OPTIMAL MODE-SWITCHING FOR HYBRID SYSTEMS WITH UNKNOWN INITIAL STATE

Henrik Axelsson * **Mauro Boccadoro** ** **Yorai Wardi** *
Magnus Egerstedt *

* {***henrik,ywardi,magnus***}***@ece.gatech.edu***
Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332
** ***boccadoro@diei.unipg.it***
Dipartimento di Ingegneria Elettronica e dell'Informazione
Universita di Perugia
06125, Perugia − Italy

Abstract: This paper concerns an optimal control problem defined on a class of switched-mode hybrid dynamical systems. Such systems change modes whenever the state intersects certain surfaces that are defined in the state space. These switching surfaces are parameterized by a finite dimensional vector called the *switching parameter*. The optimization problem we consider is to minimize a given cost-functional with respect to the switching parameter under the assumption that the initial state of the system is not completely known. Instead, we assume that the initial state can be anywhere in a given set. We will approach this problem by minimizing the worst possible cost over the given set of initial states using results from minimax optimization. The results are then applied in order to solve a navigation problem in mobile robotics.

Keywords: Hybrid Systems, Switching Modes, Optimal Control, Gradient Descent, Numerical Algorithms, Minimax Techniques, Mobile Robotics

1. INTRODUCTION

Over the last couple of decades, a lot of effort has been directed towards optimal control of hybrid systems (Branicky *et al.*, 1998; Bemporad *et al.*, 2002; Guia *et al.*, 1999; Hedlund and Rantzer, 1999; Hristu-Varsakelis, 2001; Xu and Antsaklis, 2002; Caines and Shaikh, 2005; Attia *et al.*, 2005). Hybrid systems are complex systems that are characterized by discrete logical decision making at the highest level and continuous variable dynamics at the lowest level. Examples when these systems arise include situations where a control module has to switch its attention among a number of subsystems (Lincoln and Rantzer, 2001; Rehbinder and Sanfridson, 2000; Walsh *et al.*, 1999) or collect data sequentially from a number of sensory sources (Brockett, 1995; Egerstedt and Wardi, 2002; Hristu-Varsakelis, 2001).

The type of hybrid system under consideration in this paper can be described by the following equation

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

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 belonging to some given set A. The time t is confined to a given finite-length interval $[0,T]$. A supervisory controller is normally engaged for dictating the switching law, i.e. the rule for switching among the functions f_α in the right-hand side of (1).

This paper addresses a particular class of hybrid systems, called switched autonomous systems, where the continuous-time control variable is absent and the continuous-time dynamics change at discrete times (*switching-times*). For these systems, the authors have derived gradient expressions for the cost functional with the respect to the switching times when the initial state $x_0 \in \mathbb{R}^n$ is fixed. In particular, (Egerstedt *et al.*, 2006) presented a gradient and an algorithm that finds optimal switching-times, for when to switch between a given set of modes, for the case when the switching-times are controlled directly. Furthermore, (Boccadoro *et al.*, 2005*a*) considered the case when a switch between two different modes occurs when the state trajectory intersects a switching surface, defined by $g(x(t),a)=0$, and parameterized by the parameter a. Reference (Boccadoro *et al.*, 2005*a*) can be thought of as the starting point of this paper, as we consider a similar problem but instead of optimizing with respect to a given fixed initial condition $x_0 \in \mathbb{R}^n$, we will assume that the initial state can be anywhere within a given set $S \subset \mathbb{R}^n$. In order to find a good value of the switching parameter a, independent of where in S we start, we will use the gradient formula presented in (Boccadoro *et al.*, 2005*a*) and find the optimal a such that we will minimize the worst possible cost for all trajectories starting in S. Hence, we have a minimax problem.

At this point, it should be noted that although we will focus on the case where switches occur when the state trajectory intersects a switching surface, the algorithm that will be presented in order to solve the minimax problem would also solve the free switching-time problem with only minor modifications. Hence, this paper presents a way to get rid of the dependence of the initial condition under the assumption that the initial state belongs to a given set.

Once the theoretical underpinnings have been presented, the results will be applied to a navigation problem in mobile robotics.

