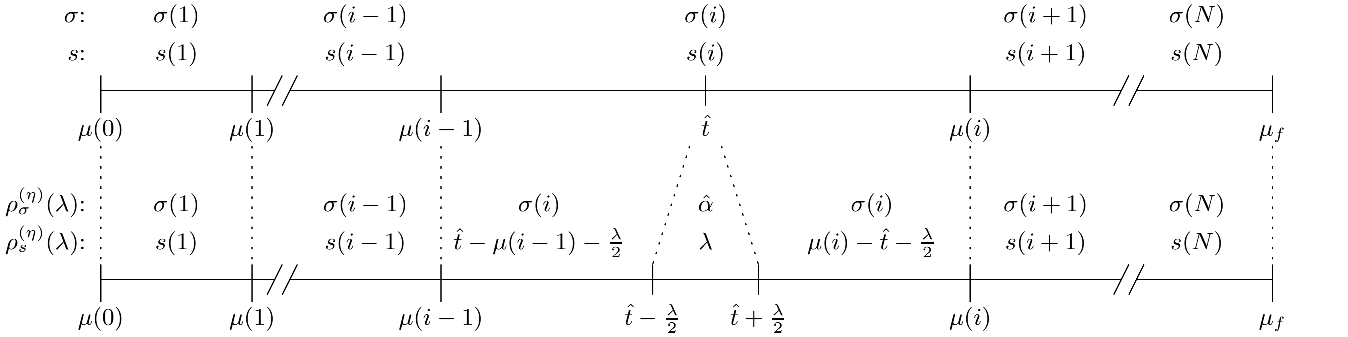

Figure 2: Diagram illustrating the transition from σ to $\rho_\sigma^{(\eta)}(\lambda)$ and s to $\rho_s^{(\eta)}(\lambda)$. The top line is the original definition of σ and s, and the bottom line shows the result for $\lambda > 0$.

Algorithm 1 Optimization Algorithm for the Switched Hybrid Optimal Control Problem

---

Data: $\xi_0 \in X$

Step 0. Let $(s_1, u_1) = \hat{a}(s_0, u_0)$, $\sigma_1 = \sigma_0$, define $\xi_1 = (\sigma_1, s_1, u_1)$.

Step 1. Set $j = 1$.

Step 2. If $\theta(\xi_j) = 0$ stop.

Step 3. $\xi_{j+1} = a(\xi_j)$ where a is defined as follows:

a. $\hat{\eta} = (\hat{\alpha}, \hat{t}, \hat{u})$ is the argument that minimizes $\theta(\xi_j)$. Let $\tilde{\sigma}_j$ be the modal sequence obtained by the insertion of $\hat{\alpha}$ at $\hat{t}$.

b. Using the new mode sequence, $\tilde{\sigma}$, let $(s_{j+1}, u_{j+1}) = \hat{a}(s_{j+1}, u_{j+1})$ be the solution to Stage 1.

c. Define $\sigma_{j+1} = \tilde{\sigma}_j$, $\xi_{j+1} = a(\xi_j) = (\sigma_{j+1}, s_{j+1}, u_{j+1})$.

Step 4. Replace j by $j + 1$ and go to step 2.

4.1 Continuity of the Cost and Constraints

In order to apply Theorem 1, we must first check that the cost function, equation (12), under Assumptions 1 and 2 is continuous. We prove the continuity of the cost function by taking a sequence, $(\xi_j)_{j=1}^{\infty}$ converging to limit ξ, in our optimization space, and proving that the corresponding sequence of trajectories $(x_j(t))_{j=1}^{\infty}$ converge to trajectory $x(t)$ corresponding to ξ. This result proves the sequential continuity of our cost function, which implies continuity since $\mathcal{X}$ is a metric space.

Throughout this subsection we simplify the notation used for the functions μ, κ, π, and μ_f. Given $\xi_j = (\sigma_j, s_j, u_j) \in \mathcal{X}$, we define $\mu_j(i) = \mu(i; s_j)$, $\kappa_j(i) = \kappa(i; s_j)$, $\pi_j(i) = \pi(i; s_j)$, and $\mu_{f,j} = \mu_f(s_j)$. As usual, when the choice of $s \in \mathcal{S}$ is clear in context we use our standard notation.

Proposition 1. *Let $(\xi_j := (\sigma_j, s_j, u_j))_{j=1}^{\infty}$ be a convergent sequence in the optimization space, $\mathcal{X}$, and let $\xi := (\sigma, s, u)$ be its limit. Let $(x_j(t))_{j=1}^{\infty}$ be the corresponding trajectories (defined using Equation 11) associated with each ξ_j, with common initial condition x_0. The sequence $(x_j(t))_{j=1}^{\infty}$ converges pointwise to the trajectory $x(t)$ associated with ξ, for all t in $[0, \infty)$ with initial condition x_0.*

In fact, we have a stronger condition on the convergence.

Proposition 2. *Let $(\xi_j := (\sigma_j, s_j, u_j))_{j=1}^{\infty}$ be a convergent sequence in our optimization space, $\mathcal{X}$, and let $\xi := (\sigma, s, u)$ be its limit. Let $(x_j(t))_{j=1}^{\infty}$ be the corresponding trajectories (defined using Equation 11) associated with each ξ_j, with common initial condition x_0. The sequence $(x_j)_{j=1}^{\infty}$ converges uniformly to the trajectory x associated with ξ, on $[0, \sum_{i=1}^{\infty} s(i)]$ with initial condition x_0.*

Given Proposition 2, we can now check the continuity of the cost function.

