Optimal Control of Switching Times in Switched Dynamical Systems

M. Egerstedt, Y. Wardi, and F. Delmotte
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332, USA

***Abstract*— This paper considers an optimal control problem for switched dynamical systems, where the objective is to minimize a cost functional de£ned on the state, and where the control variable consists of the switching times. The gradient of the cost functional is derived on an especially simple form, which lends itself to be directly used in gradient-descent algorithms. This special structure of the gradient furthermore allows for the number of switching points to become part of the control variable, instead of being a given constant. Numerical examples testify to the viability of the proposed approach.**

I. Introduction

Switched dynamical systems are often described by differential inclusions of the form

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

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^k$, and $\{f_\alpha : \mathbb{R}^{n+k} \to \mathbb{R}^n\}_{\alpha\in A}$ is a collection of continuously differentiable functions, parameterized by α in a suitable set A. The time t is con£ned to an interval $[0,T]$, where it is possible that $T = \infty$. Such systems arise in a variety of applications, including situations where a control module has to switch its attention among a number of subsystems [7], [9], [11], [14], or collect data sequentially from a number of sensor sources [3], [5], [8]. A supervisory controller is used for dictating the switching law, i.e./ the rule for switching among the functions f_α in the right-hand side of Eq. (1).

Recently, there has been a mounting interest in optimal control of switched systems, where the control variable consists of a proper switching law as well as the input function $u(t)$ (see [2], [4], [6], [12], [13], [15], [16]. A special class of problems concerns autonomous systems where the term $u(t)$ is absent, the sequence of dynamic responses (the functions in the right-hand side of Eq. (1)) is £xed, and the control variable consists of the switching times [8], [17]. The latter reference [17] formulates an optimization problem in terms of minimizing a cost functional of the state in terms of the switching times. In that work, a formula for the gradient is derived and applied in various nonlinear programming algorithms.

Ref. [17] thus provides the motivation as well as the starting point for the research presented in this paper. We consider a similar problem, where the sequence of switching functions as well as the number of switching times are £xed. We develop a formula, simpler than the one in [17], for the gradient of the cost functional, and use it in conjunction with a gradient-descent algorithm.

This paper is organized as follows: Section II presents the problem (as formulated in [17]), and our proposed formula for the gradient is derived. Section III describes the algorithm, while Section IV discusses some possible extensions. Section V concludes the paper.

II. Problem formulation and Gradient Computation

The problem considered in [17], as well as here, is the following. Let $\{f_i\}_{i=0}^N$ be a £nite sequence of continuously differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}^n$. Fix $T > 0$ and $x_0 \in \mathbb{R}^n$. Given a sequence of switching times τ_i, $i = 1,\ldots,N$; and de£ning $\tau_0 = 0$ and $\tau_{N+1} = T$, consider the dynamical system

$$\dot{x}(t) \;=\; f_i(x(t)), \tag{2}$$

for all $t \in [\tau_i,\tau_{i+1})$, and for every $i \in \{0,\ldots,N\}$, with the given initial condition $x(0) = x_0$. Let $L : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function, and consider the cost functional J, de£ned by

$$J \;=\; \int_0^T L(x(t))dt. \tag{3}$$

We point out that the cost functional in [17] also includes a £nal-state term, which we omit it here without loss of generality. We consider the control parameter to consist of the switching times $\tau_1,\ldots,\tau_N$, and denote it by the N-dimensional variable $\bar{\tau} := (\tau_1,\ldots,\tau_N)^T$. Note that J is a function of $\bar{\tau}$ via Eq. (2). In this section we derive a formula for the gradient $\nabla J(\bar{\tau})$ and to this end, a collection of preliminary results are needed.

Lemma 2.1. Let $f : \mathbb{R}^n \to \mathbb{R}^n$ be a continuously differentiable function, and consider the dynamical system

$$\dot{x} \;=\; f(x(t)), \qquad t \in [0,T], \tag{4}$$

whose initial condition is $x(0) = x_0$.