The robotics problem considered in this paper was also investigated in (Boccadoro *et al.*, 2005*a*), for a fixed initial state. However, for many applications the initial state is not known. An example of this is robotic systems that get their position from a Global Positioning System (GPS). Typically there is a nontrivial error associated with these systems. Hence, if the GPS indicates that the robot is at a point (x,y) the robot can be anywhere within the interval $(x-\Delta, x+\Delta) \times (y-\Delta, y+\Delta)$, for some positive constant Δ. As a result, solving the parameter optimization problem for a given fixed initial state might not give a good solution if the robot's position is given by a GPS.

The outline of this paper is as follows: In Section 2, the problem at hand is introduced together with some previous results relating to the gradient formula. Section 3 presents our solution using a minimax strategy. Simulation results for the robotics application are presented in Section 4 and conclusions are given in Section 5.

2. PROBLEM FORMULATION & PREVIOUS RESULTS

The state trajectory of the underlying system is given by the following equation

$$\dot{x}(t) = f_i(x(t),\ t \in [\tau_{i-1}, \tau_i)),\ i \in \{1, \ldots, N+1\}, \quad (2)$$

where we assume that the system switches N times. The modal functions are chosen from a given set $\{f_\alpha\}_{\alpha \in A}$. However, we assume that the switching times are not controlled directly. Instead, a switch occurs whenever the state trajectory intersects a *switching* surface. This problem was initially considered in (Boccadoro *et al.*, 2005*b*) for a fixed initial state. We will follow the presentation of (Boccadoro *et al.*, 2005*b*) in order to set the stage for our minimax problem when the initial state in not completely known.

We assume that the switching times and the modal functions are determined recursively in the following way. Given f_i and $\tau_{i-1} > 0$ 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. Hence, there might be a restriction on the mode sequence. For every $\alpha \in A(i)$, we let $S_\alpha \subset \mathbb{R}^n$ be the $n-1$ dimensional surface enabling the switch to mode α. Then, the next switch is defined by

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

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 $f_{i+1} = f_{\tilde{\alpha}}$. The system is initialized by setting $\tau_0 = 0$ and choosing what mode the system should start with.

The time when the state trajectory intersects a surface defines τ_i, and the index of the surface $S_{\tilde{\alpha}}$ defines f_{i+1}. In this paper, the surfaces $S_{\tilde{\alpha}}$ are defined by the solution points of parameterized equations from $\mathbb{R}^n$ to $\mathbb{R}$. We denote the parameter by a and suppose that $a \in \mathbb{R}^k$ for some integer $k \geq 1$. For every $\alpha \in A$, we let $g_\alpha : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}$ be a continuously differentiable function. For a given fixed value of $a \in \mathbb{R}^k$, 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 $\mathbb{R}^n$, and a_α can be viewed as a control parameter of the surface. Using the terminology defined earlier, we will replace the index α by i; thus, S_i is the solution set of the equation

$$g_i(x, a_i) = 0, \quad (4)$$

which is parameterized by the control variable $a_i \in \mathbb{R}^k$. To summarize, the system changes dynamics whenever the state trajectory intersect a switching curve $g(x,a) = 0$ parameterized by a control variable a, as illustrated in Figure 1.

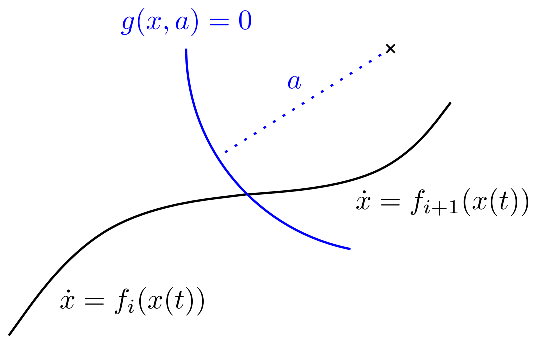

Fig. 1. Mode switching occur when the state trajectory intersect a switching surface. In this case, the switching surface is a circle parameterized by the radius a.

In order to minimize a cost criterion of the form

$$J = \int_0^T L(x(t))dt, \qquad (5)$$

where $L : \mathbb{R}^n \to \mathbb{R}$, we need to determine the optimal switching surface parameters a since the state trajectory depends on a. To this end, (Boccadoro *et al.*, 2005*b*) presented an expression of the gradient of the cost functional with respect to switching surface parameter a. This gradient was presented under the assumption that the functions f_i, g_i, $i = 1,\dots,N+1$, and L where continuously differentiable with respect to all its variables. Furthermore, it was assumed that f_i $i = 1,\dots,N+1$, was uniformly Lipschitz.