Proposition 3. *The function J as defined in equation (12) is continuous.*

Finally, we must check that $\{\xi \in \mathcal{X} \mid \psi(\xi) \le 0\}$ is a closed set in order to apply Theorem 1. Since we are employing inequality constraints, showing that ψ is continuous gives us the required result.

Proposition 4. *The function ψ as defined in equation (14) is continuous.*

4.2 Optimality Function

In this section, we prove the convergence of Algorithm 1. Our algorithm works by inserting a new mode, $\hat{\alpha}$, in a small interval of length $\lambda \ge 0$ centered at a time, $\hat{t}$, with input $\hat{u}$. We begin by defining this type of insertion.

Definition 2. *Given $\xi = (\sigma, s, u) \in \mathcal{X}$ and $\eta = (\hat{\alpha}, \hat{t}, \hat{u}) \in \mathcal{Q} \times [0, \mu_f] \times \mathcal{B}^m(0, M)$, we define the function $\rho^{(\eta)} : [0, \infty) \to \mathcal{X}$ as the perturbation of ξ after the insertion of mode $\hat{\alpha}$, at time $\hat{t}$ using $\hat{u}$ as the control, for a time interval of length λ. Let $\bar{\lambda} = \min_{\{i : |\mu(i) - \hat{t}| > 0\}} \frac{1}{2} \left|\mu(i) - \hat{t}\right|$, then we write $\rho^{(\eta)}(\lambda) = \left(\rho_\sigma^{(\eta)}(\lambda), \rho_s^{(\eta)}(\lambda), \rho_u^{(\eta)}(\lambda)\right)$, whenever $\lambda \in [0, \bar{\lambda}]$,*

$$\rho_\sigma^{(\eta)}(\lambda) = \begin{cases} (\hat{\alpha}, \sigma(1), \sigma(2), \ldots) & \text{if } \hat{t} = 0 \\ (\sigma(1), \ldots, \pi(\hat{t} - \frac{\bar{\lambda}}{2}), \hat{\alpha}, \ldots) & \text{if } \hat{t} = \mu_f \\ (\sigma(1), \ldots, \pi(\hat{t} - \frac{\bar{\lambda}}{2}), \hat{\alpha}, \\ \quad \pi(\hat{t} + \frac{\bar{\lambda}}{2}), \ldots) & \text{if } \hat{t} \ne \mu(i) \\ (\sigma(1), \ldots, \pi(\hat{t} - \frac{\bar{\lambda}}{2}), \\ \quad m(\kappa(\hat{t})) + 1, \ldots, n(\kappa(\hat{t})), \\ \quad \hat{\alpha}, \pi(\hat{t} + \frac{\bar{\lambda}}{2}), \ldots) & \text{if } \hat{t} = \mu(i) \end{cases} \tag{20}$$

$$\rho_s^{(\eta)}(\lambda) = \begin{cases} (\lambda, s(1)-\lambda, s(2), \dots) & \text{if } \hat{t}=0 \\ (s(1), \dots, s(\kappa(\mu_f)) - \lambda, \lambda, 0, \dots) & \text{if } \hat{t}=\mu_f \\ (s(1), \dots, \hat{t}-\frac{\lambda}{2} - \mu(\kappa(\hat{t}-\frac{\bar{\lambda}}{2})-1), & \\ \quad \lambda, \mu(\kappa(\hat{t}+\frac{\bar{\lambda}}{2})) - \hat{t} - \frac{\lambda}{2}, \dots) & \text{if } \hat{t} \neq \mu(i) \\ (s(1), \dots, \hat{t}-\frac{\lambda}{2} - \mu(\kappa(\hat{t}-\frac{\bar{\lambda}}{2})-1), & \\ \quad s(m(i)+1), \dots, s(n(i)), & \\ \quad \lambda, \mu(\kappa(\hat{t}+\frac{\bar{\lambda}}{2})) - \hat{t} - \frac{\lambda}{2}, \dots) & \text{if } \hat{t} = \mu(i) \end{cases} \tag{21}$$

$$\rho_u^{(\eta)}(\lambda) = \begin{cases} u(t) + \big(\hat{u} - u(t)\big)\mathbb{1}_{[0,\lambda]}(t) & \text{if } \hat{t}=0 \\ u(t) + \big(\hat{u} - u(t)\big)\mathbb{1}_{[\mu_f-\lambda,\mu_f]}(t) & \text{if } \hat{t}=\mu_f \\ u(t) + \big(\hat{u} - u(t)\big)\mathbb{1}_{[\hat{t}-\frac{\lambda}{2},\hat{t}+\frac{\lambda}{2}]}(t) & \text{otherwise} \end{cases}, \tag{22}$$

and $\rho^{(\eta)}(\lambda) = \rho^{(\eta)}(\bar{\lambda})$ *whenever* $\lambda > \bar{\lambda}$.

Note that ρ, in addition to being a function of λ and η, is also a function of ξ, but we do not make this dependence explicit for notational convenience. If the dependence of ρ with respect to ξ is not clear, then we make the declaration explicit. Importantly, observe that $\rho_u^{(\eta)}$ is a needle variation (or strong variation) of the control $u(t)$ (as defined in Chapter 2, Section 13 of [18]). Figure 2 illustrates a pair (σ, s) after they are modified by the function $\rho^{(\eta)}$.

