A Master-Slave Algorithm for the Optimal Control of Continuous-Time Switched Affine Systems

Alberto Bemporad
Dip. Ing. dell'Informazione, Università di Siena
Via Roma 56, 53100 Siena, Italy
bemporad@dii.unisi.it

Alessandro Giua, Carla Seatzu
Dip. di Ing. Elettrica ed Elettronica, Università di Cagliari
Piazza d'Armi, 09123 Cagliari, Italy
{giua,seatzu}@diee.unica.it

Abstract

For continuous-time switched affine systems, this paper proposes an approach for solving infinite-horizon optimal control problems where the decision variables are the switching instants and the sequence of operating modes. The procedure iterates between a "master" procedure that finds an optimal switching sequence of modes, and a "slave" procedure that finds the optimal switching instants.

Published as:
A. Bemporad, A. Giua, C. Seatzu, "A master-slave algorithm for the optimal control switched affine systems," *Proc. IEEE 41st Int. Conf. on Decision and Control* (Las Vegas, Nevada), pp. 1976-1981, December 2002.

1 Introduction

Switched systems are a particular class of hybrid systems that switch between many operating modes, where each mode is governed by its own characteristic dynamical law [8]. Typically, mode transitions are triggered by variables crossing specific thresholds (state events), by the elapse of certain time periods (time events), or by external inputs (input events). The problem of determining optimal control laws for this class of hybrid systems has been widely investigated in the last years and many results can be found in the control and computer science literature. For continuous-time hybrid systems, most of the literature is focused on the study of necessary conditions for a trajectory to be optimal [15], and on the computation of optimal/suboptimal solutions by means of dynamic programming or the maximum principle [6, 7, 11, 12, 16, 19].

The hybrid optimal control problem becomes less complex when the dynamics is expressed in discrete-time or as discrete-events [1, 5]. In such a case, the main source of complexity is the combinatorial number of possible switching sequences. By combining reachability analysis and quadratic optimization, in [2] we proposed a technique that rules out switching sequences that are either not optimal or simply not compatible with the evolution of the dynamical system. An algorithm to optimize switching sequences that has an arbitrary degree of suboptimality was presented in [13].

In the continuous-time case, and in particular for switched linear systems composed by stable autonomous dynamics, by assuming that the switching sequence is preassigned (and, therefore, that the only decision variables to be optimized are the switching instants), in [9, 10] we proved an important result: the control law is a state-feedback and there exists a numerically viable procedure to compute the switching regions $\mathcal{C}_{k,N}$, i.e., the points of the state space where the k-th switch of a sequence of length N should occur.

In this paper we solve the optimal control problem for continuous-time switched affine systems in which both the switching instants and the switching sequence are decision variables. The procedure exploits a synergy of discrete-time and continuous-time techniques, by alternating between a "master" procedure that finds an optimal switching sequence and a "slave" procedure that finds the optimal switching instants. A few simple heuristics can be added to the algorithm to improve its performance. Although the final optimal switching policy is computed for a given initial state, as a by-product of the algorithm it has a state-feedback nature, which is only valid however for "small" perturbations of the initial state such that the optimal switching sequence does not change. A related approach that optimizes hybrid processes by combining mixed-integer linear programming (MILP) to obtain a candidate switching sequence and dynamic simulation was proposed in [14]. A two-stage procedure which exploits the derivatives of the optimal cost with respect to the switching instants was proposed in [19].

Although we formally prove that the algorithm always converges to a local minimum, the global minimum is not always reached. We run the algorithm on a large number of random tests and observed that it converges extremely quickly to the global optimum on most of the problems.

The paper extends previous results appeared in [3] by allowing arbitrary affine dynamics (non-asymptotically stable transition matrices and constant perturbation terms).

A different solution to the same optimal control problem has been proposed by the authors in [4], that is inspired by dynamic programming and is based on the construction of switching tables. In such a case the global optimum is guaranteed and the procedure always provides a closed loop solution. Nevertheless, the computational complexity of the off-line part is significantly larger. A detailed comparison among the two approaches is given in [4].

2 Problem Formulation

In this paper we consider the following class of hybrid systems

$$\dot{x}(t) = A_{i(t)}x(t) + f_{i(t)},\ i(t) \in \mathcal{S} \tag{1}$$

that we denote as *switched affine systems*, where $x(t) \in \mathbb{R}^n$, $i(t) \in \mathcal{S}$ is the control variable, and $\mathcal{S} \triangleq \{1, 2, \ldots, s\}$ is a finite set of integers, each one associated with an affine dynamics.

For such a class of hybrid systems we want to solve the following optimal control problem

$$\begin{array}{ll} V_N^* \triangleq \min\limits_{I,\mathcal{T}} & \left\{ F(I,\mathcal{T}) \triangleq \displaystyle\int_0^\infty x'(t)Q_{i(t)}x(t)dt \right\} \\ \text{s.t.} & \dot{x}(t) = A_{i(t)}x(t) + f_{i(t)} \\ & x(0) = x_0 \\ & i(t) = i_k \text{ for } \tau_{k-1} \leq t < \tau_k \\ & i_k \in \mathcal{S},\ k = 1, \ldots, N+1 \\ & \tau_0 = 0,\ \tau_{N+1} = +\infty \\ & \tau_k \in \mathbb{R}_{\geq 0}\ \forall k = 1, \ldots, N \end{array} \tag{2}$$

where N is the maximum allowed number of switches (fixed a priori), $\mathcal{T} \triangleq \{\tau_1, \ldots, \tau_N\}$ is a finite sequence of switching times, $I \triangleq \{i_1, \ldots, i_{N+1}\}$ is a finite sequence of switching indices, and x_0 is the initial state of the system. We assume that Q_i is a positive definite weight matrix associated with the i-th dynamics, for all $i \in \mathcal{S}$. We denote by $i^*(t)$, $t \in [0, +\infty)$, $i^*(t) = i_k^*$ for $\tau_{k-1}^* \leq t < \tau_k^*$ the switching trajectory solving (2), and I^*, $\mathcal{T}^*$ the corresponding optimal sequences.