(i). Given a continuously differentiable function $L : \mathbb{R}^n \to \mathbb{R}$, de£ne the cost functional J as in Eq. (3). Consider J as a function of the initial condition x_0. De£ne the costate $p(t) \in \mathbb{R}^n$ by the (backwards) differential equation

$$\begin{aligned} \dot{p}(t) \;&=\; -\Big(\frac{\partial f}{\partial x}(x(t))\Big)^T p(t) - \Big(\frac{\partial L}{\partial x}(x(t))\Big)^T \\ p(T) &= 0. \end{aligned} \tag{5}$$

Then, the gradient $\nabla J(x_0)$ has the following form,

$$\nabla J(x_0) \; = \; p(0). \tag{6}$$

(ii). Fix $t \in [0,T]$, and consider $x(t)$ as a function of the initial state x_0. Then, the derivative of this function is given by the following expression,

$$\frac{dx(t)}{dx_0} \; = \; \Phi(t,0), \tag{7}$$

where $\Phi(t,\tau)$ is the state-transition matrix of the autonomous linear, time-varying dynamical system

$$\dot{z} \; = \; \frac{\partial f(x(t))}{\partial x} z. \tag{8}$$

Proof. Part (i) is well known in optimal control theory, while part (ii) follows immediately from variational principles. ■

Lemma 2.2. Let $f_1 : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $f_2 : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be continuously differentiable functions. Consider the switched dynamical system de£ned over a given interval $[0,T]$, whose single switching time is $\tau \in (0,T)$:

$$\dot{x} \; = \; \begin{cases} f_1(x(t)), & \text{if } t \leq \tau \\ f_2(x(t)), & \text{if } t > \tau. \end{cases} \tag{9}$$

(i). Fix $t \in (\tau,T)$, and consider $x(t)$ as a function of τ. Then, the derivative of this function is given by

$$\frac{dx(t)}{d\tau} \; = \; \Phi(t,\tau)\Big(f_1(x(\tau)) - f_2(x(\tau))\Big), \tag{10}$$

where $\Phi(t,\tau)$ is the state transition matrix of the autonomous linear system

$$\dot{z} \; = \; \frac{\partial f_2(x(t))}{\partial x} z. \tag{11}$$

(ii). Let $L : \mathbb{R}^n \rightarrow \mathbb{R}$ be a continuously differentiable function, and de£ne the cost functional J as in Eq. (3). Consider J as a function of the switching time τ, and denote its derivative by $J^{'}(\tau)$. De£ne the costate $p(t) \in \mathbb{R}^n$ by the (backwards) differential equation

$$\begin{aligned} &\dot{p}(t) = -\Big(\tfrac{\partial f_2}{\partial x}(x(t))\Big)^T p(t) - \Big(\tfrac{\partial L}{\partial x}(x(t))\Big)^T \\ &p(T) = 0. \end{aligned} \tag{12}$$

Then, $J^{'}(\tau)$ has the following form,

$$J^{'}(\tau) \; = \; p(\tau)^T\Big(f_1(x(\tau)) - f_2(x(\tau))\Big). \tag{13}$$

Proof. (i). Fix $\tau \in [0,T)$ and £x $\Delta\tau > 0$ such that $\tau + \Delta\tau \leq T$. Recall that $x(t)$ is the state trajectory of the system whose switching time is τ. Likewise, let us denote by $x(t) + \Delta x(t)$ the state trajectory obtained when the switching time is $\tau + \Delta\tau$. By applying Eq. (9) £rst to τ and then to $\tau + \Delta\tau$, we observe the following:

(I). For all $t \in [0,\tau]$, $\dot{x}(t) = f_1(x(t))$ and $\dot{x}(t) + \dot{\Delta} x(t) = f_1(x(t) + \Delta x(t))$, and moreover, $x(0) = x(0) + \Delta x(0)$; and therefore, $\Delta x(t) = 0$ for all $t \in [0,\tau]$. In particular, $\Delta x(\tau) = 0$.