Proposition 5. *Given* $\eta \in \mathcal{Q} \times [0, \mu_f] \times \mathcal{B}^m(0, M)$*, the function* $\rho^{(\eta)}$ *is continuous.*

We need this property in order to understand the variation of the cost with respect to this insertion. We begin by studying the variation of the trajectory, $x^{(\rho(\lambda))}$, as λ changes. Note that $x^{(\xi)}(t) = x^{(\rho(0))}(t)$ for each $t \geq 0$, so a first order approximation of the trajectory is characterized by the directional derivative of $x^{(\rho(\lambda))}$ at $\lambda = 0$. To reduce the number of cases we need to consider in the future propositions, we define for a given $x : [0, \infty) \to \mathbb{R}^n$, $u : [0, \infty) \to \mathbb{R}^m$, and $\eta = (\hat{\alpha}, \hat{t}, \hat{u})$:

$$\Delta f(x, u, \eta) = \begin{cases} f_{\hat{\alpha}}(x(\hat{t}), \hat{u}) - f_{\pi(\hat{t}+\bar{\lambda})}(x(\hat{t}), u(\hat{t})) & \text{if } \hat{t}=0 \\ f_{\hat{\alpha}}(x(\hat{t}), \hat{u}) - f_{\pi(\hat{t}-\bar{\lambda})}(x(\hat{t}), u(\hat{t})) & \text{if } \hat{t}=\mu_f \\ f_{\hat{\alpha}}(x(\hat{t}), \hat{u}) + & \\ \quad -\frac{1}{2} f_{\pi(\hat{t}+\bar{\lambda})}(x(\hat{t}), u(\hat{t})) + & \\ \quad -\frac{1}{2} f_{\pi(\hat{t}-\bar{\lambda})}(x(\hat{t}), u(\hat{t})) & \text{otherwise} \end{cases} \tag{23}$$

where $\bar{\lambda} = \min_{\{i : |\mu(i) - \hat{t}| > 0\}} \frac{1}{2} \left|\mu(i) - \hat{t}\right|$. Observe that $\pi(\hat{t} + \bar{\lambda}) = \pi(\hat{t} - \bar{\lambda})$ whenever $\hat{t} \notin \{\mu(i)\}_{i \in \mathbb{N}}$. With an abuse of notation, we denote $x^{(\rho(\lambda))}$ by $x^{(\lambda)}$.

Proposition 6. *The directional derivative of* $x^{(\lambda)}$ *for* λ *positive, evaluated at zero, is:*

$$\left.\frac{dx^{(\lambda)}}{d\lambda}\right|_{\lambda=0}(t) = \begin{cases} \Phi(t, \hat{t}) \Delta f(x^{(\xi)}, u, \eta) & \text{if } t \in [\hat{t}, \mu_f] \\ 0 & \text{otherwise} \end{cases}, \tag{24}$$

where $\Phi : [0, \infty) \times [0, \infty) \to \mathbb{R}^{n \times n}$ *is the solution of the matrix differential equation:*

$$\frac{dX(t, \hat{t})}{dt} = \frac{\partial f_{\pi(t)}}{\partial x}\big(x^{(\xi)}(t), u(t)\big)\, X(t, \hat{t}), \quad X(\hat{t}, \hat{t}) = I. \tag{25}$$

Given this variation of the state trajectory, we can now consider variations of the cost and constraint functions, which allows us to define our optimality function θ in a manner that guarantees if there are no feasible mode insertions which lower the cost then $\theta(\xi) = 0$.

Proposition 7. *Let* J *be the cost function as defined in equation* (12)*. Then the directional derivative of* $J(\rho^{(\eta)}(\lambda))$ *evaluated at* $\lambda = 0$ *is*

$$\left.\frac{dJ(\rho^{(\eta)}(\lambda))}{d\lambda}\right|_{\lambda=0} = \left(p^{(\xi)}(\hat{t})\right)^T \Delta f(x^{(\xi)}, u, \eta) + \\ + \left[\hat{u} - u(\hat{t})\right]^T \frac{\partial L}{\partial u}\big(x^{(\xi)}(\hat{t}), u(\hat{t})\big) \tag{26}$$

where $p^{(\xi)}$*, can be identified with the costate, and is the solution to the following differential equation*

$$\dot{p}(t) = -\frac{\partial f_{\pi(t)}}{\partial x}^T \big(x^{(\xi)}(t), u(t)\big)\, p(t) - \frac{\partial L}{\partial x}\big(x^{(\xi)}(t), u(t)\big) \\ p(\mu_f) = \frac{\partial \phi}{\partial x}\big(x^{(\xi)}(\mu_f)\big). \tag{27}$$

In order to define our optimality function, we must also consider variations of the constraint function after the mode insertion procedure.

Proposition 8. *Let* ψ *be the constraint function defined in* (14)*. The directional derivative of* $\psi(\rho^{(\eta)}(\lambda))$ *evaluated at* $\lambda = 0$ *is*

$$\left.\frac{d\psi(\rho^{(\eta)}(\lambda))}{d\lambda}\right|_{\lambda=0} = \max_{j \in \hat{\mathcal{J}}(\xi)} \max_{t \in \hat{\mathcal{T}}_j(\xi)} \frac{\partial h_j}{\partial x}\big(x^{(\xi)}(t)\big) \left.\frac{dx^{(\lambda)}}{d\lambda}\right|_{\lambda=0}(t) \tag{28}$$

where

$$\hat{\mathcal{J}}(\xi) = \left\{ j \in \{1, \dots, R\} \;\middle|\; \max_{t \in \mathcal{T}_j} h_j\big(x^{(\xi)}(t)\big) = \psi(\xi) \right\} \tag{29}$$

$$\hat{\mathcal{T}}_j(\xi) = \left\{ t \in [0, \mu_f] \;\middle|\; h_j\big(x^{(\xi)}(t)\big) = \max_{\bar{t} \in [0, \mu_f]} h_j\big(x^{(\xi)}(\bar{t})\big) \right\} \tag{30}$$

Now we can prove that if $\theta(\xi)$ as defined in equation (19) is less than zero, then there exists a feasible mode insertion which reduces the overall cost (i.e. our optimality function captures the points of interest).