In order to make the problem solvable with finite cost V_N^*, we assume the following:
Assumption 1. *There exists at least one index $i \in \mathcal{S}$ such that A_i is strictly Hurwitz and $f_i = 0$.*

The optimal control problem may be easily generalized. One may assume that whenever at time τ_k a switch from i_k to i_{k+1} occurs, the state should jump from $x(\tau_k^-)$ to $x(\tau_k^+) = M_k x(\tau_k^-)$ as in [10]. One may also assume that a cost is associated to each switch as in [9]. However, to avoid heavy notation in this paper we only restrict to the basic framework (2).

The optimal control problem (2) may also be rewritten as:

$$\begin{aligned} \min_{I,\mathcal{T}} \; & \left\{ \sum_{k=1}^{N+1} x'_{k-1}\bar{Q}_{i_k}(k)x_{k-1} + \bar{c}_{i_k}(k)x_{k-1} + \bar{\alpha}_{i_k}(k) \right\} \\ \text{s.t.} \;\; & x_k = \bar{A}_{i_k}(k)x_{k-1} + \bar{f}_{i_k},\; k = 1, \ldots, N+1 \\ & x_0 = x(0) \end{aligned} \tag{3}$$

where

$$\begin{aligned} \bar{A}_i(k) &\triangleq e^{(\tau_k - \tau_{k-1})A_i}, \\ \bar{f}_i(k) &\triangleq \int_{\tau_{k-1}}^{\tau_k} e^{(\tau_k - t)A_i} dt \end{aligned} \tag{4}$$

and $\bar{Q}_i(k)$, $\bar{c}_i(k)$, $\bar{\alpha}_i(k)$ can be obtained by simple integration and linear algebra [1].

Consider a decomposition of (3) into the following "master" and "slave" subproblems:
Problem 1 (Master)**.** *For a fixed sequence of switched times $\bar{\tau}_1, \ldots, \bar{\tau}_N$, solve the optimal control problem (3) with respect to $i_1, \ldots, i_{N+1}$. Denote by*

$$\{i_1, \ldots, i_{N+1}\} = f_M(\bar{\tau}_1, \ldots, \bar{\tau}_N) \tag{5}$$

and $V_M(\bar{\tau}_1, \ldots, \bar{\tau}_N)$ the optimizing index sequence and optimal value, respectively.
Problem 2 (Slave)**.** *For a fixed sequence of switching indices $\bar{\imath}_1, \ldots, \bar{\imath}_{N+1}$, solve the optimal control problem (3) with respect to $\tau_1, \ldots, \tau_N$. Denote by*

$$\{\tau_1, \ldots, \tau_N\} = f_S(\bar{\imath}_1, \ldots, \bar{\imath}_{N+1}) \tag{6}$$

and $V_S(\bar{\imath}_1, \ldots, \bar{\imath}_{N+1})$ the optimizing timing sequence and optimal value, respectively.

---

[1] In [3] we assumed A_i asymptotically stable and simply expressed $\bar{Q}_i(k) = Z_i - \bar{A}'_i(k)Z_i\bar{A}_i(k)$, where Z_i is the unique solution of the Lyapunov equation $A'_iZ_i + Z_iA_i = -Q_i$. The same computation is valid when the eigenvalues of A_i are all unstable. On the other hand, in this paper we are not making any assumptions on the eigenvalues of A_i, and moreover we have the affine terms f_i. We can still compute $\bar{Q}_i(k)$ by using direct integration. Indeed, if for simplicity we assume that A_i is diagonalizable, $A_i = T_i^{-1}\Lambda_iT_i$, $\Lambda_i = \text{Diag}\{\lambda_i^1, \ldots, \lambda_i^n\}$, and $f_i = 0$, we obtain

$$\int_{\tau_{k-1}}^{\tau_k} x'(t)Q_{i_k}x(t)dt = \\ x'_{k-1} \underbrace{T'_{i_k}\left(\int_{\tau_{k-1}}^{\tau_k} e^{\Lambda_{i_k}t}(T_{i_k}^{-1})'Q_{i_k}T_{i_k}^{-1}e^{\Lambda_{i_k}t}dt\right)T_{i_k}}_{\bar{Q}_{i_k}} x_{k-1}.$$

The case of A_i not diagonalizable and $f_i \neq 0$ is similar.