We define $x_i = x(\tau_i)$, and the terms R_i and L_i by

$$R_i = f_i(x_i) - f_{i+1}(x_i), \qquad (6)$$

and

$$L_i = \frac{\partial g_i}{\partial x}(x_i,a_i) f_i(x_i), \qquad (7)$$

where we recognize L_i as the Lie derivative of g_i in the direction of f_i.

Now, in order to ensure that the gradient exists, the following assumption is presented;

Assumption 1. For all $i = 1,\dots,N$, $L_i \neq 0$.

Given Assumption 1, reference (Boccadoro *et al.*, 2005*b*) derived the following expression for the derivative $\frac{dJ}{da_i}$.

Proposition 2.1. The following equation is in force,

$$\frac{dJ}{da_i} = -\frac{1}{L_i} p(\tau_i^+) R_i \frac{\partial g_i}{\partial a_i}(x_i,a_i). \qquad (8)$$

where the costate equation is given by

$$\dot{p}(t) = -\left(\frac{\partial f_{i+1}}{\partial x}(x(t),t)\right)^T p(t) - \left(\frac{\partial L}{\partial x}(x(t))\right)^T; \quad t \in [\tau_i, \tau_{i+1}),\ i = 1,\dots,N, \qquad (9)$$

with terminal condition $p^T(t_N) = 0$ when the final time is fixed, and reset conditions

$$p(\tau_i^-) = (I - \frac{1}{L_i} R_i \frac{\partial g_i}{\partial x}(x_i,a_i))^T p(\tau_i^+), \quad i = 1,\dots,N. \qquad (10)$$

Proof: See (Boccadoro *et al.*, 2005*b*).

Having presented the expression for the gradient, as derived in (Boccadoro *et al.*, 2005*b*), we can now proceed to present the minimax solution to our switching surface parametrization problem.

3. MINIMAX OPTIMIZATION

Given a set of possible initial points $S \subset \mathbb{R}^n$, a set of switching surfaces parameterized by some vector a, and an instantaneous cost L, the total cost, starting at $x_0 \in S$, is given by

$$J_{x_0}(a) = \int_0^T L(x(t))dt, \qquad (11)$$

where T is a fixed final time and subscript x_0 indicates the initial condition. Our problem, denoted by P_S, can be stated as

P_S: Given a set of initial states S and a set of switching surfaces parameterized by a, find the surface parameter a such that

$$\max\{J_x(a) \mid x \in S\} \qquad (12)$$

is minimized.

As mentioned earlier, the theory of minimax optimization and consistent approximations (Polak, 1997) will be utilized in order to implement and solve this problem.

Given a set of possible initial states $S \subset \mathbb{R}^n$, we will choose a sequence of sets of initial points, $\{\mathbb{X}_i\}_{i=0}^{\infty}$. This sequence will satisfy the following three conditions: Firstly, $\mathbb{X}_i \subset S$ $i = 1,2\dots$; secondly, the number of elements in $\mathbb{X}_i$ is bigger than the number of elements in $\mathbb{X}_{i-1}$; thirdly, every point in S will be arbitrarily close to a point in $\mathbb{X}_i$, as i goes to infinity. Choosing $\{\mathbb{X}_i\}_{i=0}^{\infty}$ in this way enables us to find the solution to (12) by solving a sequence of optimization problems, each one with a different set of initial states.

For each $\mathbb{X}_i$ we will find the optimal switching parameter a_i^o that minimizes $\max\{J_x(a_i) \mid x \in \mathbb{X}_i\}$ through a gradient descent algorithm, as described below. After we have found the optimal a_i^o, we will solve $\max\{J_x(a_{i+1}) \mid x \in \mathbb{X}_{i+1}\}$ by initializing a_{i+1} to a_i^o. This gives a good starting point for the gradient descent algorithm.

For each $\mathbb{X}_i$ we will find the optimal a_i^o by executing the following gradient descent algorithm with Armijo step size (Armijo, 1966). We assume that $\mathbb{X}_i$ have $N(i)$ elements, i.e. $\mathbb{X}_i = \{x_1,\dots,x_{N(i)}\}$ for some $x_1,\dots,x_{N(i)}$ in $S \subset \mathbb{R}^n$.