(II). For all $t \in [\tau, \tau + \Delta\tau]$, $\dot{x}(t) = f_2(x(t))$ and $\dot{x}(t) + \dot{\Delta} x(t) = f_1(x(t) + \Delta x(t))$. Therefore, $x(\tau + \Delta\tau) = x(\tau) + f_2(x(\tau))\Delta\tau + o(\Delta\tau)$ and $x(\tau+\Delta\tau) + \Delta x(\tau+\Delta\tau) = x(\tau) + \Delta x(\tau) + f_1(x(\tau) + \Delta x(\tau))\Delta\tau + o(\Delta\tau)$. Subtracting the former equation from the latter, and accounting for the fact that $\Delta x(\tau) = 0$, we obtain

$$\Delta x(\tau + \Delta\tau) \; = \; \Big(f_1(x(\tau)) - f_2(x(\tau))\Big)\Delta\tau + o(\Delta\tau). \tag{14}$$

(III). For every $t \in [\tau + \Delta\tau, T]$, $\dot{x}(t) = f_2(x(t))$ and $\dot{x}(t) + \dot{\Delta} x(t) = f_2(x(t) + \Delta x(t))$. Subtracting the former equation from the latter gives

$$\dot{\Delta} x(t) \; = \; \frac{\partial f_2(x(t))}{\partial x}\Delta x(t) + o(\Delta x(t)). \tag{15}$$

Now, consider Eq. (15) with the boundary condition in Eq. (14). Since $\Delta x(\tau) = 0$, the perturbation theory of ordinary differential equations implies that $\Delta x(t) = O(\Delta\tau)$, and hence Eq. (15), with the boundary condition in Eq. (14), imply that

$$\Delta x(t) \; = \; \Phi(t, \tau + \Delta\tau)\Big(f_1(x(\tau)) - f_2(x(\tau))\Big)\Delta\tau + o(\Delta\tau), \tag{16}$$

where $\Phi(t, \tau + \Delta\tau)$ is the state transition matrix associated with the differential equation

$$\dot{z} \; = \; \frac{\partial f_2(x(t))}{\partial x} z.$$

Since this matrix-function is continuous in its second argument, Eq. (10) follows by taking the limit $\Delta\tau \rightarrow 0$.

(ii). Differentiating with respect to τ in Eq. (3), and accounting for the fact that $x(t)$ is a function of τ and that it is continuous in t for a given τ, we obtain that

$$J^{'}(\tau) \; = \; \int_0^T \frac{\partial L(x(t))}{\partial x}\frac{dx(t)}{d\tau}dt. \tag{17}$$

Recall that $\frac{dx(t)}{d\tau} = 0$ for every $t < \tau$, and $\frac{dx(t)}{d\tau}$ is given by Eq. (10) for every $t > \tau$; plugging this into Eq. (17) yields

$$J^{'}(\tau) \; = \; \int_\tau^T \frac{\partial L(x(t))}{\partial x}\Phi(t,\tau)dt\Big(f_1(x(\tau)) - f_2(x(\tau))\Big)d\tau. \tag{18}$$

Now, de£ne $p(\tau) \in \mathbb{R}^n$ by

$$p(\tau)^T \; = \; \int_\tau^T \frac{\partial L(x(t))}{\partial x}\Phi(t,\tau)dt. \tag{19}$$

Then it is readily seen that $p(T)^T = 0$, and by differentiating in Eq. (19) with respect to τ

$$\dot{p}(\tau)^T \; = \; -\frac{\partial L(x(\tau))}{\partial x} - \int_\tau^T \frac{\partial L(x(t))}{\partial x}\Phi(t,\tau)dt\frac{\partial f_2(x(\tau))}{\partial x}. \tag{20}$$

This, with the aid of Eq. (19), imply Eq. (12), while Eq. (13) follows from Eq. (18). This completes the proof. ∎

Consider now the dynamical system given in Eq. (2), together with the associated cost functional de£ned by Eq. (3). Let the costate $p(t)$ be de£ned by the following backward equation

$$\dot{p}(t) = -\Big(\frac{\partial f_i}{\partial x}(x(t))\Big)^T p(t) - \Big(\frac{\partial L}{\partial x}(x(t))\Big)^T, \qquad (21)$$

for $t \in [\tau_i, \tau_{i+1}),\ i = N, \dots, 1$ and with initial condition $p(T) = 0$.