3 Master Algorithm

For a fixed sequence of switched times $\bar{\tau}_1, \ldots, \bar{\tau}_N$, the master algorithm solves the optimal control problem (3) with respect to $i_1, \ldots, i_{N+1}$. It is a purely combinatorial problem that can be rephrased as:

$$\begin{aligned} \min_{i_k \in \mathcal{S}} \quad & \left\{ \sum_{k=1}^{N+1} x'_{k-1} \bar{Q}_{i_k}(k) x_{k-1} + \bar{c}_{i_k}(k) x_{k-1} + \bar{\alpha}_{i_k}(k) \right\} \\ \text{s.t.} \quad & x_k = \bar{A}_{i_k}(k) x_{k-1} + f_{i_k}(k),\ k = 1, \ldots, N+1 \\ & x_0 = x(0). \end{aligned} \tag{7}$$

Problem (7) can be efficiently solved via Mixed-Integer Quadratic Programming (MIQP) (see e.g. [17] or the free Matlab solver available at `http://control.ethz.ch/~hybrid/miqp`). To this end, we need to introduce binary variables $\gamma_i^k \in \{0, 1\}$ and continuous variables $z_i^k \in \mathbb{R}^n$, $i \in \mathcal{S}$, $k = 1, \ldots, N+1$, where

$$[\gamma_i^k = 1] \quad \leftrightarrow \quad [i(k) = i], \tag{8a}$$

$$\forall k = 1, \ldots, N+1,\ \forall i \in \mathcal{S} \tag{8b}$$

$$z_i^{k+1} = (\bar{A}_i(k) x_{k-1} + \bar{f}_i(k)) \gamma_i^k, \tag{8c}$$

$$\forall k = 1, \ldots, N,\ \forall i \in \mathcal{S} \tag{8d}$$

$$z_i^1 = x_0 \gamma_i^1,\ \forall i \in \mathcal{S} \tag{8e}$$

$$x_k = \sum_{i=1}^{s} z_i^{k+1},\ \ \forall k = 0, \ldots, N \tag{8f}$$

$$\bigoplus_{i=1}^{s} \gamma_i^k = 1,\ \ \forall k = 1, \ldots, N+1 \tag{8g}$$

$$\bigoplus_{i \in \mathcal{S}_{as}} \gamma_i^{N+1} = 1, \tag{8h}$$

where the exclusive-or constraint (8g) follows by the fact that only one dynamics can be active in each interval k, and in (8h) $\mathcal{S}_{as}$ is the set of indices $i \in \mathcal{S}$ such that A_i is strictly Hurwitz and $f_i = 0$, so that the last dynamics to be asymptotically stable and linear.

Constraints (8d)-(8e) can be transformed into the following set of mixed-integer linear inequalities by using the so-called "big-M" technique (see e.g. [5, 18] for details):

$$z_i^k \leq M \gamma_i^k,\ \ \forall k = 1, \ldots, N+1 \tag{9a}$$

$$-z_i^k \leq M \gamma_i^k,\ \ \forall k = 1, \ldots, N+1 \tag{9b}$$

$$z_i^{k+1} \leq \bar{A}_i(k) x_{k-1} + \bar{f}_i(k) + M(1 - \gamma_i^k), \tag{9c}$$

$$\forall k = 1, \ldots, N \tag{9d}$$

$$-z_i^{k+1} \leq -\bar{A}_i(k) x_{k-1} - \bar{f}_i(k) + M(1 - \gamma_i^k), \tag{9e}$$

$$\forall k = 1, \ldots, N \tag{9f}$$

$$z_i^1 \leq x_0 + M(1 - \gamma_i^1) \tag{9g}$$

$$-z_i^1 \leq -x_0 + M(1 - \gamma_i^1) \tag{9h}$$

for all $i \in \mathcal{S}$, where $M \in \mathbb{R}^n$ is an upper bound on the state vector x (more precisely, the j-th component M^j of M is an upper bound on $|x^j|$, where x^j is the j-th component of the state vector), and therefore an upper bound on x_0 and on $A_i(k)x_{k-1} + \bar{f}_i(k) = x_k$, for all $k = 2, \dots, N+1$, $i \in \mathcal{S}$. Usually M can be estimated on the basis of physical considerations on the hybrid system. Constraints (8g)–(8h) can be instead expressed as

$$\begin{aligned} &\sum_{i=1}^{s} \gamma_i^k = 1, \ \ \forall k = 1, \dots, N+1 \\ &\gamma_i^{N+1} = 0, \ \ \forall i \notin \mathcal{S}_{as}. \end{aligned} \tag{10}$$

Summing up, the master problem (7) is equivalent to the MIQP

$$\begin{aligned} \min_{\substack{x_k, \gamma_i^k, z_i^k \\ k=1,\dots,N+1 \\ i=1,\dots,s}} \quad & \sum_{k=1}^{N+1} \sum_{i=1}^{s} \Big[(z_i^k)' \bar{Q}_i(k) z_i^k + \\ & \qquad \bar{c}_i(k) z_i^k + \bar{\alpha}_i(k) \gamma_k^i \Big] \\ \text{s.t.} \quad & \text{(8f), (9), (10).} \end{aligned} \tag{11}$$

4 Slave Algorithm

For a fixed sequence of switching indices $\bar{\imath}_1, \dots, \bar{\imath}_{N+1}$, the slave algorithm solves the optimal control problem (3) with respect to $\tau_1, \dots, \tau_N$.

A solution to this problem where the switching sequence is pre-assigned and the system dynamics are all asymptotically stable and linear, was already presented in [10]. In particular, in [10] it was shown that the optimal control law turns out to be a "homogeneous feedback", in the sense that for all $k \le N$: (a) it is possible to identify a region $\mathcal{C}_{k,N}$ of the state space such that the k–th switch should occur if and only if we are within this region; (b) this region is homogeneous, i.e., if $x \in \mathcal{C}_{k,N}$, then $\lambda x \in \mathcal{C}_{k,N}$, for all real numbers λ.