Algorithm 3.1 Gradient Projection Algorithm with Armijo Stepsize

Given: The Armijo constants α, β in $(0,1)$. Two constants $\delta > 0$, and $\varepsilon > 0$ and the set of initial points $\mathbb{X} = \{x_1, \ldots, x_N\} \subset S$.
Initialize: Choose a feasible initial guess on the switching surface parameter a.
Step I: Calculate the maximum cost for the given set of initial states, denoted

$$F(\mathbb{X}, a) = \max_x \{J_x(a) | x \in \mathbb{X}\}, \tag{13}$$

where J_x is given by (11). Let $I(\mathbb{X}, a)$ denote the index set of *active constraints*, i.e.

$$I(\mathbb{X}, a) = \{j \in \{1, \ldots, N\} \mid F(\mathbb{X}, a) - J_j(a) < \varepsilon\}. \tag{14}$$

Calculate the generalized gradient

$$\partial F(\mathbb{X}, a) = conv\{\nabla J_j(a) \mid j \in I(\mathbb{X}, a))\}, \tag{15}$$

where *conv* denotes the *convex hull*. Find the point in $\partial F(\mathbb{X}, a)$ closest to the origin and denote it by h. If $||h|| < \delta$ then STOP. Else, goto Step II.
Step II: Calculate the step-length λ according to Armijo's rule i.e.

$$\begin{aligned} \lambda &= \max\{z = \beta^k;\ k \geq 0 \mid \\ &F(\mathbb{X}, a - zh) - F(\mathbb{X}, a) \leq -\alpha z ||h||^2\}. \end{aligned}$$

Update a according to $a = a - \lambda h$, goto Step I. ■

A few remarks concerning Algorithm 3.1 are due.

Remark 3.1. The index set of active constraints, $I(\mathbb{X}, a)$, is introduced in order to determine what initial states in $\mathbb{X}$ we should take into consideration for a given a. If the index of an initial state is in the index set, then the gradient of the cost associated with that initial state is current in the calculation of the generalized gradient, $\partial F(\mathbb{X}, a)$. If $\varepsilon = 0$ in (14), i.e., we only optimize with respect to the initial state corresponding to the maximal cost, it is conceivable that we can only take a very small descent step since the index set changes when a changes.

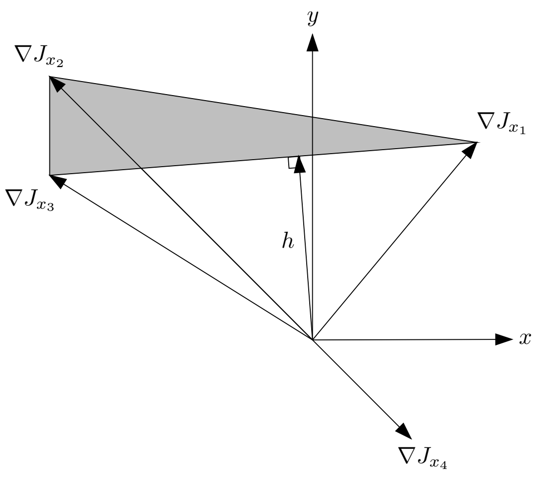

Fig. 2. Calculation of h given four initial states and their respective gradients. x_1 through x_3 are active initial states.

Remark 3.2. In order to find the optimal a for a given set of initial states, we would have to set the constants δ and ε to 0. However, doing this when we solve for a sequence of initial states, $\{\mathbb{X}_i\}_{i=0}^{\infty}$, would not give any additional benefit, instead we only require that for each consecutive problem we will solve, δ and ε will decrease, and in the limit when $i \to \infty$, they will be zero.

Remark 3.3. Solving for h is a standard quadratic optimization problem over a convex set, and can be solved using a variety of optimization algorithms.

Remark 3.4. In the robotics example presented in Section 4, a simple constraint is introduced on a. Hence we need to initialize a to be in the set of feasible points.

In order to illustrate the calculation of h, a simple example is presented. Assume that we have four different initial states, x_1 through x_4 in $\mathbb{R}^2$. In Figure 2, their respective gradients are plotted and it is assumed that x_1 through x_3 are active initial states for the given switching surface parameter a. The shaded region in Figure 2 corresponds to the convex hull of the gradients of the active initial states, and h is the closest vector in this set from the origin.