Theorem 2. *Consider the function* θ *defined in equation* (19)*. Let* $\xi \in \mathcal{X}$ *and* $\eta = (\hat{\alpha}, \hat{t}, \hat{u}) \in \mathcal{Q} \times [0, \mu_f] \times \mathcal{B}^m(0, M)$ *be the argument which minimizes* $\theta(\xi)$*. If* $\theta(\xi) < 0$*, then there exists* $\hat{\lambda} > 0$ *such that, for each* $\lambda \in (0, \hat{\lambda}]$*,* $J(\rho^{(\eta)}(\lambda)) \leq J(\xi)$ *and* $\psi(\rho^{(\eta)}(\lambda)) \leq 0$.

This result proves that the vanishing points of our optimality function for Algorithm 1 contain solutions to our optimal control problem. We now address the validity of Assumption 4. First, recall that ρ is a needle variation; therefore, as a result of the previous theorem, if $\theta(\xi) < 0$ then we are not at a minimum in the sense of Pontryagin [16]. Unfortunately, numerical methods for optimization cannot implement these types of variations since that task would require the approximation of arbitrarily narrow discontinuous functions. This means that any practical algorithm using a numerical method would find minima that do not necessarily coincide with the minima prescribed by our θ function. If we were uninterested in constructing a practical algorithm, then Assumption 4 would be trivially satisfied by any of the theoretical algorithms proposed by Pontryagin.

Fortunately, we can construct a practical algorithm using the following proposition.

Proposition 9. *If the vector fields $\{f_q\}_{q\in\mathcal{Q}}$ are affine with respect to the control and the running cost L is convex with respect to the control, then the optimality condition calculated via vector-space variations (variations that take the form of directional derivatives) and the optimality condition calculated via needle variations are equivalent.*

Under the hypotheses of the proposition above, there are numerous algorithms that satisfy Assumption 4, among them the algorithms described in Section 4.5 in [17]. Importantly, we can transform any nonlinear vector field into a new vector field that is affine with respect to its control using the following transformation:

$$\begin{pmatrix}\dot{x}(t)\\ \dot{z}(t)\end{pmatrix} = \begin{pmatrix} f_{\pi(t)}(x(t),z(t)) \\ v(t)\end{pmatrix}, \tag{31}$$

where $(x(t),z(t))^T$ in $\mathbb{R}^{m+n}$ become the new state variables, and $v(t)\in\mathbb{R}^m$ becomes the new control input. After the transformation those same algorithms would guarantee the validity of Assumption 4.

Finally, we can show that Algorithm 1 has the sufficient descent property with respect to our optimality function.

Theorem 3. *Algorithm $a:\mathcal{X}\to\mathcal{X}$, as defined in Algorithm 1 has the sufficient descent property with respect to the function $\theta:\mathcal{X}\to\mathbb{R}$.*

Using this fact and Theorem 1, we have that our algorithm converges to points that satisfy our optimality condition as desired.

5. IMPLEMENTATION

In this section, we describe the numerical implementation of Algorithm 1. First, we describe how to reformulate Stage 1 in the Bi-Level Optimization Scheme via a transformation into a canonical optimal control problem. Second, we discuss the implementation of our optimality function.

5.1 Transcription into Canonical Form

Given a ξ in $\mathcal{X}$, we discuss how to solve Stage 1 in the Bi-Level Optimization Scheme by transforming our problem into one where the optimization over the switching instances and continuous control becomes an optimization over the initial condition and the continuous control. There exist algorithms to perform optimization directly over the switching times, but we consider optimization over the initial condition and continuous control since it has been studied more extensively in the literature [3, 17].

Recall by assumption that for any ξ in $\mathcal{X}$, there exists a finite N such that for all $i>N$, $\sigma(i)=NF$. We introduce functions $\gamma_k:[0,1]\to\mathbb{R}$ and $z_k:[0,1]\to\mathbb{R}^n$ for $k=1,\dots,N$ such that:

$$\dot{\gamma}_k(t) = s(k)L(z_k(t),\bar{u}_k(t)) \tag{32}$$
$$\gamma_k(0)=0 \tag{33}$$

and

$$\dot{z}_k(t) = s(k)f_{\sigma(k)}(z_k(t),\bar{u}_k(t)) \tag{34}$$
$$z_k(0)=z_{k-1}(1) \tag{35}$$

where, with an abuse of notation, we assume $z_0(1)=x_0$ and $\bar{u}_k(t)=u\big(t\cdot s(k)+\mu(k-1)\big)$ for all t in $[0,1]$ and $k=1,\dots,N$. It is clear from these definitions that $z_k(t)=x^{(\xi)}\big(t\cdot s(k)+\mu(k-1)\big)$ for $k=1,\dots,N$, and

$$\sum_{k=1}^{N}\gamma_k(1)=\int_0^{\mu_f} L\big(x^{(\xi)}(t),u(t)\big)dt. \tag{36}$$

