Optimal Control of Switching Surfaces in Hybrid Dynamical Systems

M. Boccadoro*, Y. Wardi†‡, M. Egerstedt§, and E. Verriest

boccadoro@diei.unipg.it
Dipartimento di Ingegneria Elettronica e dell'Informazione
Università di Perugia
06125, Perugia – Italy

{ywardi,magnus,verriest}@ece.gatech.edu
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA

June 19, 2005

Abstract

This paper concerns an optimal control problem defined on a class of switched-mode hybrid dynamical systems. The system's mode is changed (switched) whenever the state variable crosses a certain surface in the state space, henceforth called a *switching surface*. These switching surfaces are parameterized by finite-dimensional vectors called the *switching parameters*. The optimal control problem is to minimize a cost functional, defined on the state trajectory, as a function of the switching parameters. The paper derives the gradient of the cost functional in a costate-based formula that reflects the special structure of hybrid systems. It then uses the formula in a gradient-descent algorithm for solving an obstacle-avoidance problem in robotics.

Keywords. Hybrid Systems, Switching Surfaces, Optimal Control, Gradient Descent, Obstacle Avoidance.

---

*The work of Boccadoro has been partially supported by MIUR under Grant PRIN 2003090090.
†Corresponding author.
‡The work of Wardi has been partly supported by a grant from the Georgia Tech Manufacturing Research Center.
§The work of Egerstedt has been partly supported by the National Science Foundation under Grant # 0237971 ECS NSF-CAREER, and by a grant from the Georgia Tech Manufacturing Research Center.

1 Introduction

Switched-mode systems are hybrid dynamical systems having the following form,

$$\dot{x} \in \{f_\alpha(x,u,t)\}_{\alpha \in A}, \tag{1}$$

where $x \in R^n$ is the state variable, $u \in R^k$ is an exogenous input, $t \in [0,T]$ for a given time-interval $[0,T]$, and A is a given set of *modes*. For every $\alpha \in A$, the function $f_\alpha : R^n \times R^k \times [0,T] \rightarrow R^n$ is called a *modal function*. The term $\alpha \in A$ can be viewed as a logical variable, and the set of rules governing its assignment, henceforth called the *switching law*, typically is implemented by a supervisory controller. Such hybrid systems arise in various application domains, including robotics [1, 7], production control [3], power converters [9], scheduling of medical treatment [18], and, generally, situations where a controller has to switch its attention among various subsystems or sensory sources [6, 11].

Optimal control problems on switched-mode dynamical systems typically involve the minimization of a cost functional defined on the state and input trajectories as a function of the input and the switching times. Ref. [5] established a general framework for the optimal control problem, and Refs. [17, 15] derived suitable variants of the maximum principle. The special case of autonomous systems, where the input term u is absent, has been considered in [8, 20, 21]; these references considered the switching times as the design variables (for a given sequence of modes) and devised various nonlinear-programming algorithms for the optimal control problem. Another important special case arises when the dynamical system is piecewise affine and the cost functional is quadratic. This case has been investigated in [2, 10, 12, 14], with results related to the computational complexity required for solving the optimal control problem.

This paper addresses such an optimal control problem in the setting of nonlinear, autonomous systems. Unlike the above-mentioned references [8, 20, 21], the design variable does not consist of the switching times, but rather of parameters of the switching surfaces. The switching surfaces, contained in the system's state space, define a feedback law for switching among the modes according to the manner they are traversed by the system's state trajectory. In various application domains like robotics [1], it is common to construct the switching surfaces in ad-hoc ways that make simple the implementation of the corresponding feedback laws. In this paper we assume that the surfaces are given in parametric forms that depend each on a finite-dimensional variable, and we consider a given cost functional, defined on the system's state trajectory, as a function of these variables.

Our motivation comes from problems in behavior-based robotics [1], where it is required to navigate mobile robots towards their targets while avoiding obstacles along the way. It is a common practice to surround each obstacle by a circle referred to as the *guard*. When the robot is contained inside the area defined by the guard it is instructed to move away from the obstacle (avoid-obstacle mode), whereas when it is outside of the areas defined by the guards, it is instructed to move towards the target (approach-goal mode). A related problem is how to choose the guards' radii so as to minimize a given cost functional that penalizes the distance from the target as well as proximity to the obstacles. This is a problem of choosing an optimal feedback law from a parameterized set, and we will address a variant thereof later in the paper (in Section 3). First, however, we will develop the theoretical framework in a broader setting of switched-mode systems that can be applied in other areas as well. Some of the theoretical results derived below are contained, in briefer forms, in Refs. [4, 19], which constitute preliminary versions of this paper.

The underlying system that we consider is nonlinear, autonomous, and its dynamics are characterized by the following equation,

$$\dot{x} \;=\; f_i(x), \qquad \text{for all } t \in [\tau_{i-1}, \tau_i], \qquad i = 1, \ldots, N+1, \tag{2}$$

with the appropriate one-sided derivatives at the boundary points τ_{i-1} and τ_i. Here $x \in R^n$ is the state variable, the initial state $x_0 = x(0)$ and the final time $T > 0$ are given, and $f_i : R^n \to R^n$ is the *ith* modal function; the collection of the intervals $[\tau_{i-1}, \tau_i)$, $i = 1, \ldots, N+1$, with $\tau_0 := 0$ and $\tau_{N+1} := T$, constitutes a partition of the time-interval $[0, T)$. The modal functions are chosen from a given set $\{f_\alpha\}_{\alpha \in A}$, and they as well as the switching times τ_i, $i = 1, \ldots, N$, are determined in the following recursive fashion. Given f_i and $\tau_{i-1} \geq 0$ for some $i = 1, 2, \ldots$, let $A(i) \subset A$ be a given finite set of modes, labelled the *set of modes enabled by* f_i. For every $\alpha \in A(i)$, let $S_\alpha \subset R^n$ be an $n-1$ - dimensional surface. Now we define τ_i by

$$\tau_i := \min\{t > \tau_{i-1} \;:\; x(t) \in \cup_{\alpha \in A(i)} S_\alpha\}, \tag{3}$$

and we note that it is possible to have $\tau_i = \infty$. If $\tau_i < \infty$ then we pick $\tilde{\alpha} \in A(i)$ such that $x(\tau_i) \in S_{\tilde{\alpha}}$, and we set $f_{i+1} = f_{\tilde{\alpha}}$.[1] This recursive procedure is initialized by defining $\tau_0 := 0$ and with a proper choice of f_1, possibly depending on the initial condition x_0. Put in words, the definitions of τ_i and the next switching function, f_{i+1}, are quite simple. Starting at time τ_{i-1} with the modal function f_i, the system's state evolves according to the equation $\dot{x} = f_i(x)$, until one of the surfaces $S_{\tilde{\alpha}}$, for some $\tilde{\alpha} \in A(i)$, is reached. The time of hitting this surface defines τ_i, and the index of the surface, $\tilde{\alpha}$, defines f_{i+1}.