Having presented Algorithm 3.1 and the remarks that follow it, we are now in the position to present Algorithm 3.2 that will solve problem P_S.

Algorithm 3.2 Minimax optimization for unknown initial state:
Given: A sequence of initial sets $\{\mathbb{X}_i\}_{i=0}^{\infty} \in S \subset \mathbb{R}^n$, where $\mathbb{X}_i = \{x_1, \ldots, x_{N(i)}\}$ and $N(i) > N(i-1)$. Two positive sequences $\{\varepsilon_i\}_{i=0}^{\infty}$ and $\{\delta_i\}_{i=0}^{\infty}$ such that in the limit when $i \to \infty$, both are 0.
Init: Set $i = 0$, pick a feasible initial guess on a_0.
Step I: Use Algorithm 3.1 to optimize over a with $\mathbb{X} = \mathbb{X}_i$, $\delta = \delta_i$, $\varepsilon = \varepsilon_i$. Initialize a with a_{i-1} if $i \neq 0$, and with a_0 if $i = 0$.
Step II: Set a_i to a given from Algorithm 3.1. Increase i by one, goto Step I.

4. NUMERICAL EXAMPLE

In order to show the usefulness of Algorithm 3.2, we consider a mobile robot navigation problem. The task of the robot is to get to a goal point $x_g \in \mathbb{R}^2$ while avoiding an obstacle located at $x_{ob} \in \mathbb{R}^2$. It has to do this by switching between two different behaviors, one *go-to-goal* and one *obstacle-avoidance* behavior. These different behaviors are denoted by f_g and f_o respectively. We model the robot having unicycle dynamics

$$\begin{cases} \dot{x_1} = v\cos(\phi), \\ \dot{x_2} = v\sin(\phi), \\ \dot{\phi} = f_q(x_1, x_2, \phi), \end{cases} \tag{16}$$

where (x_1, x_2) is the position of the robot, ϕ is its heading, and $q \in \{g, o\}$ is the current behavior the robot evolves according to. We assume that the translational velocity v is constant. Our control variable is then given by the switching surface parameters of the goal and avoid obstacle guards that dictate what behavior the robot should evolve according to. A standard pair of "approach-goal" and "avoid-obstacle" behaviors are given by

$$f_g(x_1, x_2, \phi) = \quad c_g(\phi_g - \phi), \tag{17}$$
$$f_o(x_1, x_2, \phi) = c_o(\pi + \phi_{ob} - \phi). \tag{18}$$

Here, c_g and c_o are the gains associated with each behavior, and ϕ_g and ϕ_{ob} are the angles to the goal and nearest obstacle respectively. Both of these angles are measured with respect to the x-axis and can be expressed as

$$\phi_g = \arctan(\frac{x_{g_2} - x_2}{x_{g_1} - x_1}), \tag{19}$$
$$\phi_{ob} = \arctan(\frac{x_{ob_2} - x_2}{x_{ob_1} - x_1}), \tag{20}$$

where (x_{g_1}, x_{g_2}) and (x_{ob_1}, x_{ob_2}) are the Cartesian coordinates of the goal and the nearest obstacle respectively.

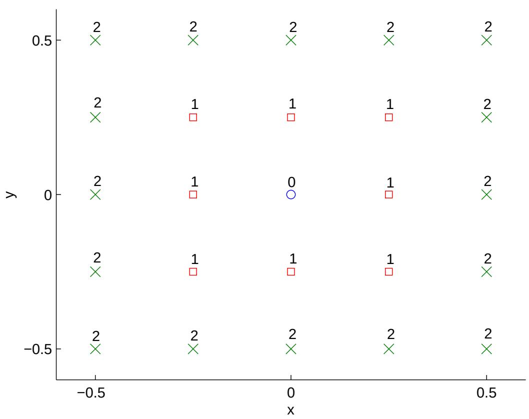

Fig. 3. Initial states used: $\mathbb{X}_i$ contains the points with index $i, i-1, \ldots, 0$.

The instantaneous cost L is given by

$$L(x(t)) = \rho ||x_g - x(t)||^2 + \alpha e^{-\frac{||x_{ob} - x(t)||^2}{\beta}}, \tag{21}$$

where ρ is the gain of the goal attraction term, α is the gain of the obstacle avoidance term, and β is a shaping parameter that affects the range of the obstacle avoidance term.

