OPTIMAL STATE-FEEDBACK QUADRATIC REGULATION OF LINEAR HYBRID AUTOMATA †

Alberto Bemporad [*] **Daniele Corona** [**]
Alessandro Giua [**] **Carla Seatzu** [**]

[*] *Dip. Ing. dell'Informazione, U. di Siena, Italy*
[**] *Dip. Ing. Elettrica ed Elettronica, U. di Cagliari, Italy*

Abstract: For linear hybrid automata, namely switched linear autonomous systems whose mode of operation is determined by a controlled automaton, in this paper we face the problem of optimal control, where the objective is to minimize a quadratic performance index over an infinite time horizon. The quantities to be optimized are the sequence of switching times and the sequence of modes (or "locations"), under the following constraints: the sequence of modes has a finite length; the discrete dynamics of the automaton restricts the possible switches from a given location to the next location, with a cost associated to each switch; the time interval between two consecutive switching times is greater than a fixed quantity. We show how a state-feedback solution can be computed off-line through a numerical procedure.

Keywords: hybrid systems, switched systems, hybrid automata, optimal control.

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 (Antsaklis, 2000). 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 (this is the class considered in this paper) most of the literature is focused on the study of necessary conditions for a trajectory to be optimal (Piccoli, 1999; Sussmann, 1999), and on the computation of optimal/suboptimal solutions by means of dynamic programming or the maximum principle (Branicky *et al.*, 1998; Gokbayrak and Cassandras, 1999; Hedlund and Rantzer, 1999; Riedinger *et al.*, 1999; Xu and Antsaklis, 2002). Optimal control of discrete-time hybrid systems is tackled in (Bemporad *et al.*, 2002*b*).

In the case of switched linear systems composed by stable autonomous dynamics, by assuming that the switching sequence is pre-assigned (thus the only decision variables to be optimized are the switching instants), in (Giua *et al.*, 2001*a*) we proved that the control law is a state-feedback and there exists a numerically viable procedure to compute the switching tables $\mathcal{C}_{k,N}$ showing the points of the state space where the k-th switch of a sequence of length N should occur. In (Bemporad *et al.*, 2002*a*) we generalized this optimization problem by taking both the switching instants and the switching sequence as decision variables. The approach we proposed in (Bemporad *et al.*, 2002*a*) is still based on the construction of "switching tables". Using a simple procedure inspired by dynamic programming, we have shown how it is possible to avoid the exponential growth of the computational cost as the length of the switching sequence is increased.

In this paper we build on the results presented in (Bemporad *et al.*, 2002*a*) and extend the state-feedback control technique based on the construction of "switching tables" to also deal with constraints of practical relevance.

†Published in the *Proc. 2003 IFAC Conf. on Analysis and Design of Hybrid Systems* (St Malo, France), pp. 407–412, June 2003.

Constraint 1. The switching sequence is subject to logical constraints of the type: if $i_k = i$ then $i_{k+1} \in \text{succ}(i)$, where i_k is the index denoting the active dynamics at step k. This means that from dynamics i not all other dynamics can be reached with a simple switch, but only those whose index belongs to a given set, the set of successors of i, namely $\text{succ}(i)$. This may be described by an automaton where to each state is associated a dynamics, and to each transition a switch.

Constraint 2. Once entered in a location i we cannot leave it before a time $\delta_{\min}(i)$ has elapsed. This is a common constraint in many real applications: $\delta_{\min}$ may be the time necessary to control an actuator, or it may be the scan time of a PLC that triggers the switches.

Note that if the automaton describing the allowed mode switches is strongly connected, then from each state it may be possible to reach all other states with a sequence of one or more transitions. Without constraint 2 more than one transition may be executed in zero time, thus practically making constraint 1 meaningless.

The main advantages of the proposed procedure may be briefly summarized as follows:

— it is guaranteed to find the optimal solution under the given constraints;

— it has a computational cost of the order $\mathcal{O}(r^{n-1}Ns^2)$, where n is the dimension of the state space, r is the number of samples in each direction, N is the length of the switching sequence and s is the number of possible operating modes;

— it provides a "global" closed-loop solution, i.e., the tables may be used to determine the optimal state feedback law for all initial states.

Finally, to show the practical relevance of the presented framework, we discuss the example of a physical system: a servomechanism with gear-box.