In this paper, the surfaces S_α are defined by the solution points of parameterized equations from R^n to R. We denote the parameter by a and suppose that $a \in R^k$ for some integer $k \geq 1$. For every $\alpha \in A$, let $g_\alpha : R^n \times R^k \to R$ be a continuously differentiable function. For a given fixed value of $a \in R^k$, denoted here by a_α, the switching curve S_α is defined by the solution points x of the equation $g_\alpha(x, a_\alpha) \;=\; 0$. Note that under mild assumption S_α is a smooth $(n-1)$ - dimensional manifold in R^n, and a_α can be viewed as a control parameter of this surface. Using the terminology defined earlier, we will replace the index α by i; thus, S_i is the solution set of the equation

$$g_i(x, a_i) \;=\; 0, \tag{4}$$

which is parameterized by the control variable $a_i \in R^k$.

Next, let $L : R^n \to R$ be a continuously differentiable function, and consider the cost functional J, defined by

$$J \;=\; \int_0^T L(x)dt. \tag{5}$$

This cost functional J can be viewed as a function of the control parameters $a_1, a_2, \ldots$, since the state trajectory depends on the switching surfaces. These parameters need not be independent of each other, and they may be tied together by simple constraints. For example, consider the robotic application discussed earlier, and suppose that the terrain has a single obstacle. Then there is only one guard, whose radius, denoted by a, determines the switchover times of the vehicle between the approach-goal mode and the avoid-obstacle mode. That guard may be traversed multiple times, in which case we have the condition that $a_i = a$ for all $i = 1, 2, \ldots,$. In any event, the optimal control problem is to minimize J over all possible (feasible) values of these control parameters. Our

[1] We can assume without loss of generality that $\tilde{\alpha}$ is unique, otherwise we can choose $\tilde{\alpha}$ according to some predetermined rule.

approach to this problem is to derive a characterization of the derivative terms $\frac{dJ}{da_i}$, $i = 1, 2, \ldots$, which will give us both an optimality condition and a way to apply gradient-descent algorithms.[2]

As already mentioned the optimal control problem, and hence its solution, depend on the initial condition x_0, and a word must be said in justification of this fact. In the classical setting of optimal control, the solution sought generally depends on the specific initial conditions. Exceptions, typically involving the optimal control via a state-feedback law that is independent of the initial conditions, are possible in some cases involving linear systems (e.g., the LQR problem and minimum-time problems whose solutions are characterized by bang-bang control). A similar situation arises in hybrid systems (see [2, 16]). This paper addresses the general nonlinear problem, and hence we expect its solution to depend on x_0. The derived initial-state-dependent control law may have some advantages over an open-loop control. In our specific robotic example, a circular guard may be crossed multiple times, and hence the closed-loop control is specified by a single number, the guard's radius, as compared to the multiple switching times required to specify the open-loop control. Moreover, in the general case, closed-loop control often has better performance robustness with respect to a system's parameter variations than open loop control.

The rest of the paper is organized as follows. In Section 2 we derive a costate-based formula for the partial derivatives, based on the special structure of the system and its hybrid nature. Section 3 deploys these formulae in a gradient-descent algorithm for solving an obstacle-avoidance problem in robotics. Finally, Section 4 concludes the paper and suggests directions for future research.

2 Formulation of the Gradient

This section derives expressions for the derivatives of the cost functional with respect to the control parameters. We point out that the functional J may be nondifferentiable, and even discontinuous, at points where the state trajectory is tangent to the switching surface, and this issue will be brought up later in the context of a concrete example problem. However, the present section focuses on the local property of differentiability, and hence it assumes throughout that the derivatives exist. Accordingly, we will assume a given sequence of switching surfaces and modal functions, so that the system has the following structure. The state equation is given by Eq. (2) with given and fixed initial state $x_0 := x(0)$ and final time $T > 0$, where we define $\tau_0 := 0$ and $\tau_{N+1} = T$ (T is fixed as well). The switching times τ_i are defined by Eqs. (3)-(4). Defining x_i by $x_i = x(\tau_i)$, (4) assumes the form

$$g_i(x_i, a_i) \;=\; g_i(x(\tau_i), a_i) \;=\; 0, \tag{6}$$

where we note that x_i generally depends on a_i. The cost functional J is defined by Eq. (5), and it is viewed as a function of the control variables a_i, $i = 1, 2, \ldots$. These control variables may have to satisfy various equality or inequality constraints; for instance, in the earlier-discussed case of a single guard that may be traversed multiple times, we have seen that $a_i = a_1$ for all $i = 2, \ldots, N$. The functions f_i, g_i, and L are assumed to have the following properties.

Assumption 2.1. (i) The functions f_i and g_i, $i = 1, \ldots, N+1$, and L are continuously differentiable throughout R^n, R^{n+k}, and R^n, respectively. (ii) There exists a constant $K > 0$ such

[2] We use the term $\frac{dJ}{da_i}$ to mean the total derivative with respect to a_i, and similarly, throughout the paper, terms like $\frac{d}{d\tau_i}$ will mean total derivatives as well. In contrast, we reserve the partial-derivative notation to situations where the function in question is assumed to be given in closed form in terms of its variables. For instance, the function $g_i(x_i, a_i)$ is assumed to have this form, and hence we will use the term $\frac{\partial g_i}{\partial x_i}$ for the partial derivative.

that, for every $x \in R^n$ and for all $i = 1, \ldots, N+1$,

$$||f_i(x)|| \;\leq\; K(||x|| + 1). \tag{7}$$

□

This assumption guarantees the existence of unique solutions to equations of the form $\dot{x} = f_i(x)$, with an initial condition x_{i-1} at a time τ_{i-1}, for any interval $[\tau_{i-1}, \tau_i]$. Furthermore, let us define the terms R_i and L_i by

$$R_i := f_i(x_i) - f_{i+1}(x_i) \tag{8}$$

and

$$L_i := \frac{\partial g_i}{\partial x}(x_i, a_i) f_i(x_i), \tag{9}$$

where we recognize the last term as the Lie derivative of g_i in the direction of the vector field f_i[3].