Observe that $p(\tau_i)$ is de£ned by the regression (in time) of the above equation in the interval $[\tau_i, \tau_{i+1}]$, and the costate function $p(t)$ is assumed to be continuous at the point τ_i. The next assertion characterizes the partial derivatives $\frac{dJ(\bar{\tau})}{d\tau_i}$, and hence the gradient $\nabla J(\bar{\tau})$.

Proposition 2.1. The following equation is in force for every $i \in \{1, \dots, N\}$,

$$\frac{dJ(\bar{\tau})}{d\tau_i} \;=\; p(\tau_i)^T \Big(f_{i-1}(x(\tau_i)) - f_i(x(\tau_i))\Big). \qquad (22)$$

Proof. For every $i \in \{1, \dots, N\}$, de£ne the costate $q_i(t)$ in the interval $[\tau_i, \tau_{i+1}]$ by the following equation,

$$\begin{aligned} &\dot{q}_i(t) = -\Big(\tfrac{\partial f_i}{\partial x}(x(t))\Big)^T q_i(t) - \Big(\tfrac{\partial L}{\partial x}(x(t))\Big)^T \\ &q_i(\tau_{i+1}) = 0. \end{aligned} \qquad (23)$$

Furthermore, let $\Phi_i(\xi, t)$ denote the state transition matrix of the autonomous linear dynamical system

$$\dot{z} \;=\; \frac{\partial f}{\partial x} z$$

in the interval $[\tau_i, \tau_{i+1}]$. Finally, de£ne the function $q(t) : [0, T] \to \mathbb{R}^n$ recursively (backwards) in $i = N, \dots, 1$, by

$$q(t) \;=\; q_i(t) + \Phi_i(\tau_{i+1}, t)^T q(\tau_{i+1}),\ t \in [\tau_i, \tau_{i+1}], \qquad (24)$$

with the boundary condition $q(\tau_{N+1}) = 0$ (recall that $\tau_{N+1} = T$). We next prove that, for every $t \in [0, T]$,

$$q(t) = p(t). \qquad (25)$$

The proof is by induction, backwards on $i = N, N-1, \dots, 1$, where we prove Eq. (25) for all $t \in [\tau_i, \tau_{i+1}]$. Consider £rst the case where $i = N$. Eq. (24) with $i = N$, together with the boundary condition $q(\tau_{N+1}) = 0$, imply that $q_N(t) = q(t)$ for all $t \in [\tau_N, T]$. Eq. (23) with $i = N$ implies that, for all $t \in [\tau_N, T]$,

$$\begin{aligned} &\dot{q}_N(t) = -\Big(\tfrac{\partial f_N}{\partial x}(x(t))\Big)^T q_N(t) - \Big(\tfrac{\partial L}{\partial x}(x(t))\Big)^T \\ &q_N(T) = 0. \end{aligned} \qquad (26)$$

This is identical to Eq. (21) de£ning $p(t)$ on the interval $[\tau_N, T]$, and hence $q_N(t) = p(t)$ for all $t \in [\tau_N, T]$. Since $q_N(t) = q(t)$ in the above interval, it follows that $q(t) = p(t)$ in the interval $[\tau_N, T]$.

Next, £x $i \in \{1, \dots, N-1\}$, and suppose that Eq. (25) is in force for all $t \in [\tau_{i+1}, T]$. We now will prove that this equation also holds for all $t \in [\tau_i, \tau_{i+1}]$. Fix $t \in (\tau_i, \tau_{i+1})$. Taking the derivative with respect to t in Eq. (24), and accounting for Eq. (23) and the de£nition of the state transition matrix, we have the following,

$$\begin{aligned} \dot{q}(t) \;&=\; -\Big(\frac{\partial f_i}{\partial x}(x(t))\Big)^T q_i(t) - \Big(\frac{\partial L}{\partial x}(x(t))\Big)^T \\ &-\; \Big(\frac{\partial f_i}{\partial x}(x(t))\Big)^T \Phi_i(\tau_{i+1}, t)^T q(\tau_{i+1}) \\ &=\; -\Big(\frac{\partial f_i}{\partial x}(x(t))\Big)^T q(t) - \Big(\frac{\partial L}{\partial x}(x(t))\Big)^T, \end{aligned} \qquad (27)$$

where the last equality follows from the de£nition of $q(t)$ (Eq. (24)). We recognize this as the equation de£ning $p(t)$ (Eq. (21)). Regarding the boundary condition, the induction hypothesis implies that $q(\tau_{i+1}) = p(\tau_{i+1})$. Consequently, the identity Eq. (25) is satis£ed throughout the interval $[\tau_i, \tau_{i+1}]$, and hence, by induction, throughout the interval $[0, T]$.