Given these definitions, we construct new state variables, $\omega_k:[0,1]\to\mathbb{R}^{n+2}$ and new flow fields, $\beta_k:\mathbb{R}^{n+2}\times\mathbb{R}^m\to\mathbb{R}^{n+2}$ for $k=1,\dots,N$ such that:

$$\omega_k(t)=\begin{pmatrix} z_k(t)\\ s(k)\\ \gamma_k(t)\end{pmatrix} \tag{37}$$

$$\beta_k(\omega_k(t),\bar{u}_k(t))=\begin{pmatrix} s(k)f_{\sigma(k)}\big(z_k(t),\bar{u}_k(t)\big)\\ 0\\ s(k)L\big(z_k(t),\bar{u}_k(t)\big)\end{pmatrix}. \tag{38}$$

Then we define a new optimal control problem whose solution is a transformed version of the solution to the problem we are interested in solving

$$\min_{\substack{\{s(k)\in[0,\infty)\mid k=1,\dots,N\}\\ \{\bar{u}_k\in\mathcal{U}\mid k=1,\dots,N\}}} \Big(\sum_{k=1}^{N}\gamma_k(1)+\phi(z_N(1))\Big) \tag{39}$$

subject to:

$$\dot{\omega}_k(t)=\beta_k\big(\omega_k(t),\bar{u}_k(t)\big),\quad \forall k=1,\dots,N \tag{40}$$

$$\omega_k(0)=\begin{pmatrix} z_{k-1}(1)\\ s(k)\\ 0\end{pmatrix},\quad \forall k=1,\dots,N \tag{41}$$

$$h_j\big(z_k(t)\big)\le 0,\quad \forall\, j\in\mathcal{J},\ \forall\, t\in[0,1],\ \forall k=1,\dots,N \tag{42}$$

where $k=1,\dots,N$, $z_0(1)=x_0$, and $\mathcal{J}$ denotes our index over the set of constraints. As desired, this problem minimizes over the initial conditions and continuous controls, rather than switching times and continuous controls. Importantly, the solution to this problem is tractable and equivalent to the solution of Stage 1.

5.2 Implementing the Optimality Function

In Algorithm 1, given a ξ in $\mathcal{X}$, we check to see if $\theta(\xi)=0$. If $\theta(\xi)<0$, we require the argument, η, which minimizes it. Unfortunately, since $\theta(\xi)$ is a nonconvex function calculating the minimum may be difficult. Fortunately, finding any value of η that makes $\theta(\xi)$ less than zero, provides us with a feasible mode insertion which reduces the overall cost. Thus Theorem 2 remains valid.

The reader may be concerned that determining any η requires solving a min-max problem. However, there exist two viable approaches to solving this type of problem. First, one can apply any min-max optimization algorithm presented in Section 2.6 of [17]. Second, one can transform any min-max problem into a constrained minimization problem by using the epigraph form transformation of the problem. In our implementation, we employ the Polak-He Algorithm from Section 2.6 of [17].

6. EXAMPLES

In this section, we apply Algorithm 1 to calculate an optimal control for two examples.

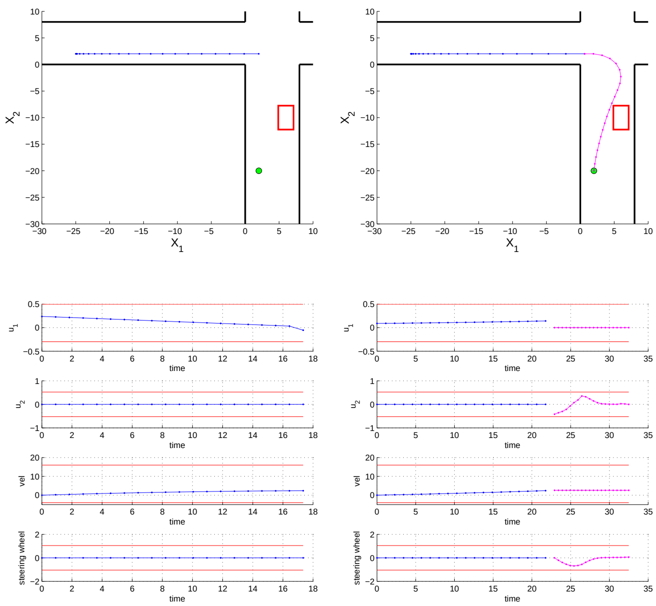

Figure 3: The top row indicates trajectories and the bottom row indicates control inputs, velocity, and steering wheel angle. Each iteration of the algorithm is drawn in a column and the iterations increase from left to right. Black lines represent the road in which the vehicle is constrained to move, and the red rectangle is a parked vehicle. The green circle represents the waypoint $\hat{w}$. In each plot, blue is associated with the *Forward* mode and magenta is associated with the *Turn* mode.

6.1 Switched System Vehicle Model following Waypoints

Consider a vehicle under the front wheel bicycle model with the equation of motion given as:

$$\begin{aligned}
\dot{x_1}(t) &= x_4(t)\cos\left(x_3(t)+x_5(t)\right)\\
\dot{x_2}(t) &= x_4(t)\sin\left(x_3(t)+x_5(t)\right)\\
\dot{x_3}(t) &= \frac{1}{b}x_4(t)\sin\left(x_5(t)\right)\\
\dot{x_4}(t) &= u_1(t)\\
\dot{x_5}(t) &= u_2(t)
\end{aligned} \tag{43}$$

where x_1, x_2 are the x, y Cartesian coordinates of the car, x_3 is the angular orientation of the car with respect to the x-axis, x_4 is the velocity of the car, and x_5 is the steering wheel angle. Also, $u_1(t) \in [-0.3, 0.5]$ corresponds to acceleration and $u_2(t) \in [-\pi/6, \pi/6]$ is the steering wheel rate of change. b is a fixed parameter describing the distance between the front and back wheels of the car. Though we assume nonzero length to describe the dynamics of car motion, we treat the car as a point in space.