For a given initial position $x_0 \in \mathbb{R}^3$ the total cost is given by (11). However, many mobile robots get their position from GPS readings which has an error associated with them. In our example, we assume that the robot get the initial position $x_0 = (0, 0, \cdot)^T$ from the GPS and that the error associated with the GPS is 0.5 meters (note that GPS do not give the direction of a stationary robot). In order to simplify our exposition, we assume that the robot is always directed towards the goal, hence we will only show the (x_1, x_2) components in $\mathbb{X}_i$, $i = 0, 1, 2$. This is a reasonable assumption if the robot can see the goal, which we assume.

Due to the error in the GPS reading, the robot can be anywhere in the interval $[-0.5, -0.5] \times [0.5, 0.5]$. Therefore we initialize Algorithm 3.2 with only one initial state, $\mathbb{X}_0 = (0, 0)^T$, and we then extend the set of initial states, in a somewhat arbitrary fashion, as shown in Figure 3. In this example, we stop the algorithm after its third iteration, i.e. when $||h|| < \delta_2$, therefore we do not define $\mathbb{X}_i$ for $i = 3, 4, \ldots$.

The switching surfaces for when to switch from f_g to f_o, and when to switch from f_o to f_g, are given by two circles with radius a_1 and a_2 respectively, where we require $a_1 \leq a_2$. Both circles are centered at the obstacle $x_o = (2, 1.25)^T$. At this point it should be noted that having circular guards might not correspond to an optimal guard shape.

We initialize a to be $(1, 1.5)^T$ and for the constants in L, we set $\rho = 0.01$, $\alpha = 10$ and $\beta = 0.1$ and we use $c_g = c_o = 1$ for the feedback gains in (17) and (18). The velocity of the robot is set to $v = 0.5$ and the goal is located at $x_g = (4, 4)^T$. For the constants in the Armijo procedure, we use $\alpha = \beta = 0.5$. The sequences of ε_j and δ_j used is given by $\delta_j = \frac{\delta_{j-1}}{2.5}$ with $\delta_0 = 0.25$, and $\varepsilon_j = \frac{\varepsilon_{j-1}}{2.5}$ with $\varepsilon_0 = 0.1$

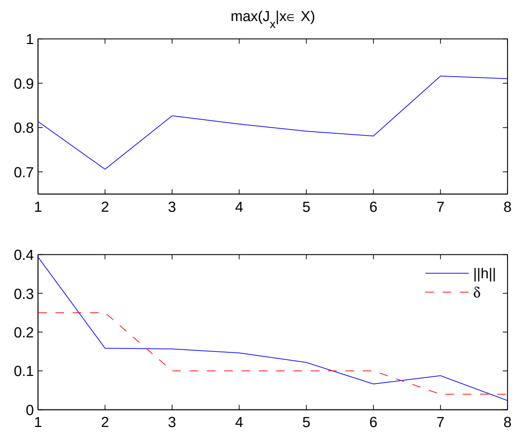

Fig. 4. (a) Change in maximum cost. (b) $||h||$ and δ as a function of the number of gradient descent iterations in Algorithm 3.2.

A plot of how the cost changes together with the norm of h and δ is shown in Figure 4. As can be seen in the figure, Algorithm 3.2 effectively reduces the maximum cost for a given set of initial states. Once the norm of h falls below δ, we update δ, ε and the set of initial states, $\mathbb{X}$.

Once we have updated $\mathbb{X}_0$ to $\mathbb{X}_1$ after iteration three, we see that the maximum of the cost increases, just as should be expected since $\mathbb{X}_1$ has more initial states that $\mathbb{X}_0$. Figure 5(a) shows how the switching surface parameters change. At the optimum, $a_1 = a_2$, i.e. both radii are the same.

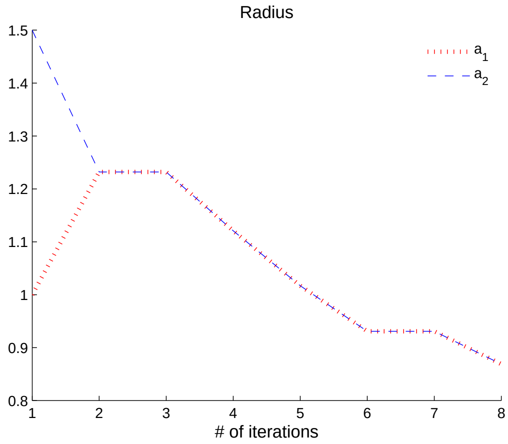

Fig. 5. Change in $a = (a_1, a_2)^T$ as a function of the number of gradient descent iterations in Algorithm 3.2. a_1 is the radius of the *obstacle-avoidance* switching surface, a_2 is the radius of the *go-to-goal* switching surface.