Next, let us £x $i \in \{1, \dots, N\}$ and consider the derivative $\frac{dJ(\bar{\tau})}{d\tau_i}$. Observe that an in£nitesimal variation (perturbation) in τ_i causes variations in $x(\tau_j)$, $j > i$, even though τ_j $(j > i)$ remain £xed. Applying Lemma 2.2(ii), and the chain rule with Lemma 2.2(i), Lemma 2.1(i), and Lemma 2.1(ii), yield the following equation,

$$\begin{aligned} \tfrac{dJ(\bar{\tau})}{d\tau_i} &= q_i(\tau_i)^T \Big(f_{i-1}(x(\tau_i)) - f_i(x(\tau_i))\Big) + \\ &+ \textstyle\sum_{j=i+1}^N q_j(\tau_j)^T \Phi_{j-1}(\tau_j, \tau_{j-1}) \cdots \Phi_i(\tau_{i+1}, \tau_i) \cdot \\ &\cdot \Big(f_{i-1}(x(\tau_i)) - f_i(x(\tau_i))\Big), \end{aligned} \qquad (28)$$

where the costates $q_j,\ j = i, \dots, N$, are de£ned on the respective intervals $[\tau_j, \tau_{j+1}]$, by Eq. (23). Rearranging the terms in Eq. (28) gives us

$$\begin{aligned} \tfrac{dJ(\bar{\tau})}{d\tau_i} &= \Big(q_i(\tau_i)^T + \\ &+ \textstyle\sum_{j=i+1}^N q_j(\tau_j)^T \Phi_{j-1}(\tau_j, \tau_{j-1}) \cdots \Phi_i(\tau_{i+1}, \tau_i)\Big) \cdot \\ &\cdot \Big(f_{i-1}(x(\tau_i)) - f_i(x(\tau_i))\Big). \end{aligned} \qquad (29)$$

Finally, a recursive ($i = N, \dots, 1$) application of Eq. (24), with $t = \tau_i$, yields that

$$\begin{aligned} &q_i(\tau_i)^T + \textstyle\sum_{j=i+1}^N q_j(\tau_j)^T \Phi_{j-1}(\tau_j, \tau_{j-1}) \cdots \Phi_i(\tau_{i+1}, \tau_i) = \\ &q(\tau_i)^T. \end{aligned} \qquad (30)$$

Plugging this into Eq. (29), together with Eq. (25), £nally yield Eq. (22), which completes the proof. ∎

III. Algorithm

This section uses the gradient formula in Eq. (22) in a descent algorithm that is applied to an example problem. The special structure of the gradient, as well as a consequent enhancement of the algorithm, will be discussed in the next section.

The algorithm that we apply is the Steepest Descent Algorithm with Armijo Stepsizes [1], [10]. Given an iteration point $\bar{\tau}(k)$ (where k is the iteration index), the next iteration point, $\bar{\tau}(k+1)$, is computed in the following way:

Steepest-descent algorithm with Armijo stepsizes.
Parameters: $\alpha \in (0,1)$ and $\beta \in (0,1)$.
Step 1: Compute $h(k) = \nabla J(\bar{\tau}(k))$.
Step 2: Compute $i(k)$, de£ned as follows: $i(k) = \min\{i \geq 0 \; : \; J(\bar{\tau}(k) - \beta^i h(k)) - J(\bar{\tau}(k)) \leq -\alpha\beta^i ||h(k)||^2\}$.
Step 3: Set $\lambda(k) = \beta^{i(k)}$, and set $\bar{\tau}(k+1) = \bar{\tau}(k) - \lambda(k)h(k)$.