Obviously, J is a function of the control parameters a_i, $i = 1, \ldots, N$, via Eqs. (5), (2), and (6). Let us fix $a_1, \ldots, a_N$. In this section we are concerned with the total derivatives $\frac{dJ}{da_i}$, and to ensure their existence, we make the following assumption.

Assumption 2.2. For all $i = 1, \ldots, N$, $L_i \neq 0$. □

The derivative $\frac{dJ}{da_i}$ can be related to the total derivative term $\frac{dJ}{d\tau_i}$ in the following way.

Proposition 2.1. The following equation is in force,

$$\frac{dJ}{da_i} \;=\; -\,\frac{1}{L_i}\,\frac{dJ}{d\tau_i}\,\frac{\partial g_i}{\partial a}(x_i, a_i). \tag{10}$$

Proof. Taking derivative with respect to a_i in Eq. (6) and applying the chain rule we obtain,

$$\frac{\partial g_i}{\partial x}(x_i, a_i)\,\frac{dx_i}{d\tau_i}\,\frac{d\tau_i}{da_i} \;+\; \frac{\partial g_i}{\partial a}(x_i, a_i) \;=\; 0. \tag{11}$$

Noting that $\frac{dx_i}{d\tau_i} = f_i(x_i)$ and recalling Eq. (9), we get that

$$\frac{d\tau_i}{da_i} \;=\; -\,\frac{1}{L_i}\,\frac{\partial g_i}{\partial a}(x_i, a_i). \tag{12}$$

Finally, noting that $\frac{dJ}{da_i} = \frac{dJ}{d\tau_i}\frac{d\tau_i}{da_i}$, Eq. (10) follows from (12). □

The term $\frac{dJ}{da_i}$ can be computed via Eq. (10) once we can compute the derivative term $\frac{dJ}{d\tau_i}$. The computation of the latter term is quite complicated since, by Eqs. (2) and (6), τ_j, $j = i+1, \ldots, N$, are all functions of τ_i. Therefore, much of the rest of the analysis in this section concerns the development of a formula for $\frac{dJ}{d\tau_i}$. To begin, we establish some preliminary results. By Eq. (5),

$$J \;=\; \int_0^T L(x(t))dt \;=\; \sum_{j=0}^{N} \int_{\tau_j}^{\tau_{j+1}} L(x(t))dt, \tag{13}$$

where we recall that, by definition, $\tau_0 := 0$ and $\tau_{N+1} := T$. Taking the derivative with respect to τ_i we obtain,

$$\frac{dJ}{d\tau_i} \;=\; \sum_{j=0}^{N} \Big(\int_{\tau_j}^{\tau_{j+1}} \frac{\partial L}{\partial x}(x(t))\frac{dx(t)}{d\tau_i}dt \;+\; L(x(\tau_{j+1}^-))\frac{d\tau_{j+1}}{d\tau_i} \;-\; L(x(\tau_j^+))\frac{d\tau_j}{d\tau_i}\Big), \tag{14}$$

[3]We henceforth adhere to the notational convention that, for a function $f : R^n \rightarrow R^m$, the derivative $\partial f/\partial x$ is an $m \times n$ matrix. Therefore, the term $\frac{\partial g_i}{\partial x}(x_i, a_i)$ is an n-dimensional *row* vector, and $L_i \in R$.

where the superscripts - and + at τ_{j+1} and τ_j indicate left limit and right limit, respectively. Now $x(\cdot)$ is continuous at all $t \in [0,T]$ and $L(\cdot)$ is continuous in x, and hence $L(x(\tau_j^-)) = L(x(\tau_j^+))$. Moreover, since $\tau_0 = 0$ and $\tau_{N+1} = T$, we have that $\frac{d\tau_0}{d\tau_i} = \frac{d\tau_{N+1}}{d\tau_i} = 0$. Therefore, we have that

$$\sum_{j=0}^{N} \big(L(x(\tau_{j+1}^-))\frac{d\tau_{j+1}}{d\tau_i} - L(x(\tau_j^+))\frac{d\tau_j}{d\tau_i}\big) = 0.$$

Consequently, and by (14),

$$\frac{dJ}{d\tau_i} \;=\; \sum_{j=0}^{N} \int_{\tau_j}^{\tau_{j+1}} \frac{\partial L}{\partial x}(x(t))\frac{dx(t)}{d\tau_i}dt. \tag{15}$$

Since by (2) $x(t)$ does not depend on τ_i for all $t < \tau_i$, we conclude that

$$\frac{dJ}{d\tau_i} \;=\; \sum_{j=i}^{N} \int_{\tau_j}^{\tau_{j+1}} \frac{\partial L}{\partial x}(x(t))\frac{dx(t)}{d\tau_i}dt. \tag{16}$$

The question is then how to compute the derivative term $\frac{dx(t)}{d\tau_i}$, from which (16) will yield $\frac{dJ}{d\tau_i}$.

To derive a formula for the above derivative term we linearize the state equation (2). This state equation is defined in a piecewise manner, and hence we linearize it one-piece-at-a-time. Thus, let us denote by $\Phi_i(t,\tau)$ the state transition matrix of the linearized equation $\dot{z} = \frac{\partial f_i(x)}{\partial x}z$. Then we have the following immediate result.