2. THE HYBRID AUTOMATA MODEL

A hybrid automaton (HA) consists of a classic automaton extended with a continuous state $x \in \mathbb{R}^n$ that may continuously evolve in time with arbitrary dynamics or have discontinuous jumps at the occurrence of a discrete event (Nicollin *et al.*, 1993). In this paper we focus our attention on a particular class of HA, that we call *switched linear systems*. We consider a structure $H = (L, act, E, M)$ defined as follows.

— L is a finite set of locations.

— $act : L \to \mathit{Diff\_Eq}$ is a function that associates to each location $l_i \in L$ a linear differential equation of the form $\dot{x} = act_i(x) = A_i x$.

— $E \subset L \times L$ is the set of edges. An edge $e = (l_i, l_j)$ is an edge from location l_i to l_j, $i \neq j$.

— $M : E \to \mathbb{R}^{n\times n}$ associates to each edge $e \in E$ a constant matrix in $\mathbb{R}^{n\times n}$. When the discrete state switches from l_i to l_j at time τ, the continuous state x is set to $x(\tau^+) = M_{i,j}x(\tau^-)$.

The state of the HA is the pair (l, x) where $l \in L$ is the discrete location and $x \in \mathbb{R}^n$ is the continuous state. The hybrid automaton starts from some initial state (l_{i_0}, x_0). The trajectory evolves with the location remaining constant and the continuous state x evolving according to the *act* function at that location. When at time τ a switch is made to location l_{i_1} the continuous state is initialized to a new value $x(\tau^+) = M_{i_0,i_1}x(\tau^-)$. The new state is the pair $(l_{i_1}, x(\tau^+))$. The continuous state now moves with the new differential equation.

The classic definition of HA (Nicollin *et al.*, 1993) is more general than the one considered here because: an invariant set may be associated to each location; the activity set may be a differential inclusion rather than a linear differential equation; guards are associated to transitions; the jump relation may be arbitrary and not necessarily defined by a matrix M.

3. OPTIMAL CONTROL PROBLEM

In this paper we deal with the problem of designing an optimal control policy for a hybrid automaton $H = (L, act, E, M)$ as defined in the previous section. Let $s = |L|$ be the number of discrete locations and $\mathcal{S} \triangleq \{1, 2, \cdots, s\}$ be a finite set of integers, each one associated with a discrete location. The index i identifies the location l_i and consequently the linear dynamics A_i. We assume that a positive semi-definite matrix Q_i is associated to each discrete location $l_i \in L$ and a cost $H_{i,j}$ is associated to a switch from l_i to l_j.

Let us define the set $\text{succ}(i) = \{j \in \mathcal{S} \,:\, (l_i, l_j) \in E\}$ which denotes the indices associated to the locations reachable from l_i, and $\delta_{\min}(i)$ which is the minimum permanence time in l_i.

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

$$
\begin{aligned}
V_N^* \triangleq \min_{I,\mathcal{T}}\ & \Big\{F(I,\mathcal{T}) \\
& \triangleq \int_0^{\infty} x'(t)Q_{i(t)}x(t)dt + \sum_{k=1}^{N} h_k(\tau_k)\Big\} \\
\text{s.t. } & \dot{x}(t) = A_{i(t)}x(t),\ \ x(0) = x_0 \\
& i(t) = i_k \text{ for } \tau_k \le t < \tau_{k+1}, \qquad k = 0,\ldots,N \\
& \tau_0 = 0,\ \ \tau_{N+1} = +\infty \\
& \tau_{k+1} \ge \tau_k + \delta_{\min}(i_k), \qquad k = 0,\ldots,N \\
& i_k \in \text{succ}(i_{k-1}), \qquad k = 1,\ldots,N \\
& x(\tau_k^+) = M_{i_{k-1},i_k}x(\tau_k^-), \qquad k = 1,\ldots,N \\
& h_k(\tau_k) = H_{i_{k-1},i_k} \text{ if } \tau_k < +\infty, \\
& h_k(\tau_k) = 0 \text{ if } \tau_k = +\infty, \qquad k = 1,\ldots,N
\end{aligned}
\tag{1}
$$

The control variables are $\mathcal{T} \triangleq \{\tau_1, \ldots, \tau_N\}$ and $I \triangleq \{i_0, \ldots, i_N\}$, where $\mathcal{T}$ is the set of switching times and I is the sequence of indices associated with discrete locations. We assume that the maximum number N of allowed switches is fixed a priori.