This algorithm is globally convergent to stationary points, and it does not jam at non-stationary points (see [10]). Practically, the parameters α and β are often set to $\alpha = \beta = 0.5$. The search for $i(k)$ in Step 2 need not start at $i = 0$. Rather, for improving the algorithm's ef£ciency, it can start at $i(k-1) - 1$ or $i(k-1) - 2$.

We applied this algorithm to the problem of minimizing $J(\bar{\tau})$ having the form in Eq. (3), where the dynamics have the form in Eq. (2).

Example
In order to verify the numerical feasibility of the proposed algorithm, we tested the gradient descent optimization on a numerical example. In Figure 1, the result of £nding locally optimal switching times using

$$\alpha = 1/2$$
$$\beta = 1/2$$

as Armijo parameters is illustrated. The Figure shows an example with three switches, initialized to

$$\tau_1 = 0.3, \;\; \tau_2 = 0.5, \;\; \tau_3 = 0.7.$$

The system was evolved over the time interval $[0, 1]$, and the individual dynamical systems were given by the two-dimensional, linear systems

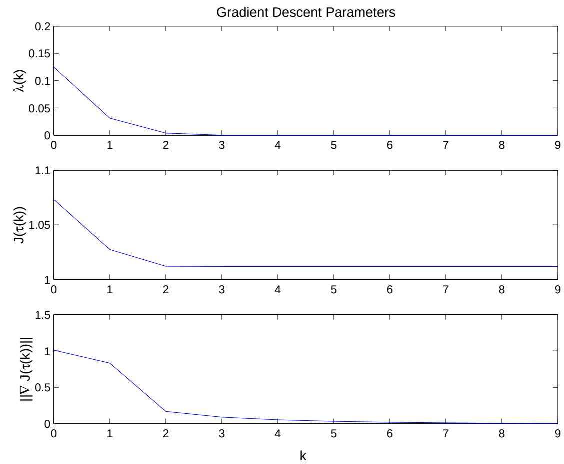
$$\begin{array}{l} \dot{x} = \begin{pmatrix} -1 & 0 \\ 1 & 2 \end{pmatrix} x = A_1 x, \;\; t \in [0, \tau_1) \\ \dot{x} = \begin{pmatrix} 1 & 1 \\ 1 & -2 \end{pmatrix} x = A_2 x, \;\; t \in [\tau_1, \tau_2) \\ \dot{x} = A_1 x, \;\; t \in [\tau_2, \tau_3) \\ \dot{x} = A_2 x, \;\; t \in [\tau_3, 1]. \end{array}$$

The cost function used in this example is

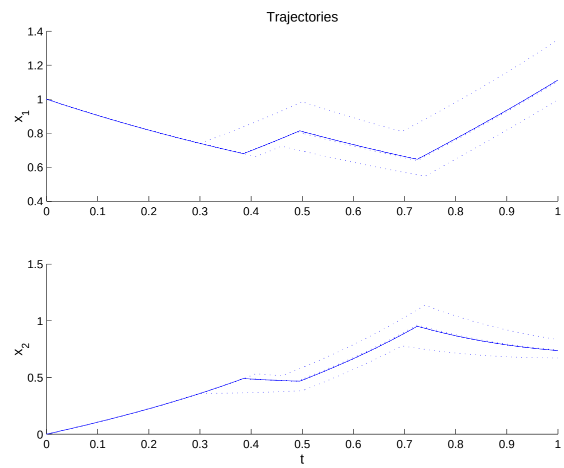
$$J = \frac{1}{2} \int_0^T \|x(t)\|^2 dt.$$

Note that the dynamics switch between modes where either the £rst or the second component of x is unstable, which means that we can expect the optimal switching-time sequence to be such that both systems get enough "attention" in order to prevent any component of x from diverging. In Figure 1, it can be seen that the optimization algorithm terminates after 9 iterations (at which point $\|\frac{\partial J}{\partial \bar{\tau}}\|$ has reached the termination value $\epsilon = 0.05$). The optimal switching times were found to be

$$\tau_1 = 0.3856, \;\; \tau_2 = 0.4952, \;\; \tau_3 = 0.7294.$$



(a)



(b)

Fig. 1. Three Switches: In the left £gure, the Armijo stepsize $\lambda(k)$ is shown (top) together with the cost J (middle) and $\|\frac{\partial J}{\partial \bar{\tau}}\|$ (bottom). In the right £gure the two x-components are displayed as functions of t. The dotted lines depict the state trajectories associated with the variable $\bar{\tau}$ computed in odd-numbered iterations (iteration 1,3, etc.), and the solid line corresponds to the £nal trajectory.