Lemma 2.1. Let $z(\cdot) : [\tau_i, \tau_{i+1}] \to R^n$ be a continuous function, and let $r \in R^n$ be a given vector. Suppose that for every $t \in [\tau_i, \tau_{i+1}]$, we have that

$$z(t) \;=\; \int_{\tau_i}^{t} \frac{\partial f_{i+1}}{\partial x}(x(\tau))z(\tau)d\tau + r. \tag{17}$$

Then, for every $t \in [\tau_i, \tau_{i+1}]$,

$$z(t) \;=\; \Phi_{i+1}(t,\tau_i)r. \tag{18}$$

Proof. Follows immediately by differentiating (17) with respect to t. □

Now fix $i \in \{1,\ldots,N\}$ and consider a time-point $t \in [\tau_j, \tau_{j+1})$ for some $j \in \{i,\ldots,N\}$. The term $\frac{dx(t)}{d\tau_i}$ involves the aforementioned linearized systems and their respective state transition matrices. To tie these systems together in the time domain across the switchover points τ_i, we define the $n \times n$ matrices $\Theta_{j,i}$, $j = i,\ldots,N$, in the following recursive manner.

$$\Theta_{i,i} \;=\; \Phi_{i+1}(\tau_{i+1},\tau_i)^{-1}, \tag{19}$$

and for every $j = i,\ldots,N-1$,

$$\Theta_{j+1,i} \;=\; \big(I - \frac{1}{L_{j+1}} \times R_{j+1}\frac{\partial g_{j+1}}{\partial x}(x_{j+1}, a_{j+1})\big)\Phi_{j+1}(\tau_{j+1},\tau_j)\Theta_{j,i}. \tag{20}$$

With these matrices we now can obtain the following expression for $\frac{dx(t)}{d\tau_i}$ that results from the linearized system.

Lemma 2.2. For every $j = i,\ldots,N$, and for every $t \in (\tau_j, \tau_{j+1})$,

$$\frac{dx(t)}{d\tau_i} \;=\; \Phi_{j+1}(t,\tau_j)\Theta_{j,i}\Phi_{i+1}(\tau_{i+1},\tau_i)R_i. \tag{21}$$

Proof. We prove the statement by induction on $j = i, \ldots, N$. Consider first the case where $j = i$. For every $t \in (\tau_i, \tau_{i+1})$,

$$x(t) \;=\; x(\tau_i) + \int_{\tau_i}^{t} f_{i+1}(x(\tau))d\tau. \tag{22}$$

Taking derivatives with respect to τ_i (and recalling that $x_i := x(\tau_i)$),

$$\frac{dx(t)}{d\tau_i} \;=\; f_i(x_i) - f_{i+1}(x_i) + \int_{\tau_i}^{t} \frac{\partial f_{i+1}}{\partial x}(x(\tau))\frac{dx(\tau)}{d\tau_i}d\tau \;=\; R_i + \int_{\tau_i}^{t} \frac{\partial f_{i+1}}{\partial x}(x(\tau))\frac{dx(\tau)}{d\tau_i}d\tau \tag{23}$$

(see (8)). By lemma 2.1 as applied to $\frac{dx(t)}{d\tau_i}$,

$$\frac{dx(t)}{d\tau_i} \;=\; \Phi_{i+1}(t, \tau_i)R_i. \tag{24}$$

By (19) and (24), (21) follows with $j = i$.

Suppose now that (21) holds for some $j \in \{i, \ldots, N-1\}$. We next prove it for $j+1$. Note that for every $t \in [\tau_j, \tau_{j+1}]$,

$$x(t) \;=\; x(\tau_j) + \int_{\tau_j}^{t} f_{j+1}(x(\tau))d\tau, \tag{25}$$

and in particular, for $t = \tau_{j+1}$,

$$x(\tau_{j+1}) \;=\; x(\tau_j) + \int_{\tau_j}^{\tau_{j+1}} f_{j+1}(x(\tau))d\tau. \tag{26}$$

Now let us compare the derivatives with respect to τ_i in Eqs. (25) and (26). The derivative in (25) yields $\frac{dx(t)}{d\tau_i}$, whose value is given by (21) by dint of the induction's hypothesis. The derivative in (26) yields the same expression as the derivative in (25) (with τ_{j+1} instead of t) plus the additional term $f_{j+1}(x_{j+1})\frac{d\tau_{j+1}}{d\tau_i}$. In other words, we have that

$$\begin{aligned}\frac{dx(\tau_{j+1})}{d\tau_i} \;&=\; \frac{dx(t)}{d\tau_i}|_{t=\tau_{j+1}} \;+\; f_{j+1}(x_{j+1})\frac{d\tau_{j+1}}{d\tau_i} \\ &=\; \Phi_{j+1}(\tau_{j+1}, \tau_j)\Theta_{j,i}\Phi_{i+1}(\tau_{i+1}, \tau_i)R_i + f_{j+1}(x_{j+1})\frac{d\tau_{j+1}}{d\tau_i},\end{aligned} \tag{27}$$

where the last equality follows from (21).

Consider next $t \in (\tau_{j+1}, \tau_{j+2})$ We have that

$$x(t) \;=\; x(\tau_{j+1}) + \int_{\tau_{j+1}}^{t} f_{j+2}(x(\tau))d\tau, \tag{28}$$

and by taking derivatives with respect to τ_i, we obtain,

$$\frac{dx(t)}{d\tau_i} \;=\; \frac{dx(\tau_{j+1})}{d\tau_i} - f_{j+2}(x_{j+1})\frac{d\tau_{j+1}}{d\tau_i} + \int_{\tau_{j+1}}^{t} \frac{\partial f_{j+2}}{\partial x}(x(\tau))\frac{dx(\tau)}{d\tau_i}d\tau. \tag{29}$$

Now plug Eq. (27) for the first term in the RHS of (29) to obtain,

$$\frac{dx(t)}{d\tau_i} \;=\; \Phi_{j+1}(\tau_{j+1}, \tau_j)\Theta_{j,i}\Phi_{i+1}(\tau_{i+1}, \tau_i)R_i + R_{j+1}\frac{d\tau_{j+1}}{d\tau_i} + \int_{\tau_{j+1}}^{t} \frac{\partial f_{j+2}}{\partial x}(x(\tau))\frac{dx(\tau)}{d\tau_i}d\tau. \tag{30}$$