The objective is to move from an initial position to a waypoint $\hat{w} \in \mathbb{R}^2$ (drawn in green in Figure 3) while avoiding obstacles and satisfying constraints. We constrain the position, x_1, x_2, to not hit other cars or sidewalks (drawn in red and black in Figure 3, respectively). We also constrain the velocity $x_4(t) \in [-4, 16]$ and the steering wheel angle $x_5(t) \in [-\pi/3, \pi/3]$. We consider a hybrid model with two modes: *Forward* mode, in which $u_2(t) = 0$, and *Turn* mode in which $u_1(t) = 0$. We define our cost function J as:

$$J(\xi) = \int_0^{\mu_f} \gamma\, u(t)^T u(t) dt + \left\| \begin{pmatrix} x_1(\mu_f) \\ x_2(\mu_f) \end{pmatrix} - \hat{w} \right\|^2, \tag{44}$$

where $\gamma \in \mathbb{R}_+$.

Figure 3 illustrates the trajectory and the control inputs of the car after each iteration of our algorithm. The car trajectory is initialized in *Forward* mode with only one nonzero modal sequence element. The inner optimization algorithm, $\hat{a}$, calculates an optimal control and final time. The trajectory is drawn in blue in the top-left plot in Figure 3, and the two controls are plotted in the bottom-left plot in Figure 3. Observe that the car arrives at a point where its x_1 state is the same as that of the waypoint (at time $\mu_f = 18.26$ seconds), which is the best the optimal control can do given the limited ability of the *Forward* mode. Next, θ is checked and found to be nonzero. Therefore a *Turn* mode is inserted at the end of the previous run, which results in a new modal sequence: first drive straight and then turn.

The optimal control and switching instants are then recalculated, with this new modal sequence held fixed. The optimal switching times are found to be $(\mu(i))_{i=1}^2 = (22.9120[s], 33.0082[s])$. The trajectory while in the *Forward* mode is drawn in blue in the top-right plot in Figure 3, and the *Turn* mode trajectory is drawn in magenta. The controls are plotted in similar colors in the bottom-right plot in Figure 3. Observe that the car is able to arrive at the waypoint. Finally, θ is checked again and falls within a predefined threshold, and the algorithm stops.

6.2 Quadrotor Helicopter Control

Next, we consider the optimal control of a quadrotor helicopter using a two dimensional simplified model. Letting x denote the position along the horizontal axes, z the height above the ground, and θ the roll angle of the helicopter, the equations of motion is given as:

$$\begin{aligned}
\ddot{x}(t) &= \frac{1}{M}\sin\left(\theta(t)\right)\sum_{k=1}^{3} T_k\\
\ddot{z}(t) &= \frac{1}{M}\cos\left(\theta(t)\right)\sum_{k=1}^{3} T_k - g\\
\ddot{\theta}(t) &= \frac{L}{I_y}(T_1 - T_3)
\end{aligned} \tag{45}$$

In the above T_1 and T_3 are the thrusts applied at the opposite ends of the quadrotor along the x axis, and T_2 is the sum of the thrusts of the other two rotors at the center of mass of the quadrotor. The parameters M, L, and I_y denote the mass, distance from center of mass of each of the rotors T_1 and T_3, and moment of inertia about y axis, respectively. The values of the parameters for this example are taken from the STARMAC experimental platform [14]. We hybridize the dynamics by introducing three modes: *Left*, *Right*, and *Up*. For the *Left* mode we set $T_1 = 0$, $T_2 = Mg$, and let $T_3 \in [0, 2]$. In the *Right* mode, we set $T_3 = 0$, $T_2 = Mg$, and let $T_1 \in [0, 2]$, and in the *Up* mode we set $T_1 = T_3 = 0$, and let $T_2 \in [0, 16]$. The objective is to reach a waypoint $\hat{w} \in \mathbb{R}^2$ (drawn in green in Figure 4) while avoiding obstacles (drawn in red in Figure 4), staying above the ground ($z = 0$) and maintaining a speed between zero and two. We define the