We have also provided an algorithmic way to construct the regions. In fact, we observed that it is sufficient to determine which points on the unitary semi–sphere belong to a region to completely determine the region itself (because it is a homogeneous space). In [10] we have also shown that these *switching regions* can be computed starting from the last one.

Here, we first show that all these results can be easily extended to the case of linear systems whose dynamics may also be *unstable*, with the only requirement that there exists at least one admissible dynamics that is strictly Hurwitz. In such a case the residual cost from the k-th to the N-th switch, given a state x, may be written as:

$$F_k(x, \delta_k, \delta_{k+1}, \dots, \delta_N) = \textstyle\sum_{j=k}^{N+1} x_{j-1}' \bar{Q}_{i_j}(j) x_{j-1} \tag{12}$$

where $\delta_j = \tau_j - \tau_{j-1}$ is the j–th switching interval and $x_{k-1} = x$.

The analytical expressions of $\bar{Q}_{i_j}$'s are not reported here for brevity, but they can be easily computed by following the same reasoning of footnote 1 at page 4. Moreover, the last system dynamics will always be chosen stable, thus $\bar{Q}_{i_{N+1}}(N+1) = Z_{i_{N+1}}$ where $Z_{i_{N+1}}$ is the unique solution to the Lyapunov equation $A'_{i_{N+1}} Z_{i_{N+1}} + Z_{i_{N+1}} A_{i_{N+1}} = -Q_{i_{N+1}}(N+1)$. The other $\bar{Q}_{i_j}$'s only depend on the j-th system dynamics and on the values of δ_r for $r = j+1, \cdots, N$. Thus, we may conclude that F_k is a quadratic function of x.

As in [10], we may also define the corresponding k-th optimal switching interval as:

$$\delta_k^*(x) = \arg \min_{\delta_k \in \mathbb{R}_0^+} F_k(x, \delta_k, \delta_{k+1}^*(x_k), \ldots, \delta_N^*(x_{N-1}))$$

where $x_j = e^{A_j \delta_j^*(x_{j-1})} x_{j-1}$. Finally we obtain

$$\mathcal{C}_{k,N} = \{x \mid \delta_k^*(x) = 0\} \quad k = 1, \ldots, N. \tag{13}$$

We also observe that for all $\lambda \in \mathbb{R}$, all $k \in \mathbb{N}$ and all $x \in \mathbb{R}^n$: $\delta_k^*(\lambda x) = \delta_k^*(x)$. Thus, by repeating the same reasoning of [10], we can conclude that the regions are still homogeneous. This implies that we may use the same procedure of [10] for the computation of the switching regions: we choose a suitable discretization step and for each point x on the unitary semi–sphere, determine if it belongs to $\mathcal{C}_{N,N}$, $\mathcal{C}_{N-1,N}$, etc., also computing step by step the corresponding values of the remaining cost. The output of this procedure is the set of switching regions. To determine the optimal switching instants, the evolution of the system is simulated starting from the initial state x_0, by switching as soon as the next switching region is reached.

Let us finally consider the most general case of *affine dynamics*. The previous approach (homogeneous regions) remains valid, because we can rewrite the original affine dynamics as a linear dynamics by augmenting the state space from $\mathbb{R}^n$ to $\mathbb{R}^{n+1}$:

$$\frac{d}{dt} \begin{bmatrix} x(t) \\ \tilde{x}(t) \end{bmatrix} = \begin{bmatrix} A_{i(t)} x(t) & f_{i(t)} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \tilde{x}(t) \end{bmatrix}.$$

Note that the $(n+1)$-th state variable $\tilde{x}(t)$ is a fictitious variable that is not taken account in the computation of the cost function. This implies that Assumption 1 should only hold for the original system and not for the augmented one.

5 Master-Slave Algorithm

The proposed master-slave algorithm is structured as follows:
Algorithm 1.

```
1. Initialize 𝒯(0) ← {τ_1,...,τ_N} (e.g., τ_k are randomly or uniformly distributed), k =
   1, I(0) = {−1,...,−1}; Let ϵ > 0 a given tolerance;
2. Solve the master problem I(k) ← f_M(𝒯(k − 1));
3. If |F(𝒯(k − 1), I(k)) − F(𝒯(k − 1), I(k − 1))| ≤ ϵ set 𝒯(k) ← 𝒯(k − 1) and go to 7.
4. Solve the slave problem 𝒯(k) ← f_S(I(k));
5. k ← k + 1;
6. Go to 2.;
7. Set {τ_1,...,τ_N} ← 𝒯(k), {i_1,...,i_{N+1}} ← I(k);
8. End
```

Proposition 1 ([3]). *Algorithm 1 stops after a finite number of steps* N_{stop}.

Definition 1. *The optimal control problem (2) is said* switch-degenerate *if there exist a sequence* $\mathcal{T}$ *and* $I_1 \neq I_2$ *such that* $F(I_1, \mathcal{T}) = F(I_2, \mathcal{T})$.

Definition 2. *The optimal control problem (2) is said* time-degenerate *if there exist a sequence* I *and* $\mathcal{T}_1 \neq \mathcal{T}_2$ *such that* $F(I, \mathcal{T}_1) = F(I, \mathcal{T}_2)$.