By Lemma 2.1 we have, for all $t \in (\tau_{j+1}, \tau_{j+2})$,

$$\frac{dx(t)}{d\tau_i} = \Phi_{j+2}(t,\tau_{j+1})\big(\Phi_{j+1}(\tau_{j+1},\tau_j)\Theta_{j,i}\Phi_{i+1}(\tau_{i+1},\tau_i)R_i + R_{j+1}\frac{d\tau_{j+1}}{d\tau_i}\big). \tag{31}$$

The last term in (31), $\frac{d\tau_{j+1}}{d\tau_i}$, can be computed from (27) as follows. By definition of τ_{j+1}, $g_{j+1}(x_{j+1}, a_{j+1}) = 0$. Taking derivative with respect to τ_i we get that

$$\frac{\partial g_{j+1}}{\partial x}(x_{j+1}, a_{j+1})\frac{dx(\tau_{j+1})}{d\tau_i} = 0, \tag{32}$$

and hence, and accounting for (27),

$$\frac{\partial g_{j+1}}{\partial x}(x_{j+1}, a_{j+1})\big(\Phi_{j+1}(\tau_{j+1},\tau_j)\Theta_{j,i}\Phi_{i+1}(\tau_{i+1},\tau_i)R_i + f_{j+1}(x_{j+1})\frac{d\tau_{j+1}}{d\tau_i}\big) = 0. \tag{33}$$

According to (9) and solving for $\frac{d\tau_{j+1}}{d\tau_i}$ in (33) we get, after some straightforward algebra,

$$\frac{d\tau_{j+1}}{d\tau_i} = -\frac{1}{L_{j+1}} \times \frac{\partial g_{j+1}}{\partial x}(x_{j+1}, a_{j+1})\Phi_{j+1}(\tau_{j+1},\tau_j)\Theta_{j,i}\Phi_{i+1}(\tau_{i+1},\tau_i)R_i. \tag{34}$$

Plugging this in (31) we obtain, for every $t \in (\tau_{j+1}, \tau_{j+2})$,

$$\frac{dx(t)}{d\tau_i} = \Phi_{j+2}(t,\tau_{j+1})\big(I - \frac{1}{L_{j+1}} \times R_{j+1}\frac{\partial g_{j+1}}{\partial x}(x_{j+1}, a_{j+1})\big)\Phi_{j+1}(\tau_{j+1},\tau_j)\Theta_{j,i}\Phi_{i+1}(\tau_{i+1},\tau_i)R_i. \tag{35}$$

It now follows from (20) that

$$\frac{dx(t)}{d\tau_i} = \Phi_{j+2}(t,\tau_{j+1})\Theta_{j+1,i}\Phi_{i+1}(\tau_{i+1},\tau_i)R_i, \tag{36}$$

which verifies Eq. (21) for $j+1$, and hence completes the proof. □

The derivative term $\frac{dJ}{d\tau_i}$ now can be obtained by plugging Eq. (21) in Eq. (16). The resulting term would be quite complicated, but it can be simplified by using the costate technique that is common in the study of optimal control. Due to the discontinuity of the state equation (2) at the switching times, the costate is not expected to be continuous at these points. Therefore, we will define it in a piecewise manner in the intervals $[\tau_i, \tau_{i+1}]$, $i \in \{1, \ldots, N\}$. For every $i = 1, \ldots, N$, define the function $p_i : [\tau_i, \tau_{i+1}] \to R^n$ by

$$p_i(\tau)^T = \int_{\tau}^{\tau_{i+1}} \frac{\partial L}{\partial x}(x(t))\Phi_{i+1}(t,\tau)dt + \sum_{j=i+1}^{N}\int_{\tau_j}^{\tau_{j+1}} \frac{\partial L}{\partial x}(x(t))\Phi_{j+1}(t,\tau_j)dt \times \Theta_{j,i}\Phi_{i+1}(\tau_{i+1},\tau); \tag{37}$$

and for every $\tau \in (\tau_i, \tau_{i+1}]$, define the costate $p(\tau)$ by $p(\tau) = p_i(\tau)$. Observe that $p(\tau)$ is thus defined for every $\tau \in (\tau_1, T]$, and as we shall see, it needs not be defined for $\tau \in [0, \tau_1]$. Note that (37) does not have the typical form of the costate, however, we will show that this costate satisfies Eq. (39), below, which is a familiar form of the costate equation. Moreover, the discontinuities of the costate will be shown to be captured by the recursive equation (40). Finally, we will show that the derivative $\frac{dJ}{d\tau_i}$ is related to the costate by Eq. (38).[4] All of this will be proved next.

Proposition 2.2. The following relations hold.

[4]Note that $p_i(\tau)$ is defined for all $\tau \in [\tau_i, \tau_{i+1}]$ via (37), and hence the term $p_i(\tau_i)$ is defined unambiguously.

1. For all $i = 1, \ldots, N$,
$$\frac{dJ}{d\tau_i} \;=\; p_i(\tau_i)^T R_i. \tag{38}$$

2. For all $i = 1, \ldots, N$,
$$\dot{p}_i(\tau) \;=\; -\big(\frac{\partial f_{i+1}}{\partial x}(x(\tau))\big)^T p_i(\tau) - \big(\frac{\partial L}{\partial x}(x(\tau))\big)^T \tag{39}$$
throughout the interval (τ_i, τ_{i+1}).

3. The following recursive relations hold for the boundary conditions: $p_N(T) = 0$, and for all $i = N, \ldots, 2$,
$$p_{i-1}(\tau_i) \;=\; \big(I - \frac{1}{L_i} \times R_i \frac{\partial g_i}{\partial x}(x_i, a_i)\big)^T p_i(\tau_i). \tag{40}$$

□

Whereas Eqs. (38) and (39) admit simple proofs, the proof of Eq. (40) is technically involved and it requires the following lemma, concerning a recursive relation of the matrices $\Theta_{j,i}$ in their second index.

Lemma 2.3. For every $i = 2, \ldots, N$, and for all $j = i, \ldots, N$,
$$\Theta_{j,i-1} \;=\; \Theta_{j,i}\Phi_{i+1}(\tau_{i+1}, \tau_i)\big(I - \frac{1}{L_i} \times R_i \frac{\partial g_i}{\partial x}(x_i, a_i)\big). \tag{41}$$

Proof. Fix $i \in \{2, \ldots, N\}$. We will prove (41) by induction on $j = i, \ldots, N$.