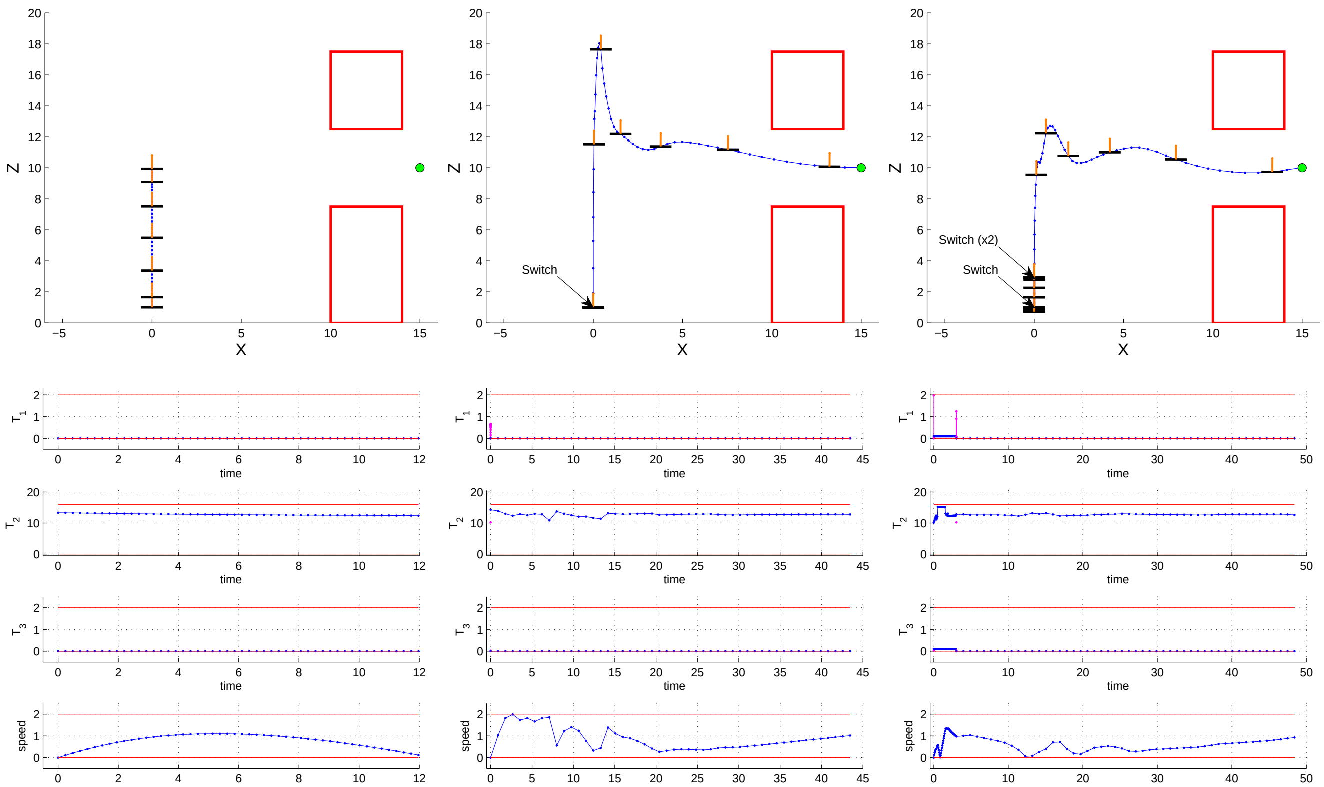

Figure 4: **The top row indicates trajectories while the bottom row indicates control inputs and speed. Each iteration of the algorithm is drawn in a column and the iterations increase from left to right. Constraints are drawn in red, the STARMAC quadrotor is drawn in black and the normal direction to the quadrotor frame is drawn in orange. The green circle represents the waypoint $\hat{w}$. In each plot, magenta is associated with the *Right* mode and blue is associated with the *Up* mode.**

cost function as

$$J(\xi) = \int_0^{\mu_f} \gamma\, \tilde{u}(t)^T \tilde{u}(t) dt + \left\| \begin{pmatrix} x(\mu_f) \\ z(\mu_f) \end{pmatrix} - \hat{w} \right\|^2 + \\ + \left\| \begin{pmatrix} \dot{x}(\mu_f) \\ \dot{z}(\mu_f) \end{pmatrix} \right\|^2 \quad (46)$$

where $\gamma \in \mathbb{R}_+$ and $\tilde{u}(t) = u(t) - u_{ss}$, with $u_{ss} = [0 \ Mg \ 0]$ being the steady-state input.

Figure 4 illustrates the trajectory and the control inputs of the quadrotor after each iteration of our algorithm. The algorithm is initialized in the *Up* mode and the optimal control and switching times for this initialization are calculated. The optimal trajectory and control are drawn in the left column and the optimal final time is $\mu_f = 12.22[s]$. Next, the algorithm reduces the cost by inserting a *Right* mode before the *Up* mode and the optimal control and switching times are calculated under this modal sequence. The optimal trajectory and control are drawn in the middle column and the optimal switching times are $(\mu(i))_{i=1}^2 = (5.48 \times 10^{-5}[s]\,, 44.35[s])$. Finally, the algorithm attempts to reduce the cost by inserting a *Right* mode during the *Up* mode and the optimal control and switching times are calculated under this modal sequence. The optimal trajectory and control are drawn in the right column and the optimal switching times are $(\mu(i))_{i=1}^4 = (1.06 \times 10^{-5}[s]\,, 3.04[s]\,, 1.58 \times 10^{-7}[s]\,, 49.33[s])$.

7. CONCLUSION

This paper presents an algorithm to numerically determine the optimal control for constrained nonlinear switched hybrid systems. For such systems, the control parameter has both a discrete component, the sequence of modes, and two continuous components, the duration of each mode and the continuous input. We develop a bi-level hierarchical algorithm that divides the problem into two subproblems.

At the lower level, we keep the modal sequence fixed and construct the optimal mode duration and optimal continuous input. At the higher level, we employ a single mode insertion technique to construct a new reduced cost sequence. We prove the convergence of this algorithm, and illustrate its utility of this algorithm on two numerical examples. In practice, the algorithm presented in this paper can be applied to any constrained nonlinear switched dynamical system to determine an optimal control.

8. ACKNOWLEDGMENTS

We would like to express our appreciation to Professor Elijah Polak for his valuable insight and to Saurabh Amin for providing his feedback. We would also like to thank Jeremy Gillula, Haomiao Huang, and Michael Vitus for their help with the STARMAC model.

This research work was sponsored by the Air Force Office of Scientific Research (AFOSR) under Agreement Number

FA9550-06-1-0312 and PRET Grant 18796-S2, the National Science Foundation (NSF) under grants 0703787 and 0724681, and the National Science and Engineering Research Canada (NSERC). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government. The U.S. Government is authorized to reproduce and distribute for Government purposes notwithstanding any copyright notation herein.