IV. Enhancements and Extensions

The gradient $\nabla J(\bar{\tau})$, based on the formula for the partial derivatives given in Eq. (22), has a special structure that makes it possible to extend and enhance gradient-descent algorithms in a number of directions. The special structure is inherent in the fact that the same costate $p(t)$ is used, in Eq. (22), for all the partial derivatives $\frac{dJ(\bar{\tau})}{d\tau_i}$, $i \in \{1, \ldots, N\}$. Thus, the computation of these partial derivatives involves two stages: (i) computing $p(t)$ by £rst solving (numerically) Eq. (2) forward and then solving Eq. (21) backwards, and (ii) evaluating the terms $f_{i-1}(x(\bar{\tau}_i)) - f_i(x(\bar{\tau}_i))$ and then multiplying them by $p(\tau_i)^T$ to obtain the partial derivatives in Eq. (22). It is readily seen that the former stage is much more complicated than the latter stage, and, in fact, the dimension of the optimization problem (i.e., the number of switching times, N) appears not to pose a problem as far as the computational complexity, associated with the gradient estimation, is concerned. In this section we describe a potential enhancement of the gradient-based algorithm that take advantage of the above special structure of the gradient.

Optimization on the number of switching times. The algorithm that we discussed earlier moves the location of the N switching points, $\tau_1, \ldots, \tau_N$. It is possible, of course, that τ_N may be pushed to T or that τ_1 be pushed to 0, and in that case the control parameter $\bar{\tau}$ will have fewer than N switching times. Also, it is possible for two adjacent switching times, τ_{i-1} and τ_i, to move toward each other until they merge into a single switching point. This will eliminate the number of switching functions in the right-hand side of Eq. (2) and reduce the number of switching times. Thus, we see that the gradient-descent algorithm, presented in Section III, can naturally reduce the number of switching times, N.

It is also possible to use gradient-related information to increase the number of switching times. To illustrate, suppose, for example, that the switching functions in the right-hand side of Eq. (2) alternate between two given functions, $g_1(x)$ and $g_2(x)$, in a round-robin fashion. Thus, assuming that g_1 and g_2 are continuously differentiable functions from $\mathbb{R}^n$ to $\mathbb{R}^n$, suppose that $f_i = g_1$ for all odd i, and $f_i = g_2$ for all even i. Now consider an interval $[\tau_i, \tau_{i+1}]$ for some odd i, so that $\dot{x} = g_1(x)$ throughout the interval. Fix $t \in (\tau_i, \tau_{i+1})$, and let $\lambda > 0$ be such that the interval $(t - \lambda/2, t + \lambda/2)$ is contained in the interval (τ_i, τ_{i+1}). We now insert two switching times, one at $t - \lambda/2$ and one at $t + \lambda/2$, and let the switching function be g_2 in the interval $[t-\lambda/2, t+\lambda/2)$. Consider J as a function of λ, with $\lambda \geq 0$. We next derive an expression for the directional derivative $\frac{dJ(0)}{d\lambda^+}$.

Observe that, at a given $\lambda > 0$, Proposition 2.1 implies the following formula for the derivative $dJ/d\lambda$:

$$\begin{aligned}\frac{dJ}{d\lambda} = \frac{1}{2}\Big(&p(t+\lambda/2)^T\Big(g_2(x(t+\lambda/2)) - g_1(x(t+\lambda/2))\Big) \\ &- p(t-\lambda/2)^T\Big(g_1(x(t-\lambda/2)) - g_2(x(t-\lambda/2))\Big)\Big).\end{aligned} \tag{31}$$