First, consider the case where $j = i$. By (19) with $i - 1$,
$$\Theta_{i-1,i-1} \;=\; \Phi_i(\tau_i, \tau_{i-1})^{-1}.$$
Therefore, and by (20), the left-hand side (LHS) of (41) has the following form,
$$\Theta_{i,i-1} \;=\; I - \frac{1}{L_i} \times R_i \frac{\partial g_i}{\partial x}(x_i, a_i).$$
By Eq. (19), the RHS of (41) (with $j = i$) has the same form. This proves (41) for $j = i$.

Next, suppose that (41) is in force for some $j \in \{i, \ldots, N-1\}$, and consider the case of $j+1$. An application of (20) yields,
$$\Theta_{j+1,i-1} \;=\; \big(I - \frac{1}{L_{j+1}} \times R_{j+1} \frac{\partial g_{j+1}}{\partial x}(x_{j+1}, a_{j+1})\big)\Phi_{j+1}(\tau_{j+1}, \tau_j)\Theta_{j,i-1},$$
and by using the induction's hypothesis (Eq. (41)) in the last term we obtain,
$$\begin{aligned} &\Theta_{j+1,i-1} \\ =\; &\big(I - \frac{1}{L_{j+1}} \times R_{j+1} \frac{\partial g_{j+1}}{\partial x}(x_{j+1}, a_{j+1})\big)\Phi_{j+1}(\tau_{j+1}, \tau_j)\Theta_{j,i}\Phi_{i+1}(\tau_{i+1}, \tau_i) \\ &\times \big(I - \frac{1}{L_i} \times R_i \frac{\partial g_i}{\partial x}(x_i, a_i)\big). \end{aligned} \tag{42}$$
Now we recognize the first three multiplicative terms in the RHS of (42) as the RHS of (20), and therefore, plugging in the LHS of (20), we obtain,
$$\Theta_{j+1,i-1} \;=\; \Theta_{j+1,i}\Phi_{i+1}(\tau_{i+1}, \tau_i)\big(I - \frac{1}{L_i} \times R_i \frac{\partial g_i}{\partial x}(x_i, a_i)\big).$$
But this is Eq. (41) with $j + 1$, thus completing the proof. □

Proof of Proposition 2.2.

1. Fix $i \in \{1, \ldots, N\}$. Plug τ_i for τ in (37) to get

$$\begin{aligned} p_i(\tau_i)^T &= \int_{\tau_i}^{\tau_{i+1}} \frac{\partial L}{\partial x}(x(t))\Phi_{i+1}(t,\tau_i)dt \\ &+ \sum_{j=i+1}^{N} \int_{\tau_j}^{\tau_{j+1}} \frac{\partial L}{\partial x}(x(t))\Phi_{j+1}(t,\tau_j)dt \times \Theta_{j,i}\Phi_{i+1}(\tau_{i+1},\tau_i). \end{aligned} \tag{43}$$

   Accounting for (19), (43) implies that

$$p_i(\tau_i)^T = \sum_{j=i}^{N} \int_{\tau_j}^{\tau_{j+1}} \frac{\partial L}{\partial x}(x(t))\Phi_{j+1}(t,\tau_j)dt \times \Theta_{j,i}\Phi_{i+1}(\tau_{i+1},\tau_i). \tag{44}$$

   Therefore, and by (16) and (21), (38) is obtained.

2. The differential equation (39) immediately follows from the definition of $p_i(\tau)^T$ in Eq. (37).

3. By (37), for all $\tau \in (\tau_N, T)$,

$$p_N(\tau)^T = \int_{\tau}^{T} \frac{\partial L}{\partial x}(x(t))\Phi_{N+1}(t,\tau)dt,$$

   and hence $p_N(T) = 0$.

   Next, fix $i \in \{2, \ldots, N\}$. Consider (37) with $i-1$ and evaluate it at $\tau = \tau_i$ to obtain,

$$p_{i-1}(\tau_i)^T = \sum_{j=i}^{N} \int_{\tau_j}^{\tau_{j+1}} \frac{\partial L}{\partial x}(x(t))\Phi_{j+1}(t,\tau_j)dt \times \Theta_{j,i-1}. \tag{45}$$

   Use Lemma 2.3 to plug in the RHS of (41) in (45) to get,

$$p_{i-1}(\tau_i)^T = \sum_{j=i}^{N} \int_{\tau_j}^{\tau_{j+1}} \frac{\partial L}{\partial x}(x(t))\Phi_{j+1}(t,\tau_j)dt \times \Theta_{j,i}\Phi_{i+1}(\tau_{i+1},\tau_i)\big(I - \frac{1}{L_i} \times R_i \frac{\partial g_i}{\partial x}(x_i,a_i)\big). \tag{46}$$

   By (44) we recognize the RHS of (46) as $p_i(\tau_i)^T\big(I - \frac{1}{L_i} \times R_i \frac{\partial g_i}{\partial x}(x_i,a_i)\big)$, whence (40) follows.

This completes the proof. □

In summary, the derivatives $\frac{dJ}{da_i}$ can be computed as follows: compute the state trajectory forward in time by Eq. (2), and then compute the costate trajectory backwards by Eqs. (39) and (40). Then, $\frac{dJ}{d\tau_i}$ is given by (38), and $\frac{dJ}{da_i}$ is given in terms of $\frac{dJ}{d\tau_i}$ via (10).

3 Robotics Example

As an example we consider the problem of controlling an autonomous mobile robot in the framework of behavior-based control [1]. The robot has to reach a pre-specified target from a given initial condition (position, orientation) while avoiding an obstacle along the way. Typically, the obstacle is surrounded by a circular guard that determines the mode of the robot: avoid-obstacle mode when the robot is within the area defined by the guard, and approach-goal mode, when outside of that area. The radius of the guard often is determined by ad-hoc ways that balance the distance from the obstacle with proximity to the goal (see [1, 7] and references therein).