5. CONCLUSIONS

This paper presented a way of getting rid of the dependence on the initial condition when optimizing over when to switch between different modes in a switched-mode system. The dependence on the initial condition was dealt with by minimizing the switching parameter over the maximum cost for a given set of initial states. The only assumption made was that the initial state was confined to a given region in the state space.

REFERENCES

Armijo, L. (1966). Minimization of functions having lipschitz continuous first-partial derivatives. *Pacific Journal of Mathematics* **16**, 1–3.

Attia, S.A., M. Alamir and C. Canudas de Wit (2005). Sub optimal control of switched nonlinear systems under location and switching constraints. In: *IFAC World Congress*.

Bemporad, A., F. Borrelli and M. Morari (2002). Piecewise Linear Optimal Controllers for Hybrid Systems. In: *Hybrid Systems: Computation and Control*, M. Greenstreet and C. Tomlin, Editors, Springer-Verlag Lecture Notes in Computer Science, Number 2289. pp. 105–117.

Boccadoro, M., M. Egerstedt and Y. Wardi (2005*a*). Obstacle avoidance for mobile robots using switching surface optimization. *IFAC World Congress*, Prague, The Czech Republic.

Boccadoro, M., Y. Wardi, M. Egerstedt and E. Verriest (2005*b*). Optimal control of switching surfaces in hybrid dynamical systems. *Journal of Discrete Event Dynamic Systems* **15**, 433–448.

Branicky, M.S., V.S. Borkar and S.K. Mitter (1998). A unified framework for hybrid control: Model and optimal control theory. *IEEE Transactions on Automatic Control* **43**, 31–45.

Brockett, R. (1995). Stabilization of motor networks. In: 35^{th} *IEEE Conference on Decision and Control*. pp. 1484–1488.

Caines, P. E. and M. S. Shaikh (2005). Optimality zone algorithms for hybrid systems computation and control: From exponential to linear complexity. In: *Proceedings of the 2005 International Symposium on Intelligent Control/ 13th Mediterranean Conference on Control and Automation*, Cyprus. pp. 1292–1297.

Egerstedt, M. and Y. Wardi (2002). Multi-process control using queuing theory. In: 41^{th} *IEEE Conference on Decision and Control*, Las Vegas, NV.

Egerstedt, M., Y. Wardi and H. Axelsson (2006). Transition-time optimization for switched-mode dynamical systems. *IEEE Transactions on Automatic Control* **51**, to appear.

Guia, A., C. Seatzu and C. Van der Mee (1999). Optimal control of switched autonomous linear systems. In: 38^{th} *IEEE Conference on Decision and Control*, Phoenix, AR. pp. 1816–1821.

Hedlund, S. and A. Rantzer (1999). Optimal control of hybrid systems. In: 38^{th} *IEEE Conference on Decision and Control*, Phoenix, AR. pp. 1972–3977.

Hristu-Varsakelis, D. (2001). Feedback control systems as users of shared network: Communication sequences that guarantee stability . In: 40^{th} *IEEE Conference on Decision and Control*, Orlando, FL. pp. 3631–3631.

Lincoln, B. and A. Rantzer (2001). Optimizing linear systems switching. In: 40^{th} *IEEE Conference on Decision and Control*, Orlando, FL. pp. 2063–2068.

Polak, E. (1997). *Optimization Algorithms and Consistent Approximations*. Springer-Verlag. New York, New York.

Rehbinder, H. and M. Sanfridson (2000). Scheduling of a limited communication channel for optimal control. In: 38^{th} *IEEE Conference on Decision and Control, Sidney, Australia*.

Walsh, G., H. Ye and L. Bushnell (1999). Stability analysis of networked control systems. In: *American Control Conference*. pp. 2876–2880.

Xu, X. and P. Antsaklis (2002). Optimal control of switched autonomous systems. In: 41^{th} *IEEE Conference on Decision and Control*, Las Vegas, NV.