9. REFERENCES

[1] H. Axelsson, Y. Wardi, M. Egerstedt, and E. Verriest. Gradient Descent Approach to Optimal Mode Scheduling in Hybrid Dynamical Systems. *Journal of Optimization Theory and Applications*, 136(2):167–186, 2008.

[2] A. Bemporad, F. Borrelli, and M. Morari. Piecewise Linear Optimal Controllers for Hybrid Systems. In *American Control Conference, 2000. Proceedings of the 2000*, volume 2, 2000.

[3] D. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific Belmont, MA, 1995.

[4] F. Borrelli, M. Baotić, A. Bemporad, and M. Morari. Dynamic Programming for Constrained Optimal Control of Discrete-Time Linear Hybrid Systems. *Automatica*, pages 1709–1721, June 2005.

[5] M. Branicky, V. Borkar, and S. Mitter. A Unified Framework for Hybrid Control: Model and Optimal Control theory. *IEEE Transactions on Automatic Control*, 43(1):31–45, 1998.

[6] R. Brockett. Stabilization of Motor Networks. In *Decision and Control, 1995., Proceedings of the 34th IEEE Conference on*, volume 2, 1995.

[7] M. Egerstedt and Y. Wardi. Multi-Process Control Using Queuing Theory. In *IEEE Conference on Decision and Control*, volume 2, pages 1991–1996. IEEE; 1998, 2002.

[8] M. Egerstedt, Y. Wardi, and H. Axelsson. Transition-Time Optimization for Switched-Mode Dynamical Systems. *IEEE Transactions on Automatic Control*, 51(1):110–115, 2006.

[9] M. Egerstedt, Y. Wardi, and F. Delmotte. Optimal Control of Switching Times in Switched Dynamical Systems. In *Proceedings 42nd IEEE Conference on Decision and Control*, pages 2138– 2143, Dec. 2003.

[10] R. Ghosh and C. Tomlin. Hybrid System Models of Biological Cell Network Signaling and Differentiation. In *Proceedings of the Annual Allerton Conference on Communication Control and Computing*, volume 39, pages 942–951, 2001.

[11] A. Giua, C. Seatzu, and C. Van Der Mee. Optimal Control of Switched Autonomous Linear systems. In *IEEE Conference on Decision and Control*, volume 3, pages 2472–2477. IEEE; 1998, 2001.

[12] H. Gonzalez, R. Vasudevan, M. Kamgarpour, S. S. Sastry, R. Bajcsy, and C. Tomlin. A descent algorithm for the optimal control of constrained nonlinear switched dynamical systems: Appendix. Technical Report UCB/EECS-2010-9, EECS Department, University of California, Berkeley, Jan 2010.

[13] S. Hedlund and A. Rantzer. Optimal Control of Hybrid Systems. In *Proceedings of the 38th IEEE Conference on Decision and Control*, Phoenix, Arizona, Dec. 1999.

[14] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin. Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment. In *Proc. of the AIAA Guidance, Navigation, and Control Conference*. Citeseer, 2007.

[15] B. Lincoln and A. Rantzer. Optimizing Linear System Switching. In *IEEE Conference on Decision and Control*, volume 3, pages 2063–2068. IEEE; 1998, 2001.

[16] D. Mayne and E. Polak. First-Order Strong Variation Algorithms for Optimal Control. *Journal of Optimization Theory and Applications*, 16(3):277–301, 1975.

[17] E. Polak. *Optimization: Algorithms and Consistent Approximations*. Springer, 1997.

[18] L. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mishchenko. *The Mathematical Theory of Optimal Processes*. Interscience New York, 1962.

[19] A. Rantzer and M. Johansson. Piecewise Linear Quadratic Optimal Control. *IEEE Transactions on Automatic Control*, 45(4):629–637, 2000.

[20] H. Rehbinder and M. Sanfridson. Scheduling of a Limited Communication Channel for Optimal Control. *Automatica*, 40(3):491–500, 2004.

[21] M. Rinehart, M. Dahleh, D. Reed, and I. Kolmanovsky. Suboptimal Control of Switched Systems with an Application to the Disc Engine. *IEEE Transactions on Control Systems Technology*, 16(2):189–201, 2008.

[22] M. Shaikh and P. Caines. On the Optimal Control of Hybrid Systems: Optimization of Trajectories, Switching Times, and Location Schedules. *Lecture Notes in Computer Science*, 2623:466–481, 2003.

[23] R. Vinter. Optimal control, Systems & Control: Foundations & Applications, 2000.

[24] G. Walsh, H. Ye, L. Bushnell, et al. Stability Analysis of Networked Control Systems. *IEEE Transactions on Control Systems Technology*, 10(3):438–446, 2002.

[25] X. Xu and P. Antsaklis. Optimal Control of Switched Systems via Nonlinear Optimization Based on Direct Differentiation of Value Functions. *International Journal of Control, 75*, 16(17):1406–1426, 2002.

[26] X. Xu and P. Antsaklis. Results and Perspective on Computational Methods for Optimal Control of Switched Systems. *Lecture Notes in Computer Science*, 2623:540–556, 2003.

[27] W. Zhang, A. Abate, and J. Hu. Stabilization of Discrete-Time Switched Linear Systems: A Control-Lyapunov Function Approach. In R. Majumdar and P. Tabuada, editors, *Hybrid Systems: Computation and Control*, volume 5469 of *Lecture Notes in Computer Science*, pages 411–425. Springer, 2009.