One potential problem with this approach is that the robot may traverse the guard, and hence change its mode, many times in a short time-interval, since each change of the mode may steer it back towards the guard. To get around this problem we replace the single guard with two circles having a common center at the obstacle. When in the goal-approach mode the robot is outside of the inner circle, and it will change its mode once that circle is traversed. Likewise, in the avoid-obstacle mode the robot is inside the outer circle, and it will change its mode once it traverses that circle. We denote the radii of these circles by a_1 and a_2, where $a_2 \geq a_1$, and we will consider these radii as the control parameters of our optimization problem.

The robot dynamics are of the unicycle type, i.e.

$$\begin{aligned} \dot{x} &= v \cos \phi \\ \dot{y} &= v \sin \phi \\ \dot{\phi} &= \omega, \end{aligned}$$

where (x, y) is the position of the robot and ϕ is its orientation, while v and ω are its translational and angular velocities. v is assumed to have a constant value, while ω is controlled through the particular behavior (mode) currently in force. Let the target be located at a given point $(x_g, y_g) \in R^2$, and let the obstacle be located at another given point, $(x_o, y_o) \in R^2$ (the subscripts are 'g' for "goal" and 'o' for "obstacle"). In the approach-goal mode, henceforth denoted by Mode 1, the controlled velocities are given by

$$\text{Mode 1:} \quad \begin{cases} v = 1 \\ \omega = C_1(\phi_g - \phi), \end{cases}$$

where $\phi_g = \arctan((y_g - y)/(x_g - x))$ and $C_1 > 0$ is a given constant. In other words, the translational velocity has the constant value of 1, and the angular velocity acts to orient the robot towards the target. Similarly, in the avoid-obstacle mode (henceforth denoted by Mode 2), the controlled velocities are given by

$$\text{Mode 2:} \quad \begin{cases} v = 1 \\ \omega = C_2(\phi - \phi_o), \end{cases}$$

where $\phi_o = \arctan((y_o - y)/(x_o - x))$, and C_2 is a given constant. The translational velocity has the same constant value of 1, while the angular velocity steers the robot away from the obstacle.

The performance function that we minimize penalizes distance from the target as well as proximity to the obstacle, and it was defined, in a somewhat arbitrary fashion, as follows.

$$J = \int_0^T \Big((x_g - x(t))^2 + (y_g - y(t))^2 + \beta e^{-\alpha[(x_o - x(t))^2 + (y_o - y(t))^2]} \Big) dt, \tag{47}$$

where $\alpha > 0$ and $\beta > 0$ are two given constants. As earlier mentioned, we view J as a function of the radii a_1 and a_2. Thus, we define the functions $g_i(x, y, a_i)$, $i = 1, 2$, by

$$g_i(x, y, a_i) = (x_o - x)^2 + (y_o - y)^2 - a_i^2,$$

and we note that the system switches between Mode i and Mode $i+1(mod 2)$ when $g_i(x, y, a_i) = 0$.

To make the problem concrete we fixed its parameters as follows,

$$\begin{aligned} &C_1 = 1.2,\ C_2 = 0.5,\ T = 3,\ \alpha = 10,\ \beta = 10 \\ &(x_g, y_g) = (2.25, 2),\ (x_o, y_o) = (1, 1),\ (x_0, y_0, \phi_0) = (0, 0, 0). \end{aligned}$$

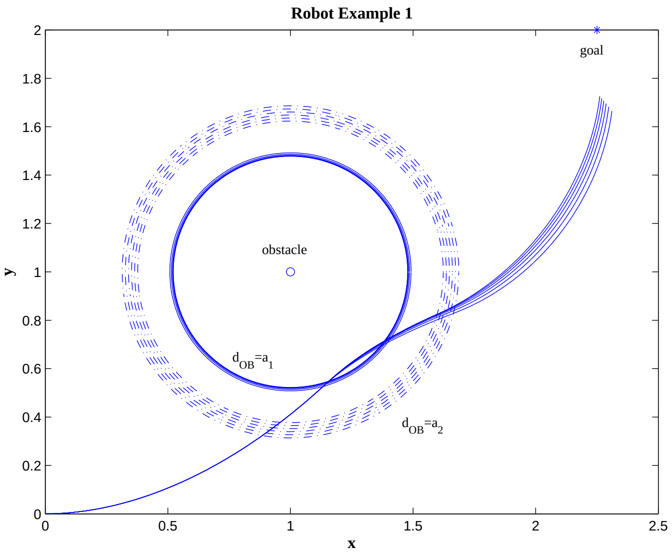

Figure 1: Trajectory of the robot for various parameter values

We minimized J by applying the well-known and well-tested steepest-descent algorithm with Armijo stepsizes (see [13] for a detailed discussion), and we computed the derivatives $\frac{dJ}{da_i}$, $i = 1, 2$ by Proposition 2.2. The results are shown in Figure 1, where the initial parameter-values were $a_1 = 0.5$, $a_2 = 0.7$. a_1 did not vary much, but a_2 changed and approached its final value of 0.61. The trajectories of the robot according to the various parameter values are shown. Figure 2 indicates that convergence has indeed taken place: the cost value declines from 9.62 to about 9.2, while the gradient's magnitude declines from about 0.57 to about 0.

Now it should be clear that such an algorithm has to be deployed with caution. The reason is that, if the motion trajectory approaches a guard in a tangential fashion, then the cost functional is no longer continuous at the parameter point. Slight perturbation causing the trajectory to miss the guard may result in the elimination of one or more mode-switchings, and such perturbations in the switching schedule results in the discontinuities. This situation arises when Assumption 2.2 fails to be satisfied, i.e., the Lie derivative L_i satisfies the equation $L_i = 0$. Addressing this difficulty in an algorithmic framework is the subject of an on-going research.

4 Conclusions

This paper considers the problem of minimizing a cost functional defined on the state trajectory of switched-mode hybrid dynamical systems with respect to their switching times. The switching times are not the free parameters, but rather are determined whenever the state trajectory intercepts a controlled surface in the state space. The parameters controlling the surfaces are the variables of the optimization problem.