Taking the limit $\lambda \to 0$ and accounting for the continuity of the costate, we obtain,

$$\frac{dJ(0)}{d\lambda^+} \;=\; p(t)^T\Big(g_2(t) - g_1(t)\Big). \tag{32}$$

Note that a gradient-descent algorithm can exploit the above formula by £nding a point $t \in (\tau_i, \tau_{i+1})$ such that $p(t)^T\Big(g_2(t) - g_1(t)\Big) < 0$, and then pursuing a descent in the direction of injecting the switching function g_2 in the interval $[t - \lambda/2, t + \lambda/2)$. Finally, we observe that this procedure is applicable in a much broader context than the round-robin regime. Generally, Eq. (32) can be used for injecting any switching function during a given interval (τ_i, τ_{i+1}), possibly subject to constraints on the sequence of switching functions f_α, $\alpha \in A$.

V. Conclusions

This paper presented an algorithm for computing optimal switching times in a switched dynamical system. A key aspect of the algorithm is the formula for the gradient, consisting of two elements: a common costate and various time-dependent function evaluations. The computation of the costate generally requires many more operations than the function evaluations. However, once computed, the same costate can be used for all the partial derivatives of the cost functional.

The paper furthermore proposed a gradient formula in conjunction with a steepest-descent optimization algorithm, and veri£ed rapid convergence to the optima for an example problem. Moreover, the special structure of the gradient can be exploited when extending the scope of gradient-based algorithms by considering the number of switching times as a design variable.

Acknowledgments

The support from the Manufacturing Research Center at the Georgia Institute of Technology is gratefully acknowledged.

VI. References

[1] L. Armijo. Minimization of Functions Having Lipschitz Continuous First-Partial Derivatives. *Paci£c Journal of Mathematics*, Vol. 16, ppm. 1-3, 1966.

[2] M.S. Branicky, V.S. Borkar, and S.K. Mitter. A Uni£ed Framework for Hybrid Control: Model and Optimal Control Theory. *IEEE Transactions on Automatic Control*, Vol. 43, pp. 31-45, 1998.

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

[4] J. Chudoung and C. Beck. The Minimum Principle for Deterministic Impulsive Control Systems. *IEEE Conference on Decision and Control*, Vol. 4, pp. 3569 -3574, Dec. 2001.

[5] M. Egerstedt and Y. Wardi. Multi-Process Control Using Queuing Theory. *IEEE Conference on Decision and Control*, Las Vegas, NV, Dec. 2002.

[6] S. Hedlund and A. Rantzer. Optimal Control of Hybrid Systems. *Proceedings of the 38th IEEE Conference on Decision and Control*, pp. 3972-3977, 1999.

[7] D. Hristu and K. Morgansen. Limited Communication Control. *Systems & Control Letters*, No. 37, pp. 193–205, 1999.

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

[9] B. Lincoln and A. Rantzer. Optimizing Linear Systems Switching. *IEEE Conference on Decision and Control*, pp. 2063–2068, Orlando, FL, 2001.

[10] E. Polak. *Optimization Algorithms and Consistent Approximations*. Springer-Verlag, New York, New York, 1997.

[11] H. Rehbinder and M. San£rdson. Scheduling of a Limited Communication Channel for Optimal Control. *IEEE Conference on Decision and Control*, Sidney, Australia, Dec. 2000.

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

[13] H.J. Sussmann. Set-Valued Differentials and the Hybrid Maximum Principle. *IEEE Conference on Decision and Control*, Vol. 1, pp. 558 -563, Dec. 2000.

[14] G. Walsh, H. Ye, and L. Bushnell. Stability Analysis of Networked Control Systems. *American Control Conference*, pp. 2876–2880, 1999.

[15] L.Y. Wang, A. Beydoun, J. Cook, J. Sun, and I. Kolmanovsky. Optimal Hybrid Control with Applicatiuons to Automotive Powertrain Systems. In *Control Using Logic-Based Swirtching*, Vol. 222 of LNCIS, pp. 190-200, Springer-Verlag, 1997.

[16] X. Xu and P.J. Antsaklis. An Approach for Solving General Switched Linear Quadratic Optimal Control Problems. In *Proceedings of the 40th CDC*, pp. 2478-2483, 2001.

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