The following Proposition 2 proves that Algorithm 1 cannot cycle over the same switching sequences, and $I(k) \neq I(j)$ for all $j \neq k$, $j, k \in \{1, \ldots, N_{\text{stop}} - 1\}$:

Proposition 2. *Let* $\epsilon = 0$ *and assume problem (2) is not switch-degenerate. Let step 3. be modified as follows*

```
3′.  If I(k) = I(k − 1) go to 7.;
```

Then Algorithm 1 stops after a finite number of steps N_{stop}.

Proof. See [3].

We remark that although Algorithm 1 converges to a solution $(I, \mathcal{T})$ after a finite number N_{stop} of steps, such a solution may not be the optimal one, as it may be a local minimum where both the master and the slave problems do not give any further improvement. Note that the global solution can be computed by enumeration by solving a slave problem for all possible s^N switching sequences I.

Algorithm 1 computes the optimal switching policy for a given initial state. On the other hand, for small enough perturbations of the initial state such that the optimal switching sequence does not change, the optimal time-switching policy is immediately available as a by-product of the slave algorithm, because of its state-feedback nature.

We finally remark that Algorithm 1 may be formulated by optimizing with respect to $\mathcal{T}$ first, for a given initialization of the switching sequence I. The advantage of switching between the master and slave procedures depends on the information available a priori about the optimal solution. For instance when the algorithm is solved repeatedly for subsequent values of the state vector (such as in a receding horizon scheme), it may be useful to use the previous switching sequence as a warm start and optimize with respect to $\mathcal{T}$ first.

5.1 Degeneracies

We remark the following about degeneracies:

step		τ_1	τ_2	τ_3	i_1	i_2	i_3	i_4	$F(I,T)$
1	M	0.290	0.498	0.672	1	3	3	3	1.44619
1	S	0.280	0.290	0.300	1	3	3	3	1.44615
2	M	0.280	0.290	0.300	1	2	3	3	1.44459
2	S	0.180	0.240	0.240	1	2	3	3	1.44026
3	M	0.180	0.240	0.240	1	2	3	3	1.44026

Table 1: Detailed results of Example 1.

1. Time-degeneracy: $i_k = i_{k-1}$ implies that the switching instant τ_k is undetermined (multiple solutions for $\mathcal{T}$)
2. Switch-degeneracy: $\tau_k = \tau_{k-1}$ implies that the switching index i_k is undetermined (multiple solutions for I)

We will show in the next section how degeneracy can be handled.

6 Numerical Examples

Example 1

Let us consider a second order linear system whose dynamics may be chosen within a finite set $\{A_1, A_2, A_3\}$. In particular, we assume $A_1 = \left[\begin{smallmatrix} -5.179 & -1.414 \\ 1 & 0 \end{smallmatrix}\right]$, $A_2 = \left[\begin{smallmatrix} -10.115 & -3.082 \\ 2 & 0 \end{smallmatrix}\right]$, $A_3 = \left[\begin{smallmatrix} -2.414 & -1.414 \\ 1 & 0 \end{smallmatrix}\right]$. We associate to each dynamics a weight matrix: $Q_1 = \text{diag}\{1,\ 1\}$, $Q_2 = \text{diag}\{8,\ 2\}$, $Q_3 = \left[\begin{smallmatrix} 1 & 0.5 \\ 0.5 & 1 \end{smallmatrix}\right]$. We also assume that only three switches are possible, thus $N = 3$ and the control variable $i(t)$ may only take values from the finite set of integers $\mathcal{S} = \{1, 2, 3\}$. Let the initial state vector be $x_0 = \left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right]$.

We apply the master–slave algorithm to determine the optimal index sequence. The initial timing sequence is $\mathcal{T}_0 = \{0.290, 0.498, 0.672\}$ (randomly generated).

The master–slave algorithm finds out that the optimizing index sequence is $I^* = \{1, 2, 3, 3\}$ and the optimal cost value is $V_3^* = 1.44026$. Note that in this case only two switches are required to get the optimal cost value.

Detailed intermediate results are reported in Table 1, where we may also observe that the procedure converges after only 3 steps. This also implies that the most burdensome part of the algorithm, i.e., the slave problem, was only solved twice.

The correctness of the solution has been validated through an exhaustive inspection of all admissible index sequences. More precisely, for each admissible index sequence we have computed the optimizing timing sequence and the corresponding cost value through the slave algorithm. In such a way we verified that $V_3^* = 1.44026$ is indeed a global minimum. Obviously, being only two the switches required to optimize the cost value, the minimum cost may also be obtained by using other index sequences. As an example, if we consider $I = \{3, 1, 2, 3\}$ and $\mathcal{T} = \{0, 0.180, 0.240\}$, this solution is optimal as well.