The problem is cast in the framework of optimal control, where variational principles are used to derive a costate-based formula for the gradient of the cost functional with respect to the control parameters. An example concerning a robot approaching a target while avoiding an obstacle is provided. The example suggests that the problem may have inherent discontinuities, which generally arise whenever the state trajectory approaches the controlled surface in a tangential

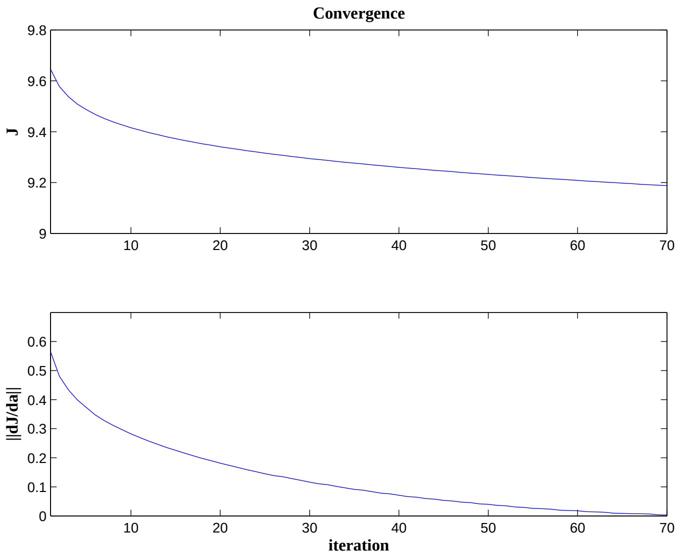

Figure 2: Cost functional and its gradient's magnitude as functions of the iteration count

fashion. This presents a challenge, since such discontinuities require a special care when using gradient-descent algorithms. Future research will address this challenge in the context of designing optimal navigation systems for mobile robots operating in obstacle-ridden environments.

References

[1] R.C. Arkin. *Behavior Based Robotics*, The MIT Press, Cambridge, Massachusetts, 1998.

[2] A. Bemporad, A. Giua, and C. Seatzu. A Master-Slave Algorithm for the Optimal Control of Continuous-Time Switched Affine Systems. *41st IEEE Conference on Decision and Control*, pp. 1976–1981, Las Vegas, Nevada, Dec. 2002.

[3] M. Boccadoro and P. Valigi. A Modelling Approach for the Dynamic Scheduling Problem of Manufacturing Systems with Non-Negligible Setup Times and Finite Buffers. *42nd IEEE Conference on Decision and Control*, Maui, Hawaii, Dec. 2003.

[4] M. Boccadoro, M. Egerstedt, and Y. Wardi. Optimal Control of Switching Surfaces in Hybrid Dynamic Systems. *IFAC Workshop on Discrete Event Systems*, Reims, France, Sept. 2004.

[5] M. Branicky, V. Borkar and S. Mitter. A Unified Framework for Hybrid Control: Model and Optimal Control Theory. *IEEE Trans. on Automatic Control*, Vol. 43, No. 1, pp. 31–45, 1998.

[6] R. Brockett. Stabilization of Motor Networks. *34th IEEE Conference on Decision and Control*, pp. 1484–1488, Dec. 1995.

[7] M. Egerstedt. Behavior Based Robotics Using Hybrid Automata. *Lecture Notes in Computer Science: Hybrid Systems III: Computation and Control*, Springer Verlag, pp. 103-116, Pittsburgh, Pennsylvania, March 2000.

[8] M. Egerstedt, Y. Wardi, and F. Delmotte. Optimal Control of Switching Times in Switched Dynamical Systems. *42nd IEEE Conference on Decision and Control*, Maui, Hawaii, Dec. 2003.

[9] D. Flieller, J.P. Louis, and J. Barrenscheen. General Sampled Data Modeling of Power Systems Supplied by Static Converter with Digital and Analog Controller. *Mathematics and Computer in Simulation*, Vol. 46, pp. 373-385, 1998.

[10] A. Giua, C. Seatzu, and C. Van Der Mee. Optimal Control of Switched Autonomous Linear Systems. *40th IEEE Conference on Decision and Control*, pp. 2472–2477, Orlando, Florida, Dec. 2001.

[11] D. Hristu-Varsakelis. Feedback Control Systems as Users of Shared Network: Communication Sequences that Guarantee Stability. *40th IEEE Conference on Decision and Control*, pp. 3631–3631, Orlando, Florida, Dec. 2001.

[12] M. Johansson and A. Rantzer. Piecewise Linear Quadratic Optimal Control. *IEEE Trans. on Automatic Control*, Vol. 43, No. 4, pp. 629–637, 2000.

[13] E. Polak, *Optimization: Algorithms and Consistent Approximations*. Springer-Verlag, 1997.

[14] P. Riedinger F. Kratz C. Iung, and C. Zanne. Linear Quadratic Optimization for Hybrid Systems. *38th IEEE Conference on Decision and Control*, pp. 3059–3064, Phoenix, Arizona, Dec. 1999.

[15] M.S. Shaikh and P.E. Caines. On Trajectory Optimization for Hybrid Systems: Theory and Algorithms for Fixed Schedules. *41st IEEE Conference on Decision and Control*, Las Vegas, Nevada, Dec. 2002.

[16] M.S. Shaikh and P. Caines. On the Optimal Control of Hybrid Systems: Optimization of Switching Times and Combinatoric Location Schedules. *American Control Conference*, Denver, Colorado, June 2003.

[17] H.J. Sussmann. A Maximum Principle for Hybrid Optimal Control Problems. *38th IEEE Conference on Decision and Control*, pp. 425–430, Phoenix, Arizona, Dec. 1999.

[18] E. Verriest. Regularization Method for Optimally Switched and Impulsive Systems with Biomedical Applications (I). *42nd IEEE Conference on Decision and Control*, Maui, Hawaii, Dec. 2003.

[19] Y. Wardi, M. Boccadoro, M. Egerstedt, and E. Verriest. Optimal Control of Switching Surfaces. *43rd IEEE Conference on Decsion and Control*, pp. 1854-1859, Nassau, Bahamas, Dec. 14-17, 2004.

[20] X. Xu and P. Antsaklis. Optimal Control of Switched Autonomous Systems. *41st IEEE Conference on Decision and Control*, Las Vegas, Nevada, Dec. 2002.

[21] X. Xu and P.J. Antsaklis. Optimal Control of Switched Systems via Nonlinear Optimization Based on Direct Differentiations of Value Functions. *International Journal of Control*, Vol. 75, pp. 1406-1426, 2002.