In Figure 1 we have reported the switching regions $\mathcal{C}_{j,3}$, $j = 1, 2, 3$, when the index sequence is the

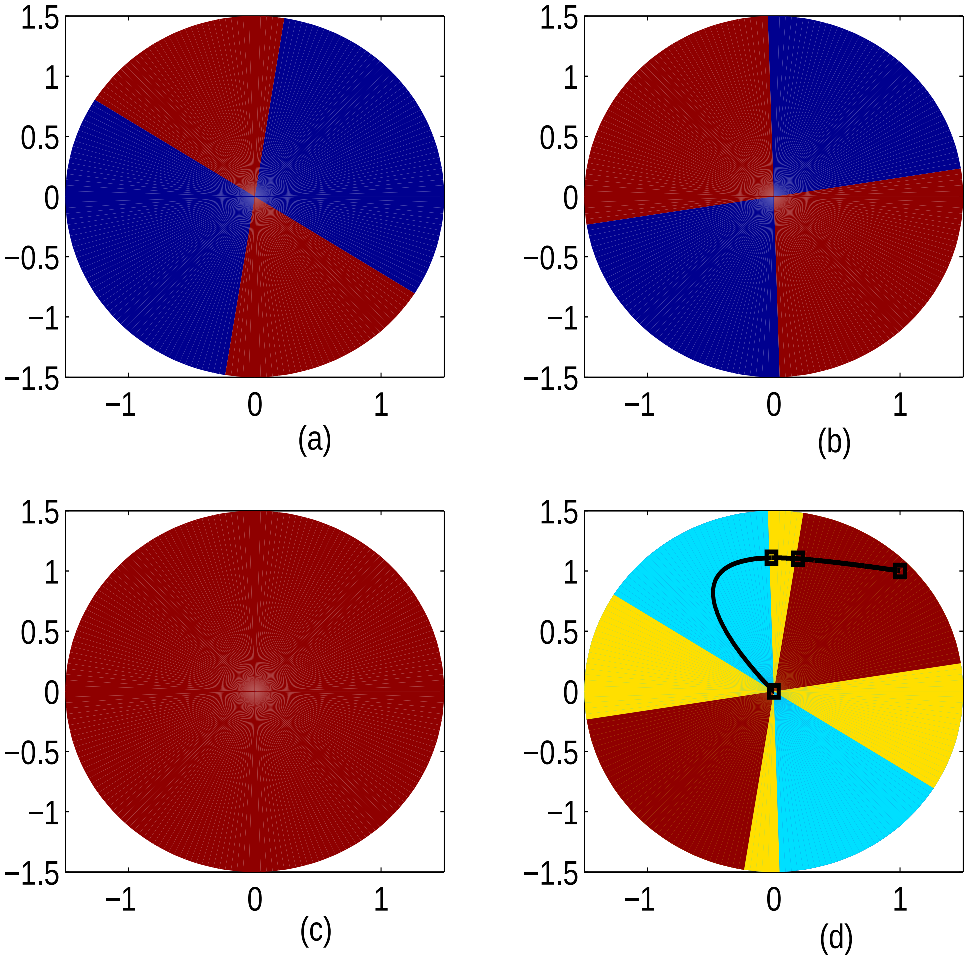

Figure 1: The switching regions for the system in Example 1 when the initial state is $x_0 = [1\ 1]'$ and the index sequence is the optimal one: (a) $\mathcal{C}_{1,3}$, (b) $\mathcal{C}_{2,3}$, (c) $\mathcal{C}_{3,3}$. (d) The system evolution from the initial state $x_0 = [1\ 1]'$.

step		τ_1	τ_2	τ_3	i_1	i_2	i_3	i_4	$F(I,T)$
1	M	0.001	0.005	0.010	2	2	2	3	5.63017
1	S	0.000	0.000	0.009	2	2	2	3	5.63017
2	M	0.000	0.000	0.009	1	1	2	3	5.63017
2	S	0.000	0.089	0.143	1	1	2	3	1.42998
3	M	0.000	0.089	0.143	1	1	2	3	1.42998

Table 2: Detailed results of Example 2 when the master-slave algorithm is applied in its original form.

optimal one. The blue (darker) region represents the set of states where the system still evolves with the same dynamics, while the red (lighter) region represents the set of states where the system switches to the next dynamics. Clearly, in $\mathcal{C}_{3,3}$ we have no partitioning because it corresponds to a non–effective switch, being $i(3) = i(4) = 3$.

Finally, in the bottom right of Figure 1 we show the system evolution from the chosen initial state $x_0 = [1\ 1]'$.

On the basis of several random tests we performed, we observed that the convergence of the algorithm to a global minimum is heavily influenced by two factors. Firstly, the initial switching times sequence should be such that $\tau_k > \tau_{k-1}$: in fact, if $\tau_k = \tau_{k+1}$ for some k, only a suboptimal solution — that corresponds to a minor number of switches — is usually computed. Secondly, the first switching time should not be greater than two or three times the maximum time constant associated to each dynamics: if this is not the case, only degenerate solutions with no switch are usually found.

Example 2

We present here an heuristics that in many cases improves the performance of the algorithm. Let us consider a second order linear system whose dynamics may be chosen within a finite set $\{A_1, A_2, A_3\}$. In particular, we assume: $A_1 = \left[\begin{smallmatrix} 1 & -10 \\ 100 & 1 \end{smallmatrix}\right]$, $A_2 = \left[\begin{smallmatrix} 1 & -100 \\ 10 & 1 \end{smallmatrix}\right]$, $A_3 = \left[\begin{smallmatrix} -0.1 & 0 \\ 0 & -0.1 \end{smallmatrix}\right]$ and $f_1 = f_2 = f_3 = 0$. Note that A_1 and A_2 are unstable matrices while A_3 is strictly Hurwitz, thus Assumption 1 is verified. We associate the same weighting matrix to each dynamics: $Q_1 = Q_2 = Q_3 = \left[\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right]$. We also assume that $N = 3$ and the control variable $i(t)$ may only take values from the finite set of integers $\mathcal{S} = \{1, 2, 3\}$. The initial state vector is $x_0 = \left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right]$.

We take as initial timing sequence $\mathcal{T}_0 = \{0.001, 0.005, 0.010\}$ and apply the master-slave algorithm to determine the optimal index sequence. The provided solution is $I = \{1, 1, 2, 3\}$ and the corresponding performance index is $V_3 = 1.42998$. Detailed results are reported in Table 2. Nevertheless, this solution is not optimal and this may be easily verified through an exhaustive inspection of all admissible switching sequences.

A careful examination of the solution suggests the presence of time-degeneracy, being $I = \{1, 1, 2, 3\}$ a switching sequence that corresponds to only two switches. Thus, when it is used by the slave algorithm, it may only compute a suboptimal solution.

A simple heuristic solution to this problem — that is effective in this case, as well as in many other numerical examples we have examined — consists of modifying the switching sequence computed via the master algorithm that corresponds to a number of switches that is less that N before running the slave

step		τ_1	τ_2	τ_3	i_1	i_2	i_3	i_4	$F(I,T)$
1	M	0.001	0.005	0.010	2	2	2	3	5.63017
1	S	0.000	0.089	0.143	3	1	2	3	1.42998
2	M	0.000	0.089	0.143	1	1	2	3	1.42998
2	S	0.009	0.062	0.116	2	1	2	3	0.12569
3	M	0.009	0.062	0.116	2	1	2	3	0.12569

Table 3: Detailed results of Example 2 when the master-slave algorithm is applied in its modified form.

algorithm. In particular, we suggest to arbitrarily change the index sequence so that the original sequence is still contained in the new one but two consecutive indices should never be the same.

Using such an heuristics, the results of the master–slave algorithm we obtain the results reported in Table 3. In particular, we observe that at the first step of the whole procedure, the slave algorithm does not examine the switching sequence firstly computed by the master algorithm, but computes the optimal timing sequence corresponding to a new sequence $I = \{3, 1, 2, 3\}$, that has been randomly generated by arbitrarily modifying the first index so as to avoid time-degeneracy. At this step, the value of the performance index decreases but the optimum is not computed yet. The same reasoning is repeated at the third step and in this case the optimal value of the cost is found and the procedure stops. The results of an exhaustive search show that the computed solution is optimal thus revealing the effectiveness of the modified procedure.

Although this heuristics is not always effective, it often improves the performance of the algorithm while it may never make it worse. Its only drawback is that, to avoid cycling, it is necessary to add a stopping condition that detects loops.

Finally, in Figure 2 we have reported the switching regions $\mathcal{C}_{j,3}$, $j = 1, 2, 3$, when the switching sequence is the optimal one. The color notation is the same as in the previous example. In the bottom right of Figure 2 we showed the system evolution from the chosen initial state $x_0 = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$.

7 Conclusions and Possible Extensions

In this paper we have proposed a master-slave algorithm for solving infinite-horizon linear quadratic optimal control problems for autonomous continuous-time switched affine systems, where both the switching instants and the sequence of operating modes must be determined.

An easy way of extending the approach described in this paper to non-autonomous switched linear systems $\dot{x}(t) = A_i x(t) + B_i u(t)$ is to set $u(t) = K_i x(t)$, where K_i is the LQR gain. Clearly, this would provide only a suboptimal solution to the original LQR problem for the switched linear system. The idea can be also extended similarly to non-autonomous switched affine systems.

The algorithm can be also easily extended to the case where the switching is not arbitrary, but is driven by a finite-state machine excited by exogenous discrete inputs. In this case, the reachable set of the automaton due to the discrete inputs and initial discrete state can be easily embedded in the MIQP master

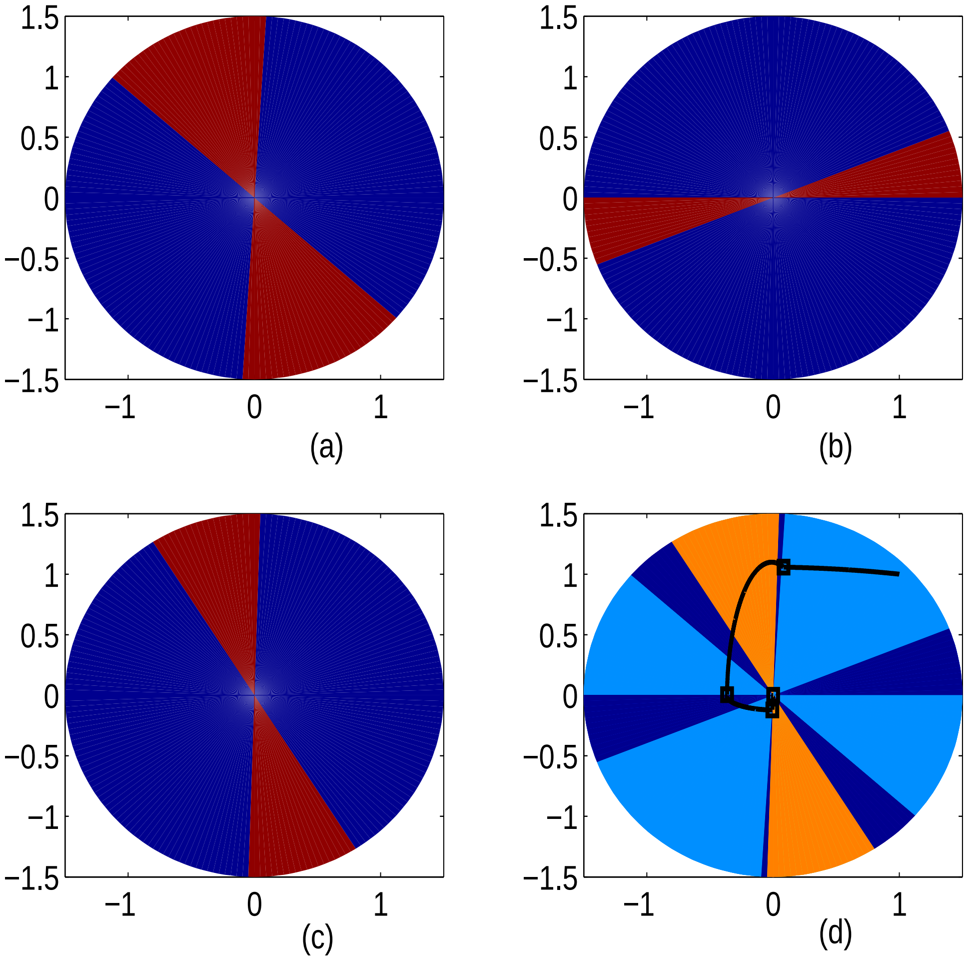

Figure 2: The switching regions for the system in Example 2 when the initial state is $x_0 = [1\ 1]'$ and the index sequence is the optimal one: (a) $\mathcal{C}_{1,3}$, (b) $\mathcal{C}_{2,3}$, (c) $\mathcal{C}_{3,3}$. (d) The system evolution for $x_0 = [1\ 1]'$.

problem to restrict the set of possible switching sequences.

References

[1] A. Bemporad, F. Borrelli, and M. Morari. Piecewise linear optimal controllers for hybrid systems. In *Proc. American Contr. Conf.*, pages 1190–1194, Chicago, IL, June 2000.

[2] A. Bemporad, L. Giovanardi, and F.D. Torrisi. Performance driven reachability analysis for optimal scheduling and control of hybrid systems. In *Proc. 39th IEEE Conf. on Decision and Control*, pages 969–974, Sydney, Australia, December 2000.

[3] A. Bemporad, A. Giua, and C. Seatzu. An iterative algorithm for the optimal control of continuous–time switched linear systems. In *6th Int. Work. on Discrete Event Systems (WODES)*, Zaragoza, Spain, October 2002.

[4] A. Bemporad, A. Giua, and C. Seatzu. Synthesis of state-feedback optimal controllers for switched linear systems. In *Proc. 41th IEEE Conf. on Decision and Control*, Las Vegas, USA, December 2002.

[5] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, March 1999.

[6] M.S. Branicky, V.S. Borkar, and S.K. Mitter. A unified framework for hybrid control: model and optimal control theory. *IEEE Trans. Automatic Control*, 43(1):31–45, 1998.

[7] M.S. Branicky and S.K. Mitter. Algorithms for optimal hybrid control. In *Proc. 34th IEEE Conf. on Decision and Control*, New Orleans, USA, December 1995.

[8] P.J. Antsaklis (ed.). Special issue on hybrid systems: Theory and applications. *Proceedings of the IEEE*, 88(7), July 2000.

[9] A. Giua, C. Seatzu, and C. Van Der Mee. Optimal control of autonomous linear systems switched with a pre–assigned finite sequence. In *Proc. 2001 IEEE Int. Symp. on Intelligent Control*, pages 144–149, Mexico City, Mexico, 2001.

[10] A. Giua, C. Seatzu, and C. Van Der Mee. Optimal control of switched autonomous linear systems. In *Proc. 40th IEEE Conf. on Decision and Control*, pages 2472–2477, Orlando, Florida USA, 2001.

[11] K. Gokbayrak and C.G. Cassandras. A hierarchical decomposition method for optimal control of hybrid systems. In *Proc. 38th IEEE Conf. on Decision and Control*, pages 1816–1821, Phoenix, AZ, December 1999.

[12] S. Hedlund and A. Rantzer. Optimal control of hybrid systems. In *Proc. 38th IEEE Conf. on Decision and Control*, pages 3972–3976, Phoenix, AZ, December 1999.

[13] B. Lincoln and A. Rantzer. Optimizing linear system switching. In *Proc. 40th IEEE Conf. on Decision and Control*, pages 2063–2068, 2001.

[14] C.C. Pantelides, M.P. Avraam, and N. Shah. Optimization of hybrid dynamic processes. In *Proc. American Contr. Conf.*, 2000. Available upon request from the authors.

[15] B. Piccoli. Necessary conditions for hybrid optimization. In *Proc. 38th IEEE Conf. on Decision and Control*, Phoenix, Arizona USA, December 1999.

[16] A. Rantzer and M. Johansson. Piecewise linear quadratic optimal control. *IEEE Trans. Automatic Control*, 45(4):629–637, April 2000.

[17] N. V. Sahinidis. BARON — Branch And Reduce Optimization Navigator. Technical report, University of Illinois at Urbana-Champaign, Dept. of Chemical Engineering, Urbana, IL, USA, 2000.

[18] H.P. Williams. *Model Building in Mathematical Programming*. John Wiley & Sons, Third Edition, 1993.

[19] X. Xu and P.J. Antsaklis. An approach to switched systems optimal control based on parameterization of the switching instants. In *Proc. IFAC World Congress*, Barcelona, Spain, 2002.