The cost $F(I, \mathcal{T})$ consists of two components: a quadratic cost that depends on the time evolution (the integral) and a cost that depends on the switches (the sum). Note that $\tau_k < +\infty$ means that the $k-$th switch occurs after a finite amount of time, while $\tau_k = +\infty$ means that the $k-$th switch does not occur: in the latter case $h_k(\tau_k) = 0$ thus its cost is not considered.

We denote by $i^*(t)$, $t \in [0, +\infty)$, $i^*(t) = i^*_k$ for $\tau^*_k \le t < \tau^*_{k+1}$ the switching trajectory solving (1), 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.

Let us define $\delta_k = \tau_{k+1} - \tau_k$. The optimal control problem (1) may also be rewritten as:

$$\begin{array}{l} \min\limits_{I,\mathcal{T}} \left\{ \sum\limits_{k=0}^{N} x'_k \bar{Q}_{i_k}(\delta_k) x_k + \sum\limits_{k=1}^{N} h_k(\tau_k) \right\} \\ \text{s.t. } x_{k+1} = M_{i_k,i_{k+1}} \bar{A}_{i_k}(\delta_k) x_k,\ k = 0,\ldots,N-1 \\ \qquad x_0 = x(0) \\ \qquad i_0 \in \mathcal{S} \\ \qquad i_k \in \text{succ}(i_{k-1}),\ k = 1,\cdots,N \\ \qquad \delta_k \ge \delta_{\min}(i_k),\ k = 0,\cdots,N \end{array} \tag{2}$$

where

$$\bar{A}_i(\delta_k) \triangleq e^{A_i \delta_k}, \tag{3}$$

$$\begin{aligned} \bar{Q}_i(\delta_k) &\triangleq \left(\int_{\tau_k}^{\tau_{k+1}} e^{A'_i(t-\tau_k)} Q_i e^{A_i(t-\tau_k)} dt \right) \\ &= \left(\int_0^{\delta_k} e^{A'_i t} Q_i e^{A_i t} dt \right), \end{aligned} \tag{4}$$

thus $\bar{Q}_i(\delta_k)$ can be obtained by simple integration and linear algebra. When A_i is asymptotically stable it is possible to write $\bar{Q}_i(\delta_k) = Z_i - \bar{A}'_i(\delta_k) Z_i \bar{A}_i(\delta_k)$, where Z_i is the unique solution of the Lyapunov equation $A'_i Z_i + Z_i A_i = -Q_i$ (Giua *et al.*, 2001*b*).

4. STATE-FEEDBACK CONTROL LAW

In this section we show that the optimal control law for the optimization problem described in the previous section takes the form of a state-feedback, i.e., it is only necessary to look at the current system state x in order to determine if a switch from location $l_{i_{k-1}}$ to l_{i_k},or equivalently from linear dynamics $A_{i_{k-1}}$ to A_{i_k}, should occur.

In particular, for a given mode $i \in \mathcal{S}$ and for a given switch $k \in 1,\ldots,N$ it is possible to construct a table $\mathcal{C}^i_{k,N}$ that partitions the state space $\mathbb{R}^n$ into s_i regions $\mathcal{R}_j$'s, where $s_i = |\text{succ}(i)| + 1$. Whenever $i_{k-1} = i$ we use table $\mathcal{C}^i_{k,N}$ to determine if a switch should occur: as soon as the state reaches a point in the region $\mathcal{R}_j$ for a certain $j \in \text{succ}(i)$ we will switch to mode $i_k = j$; on the contrary, no switch will occur while the system's state belongs to $\mathcal{R}_i$.

This is an important result because it is well known that a state-feedback control law has many advantages over an open-loop control law, including that the computation of the control law can be done off line as opposed to being performed on line.

To prove this result, we show constructively how the tables $\mathcal{C}^i_{k,N}$ can be computed using a dynamic programming argument. We first show how the tables $\mathcal{C}^i_{N,N}$ ($i \in \mathcal{S}$) for the last switch can be determined. Then, we show by induction how the tables $\mathcal{C}^i_{k,N}$ can be computed once the tables $\mathcal{C}^i_{k+1,N}$ are known.

4.1 Computation of the Tables for the Last Switch

Let us assume that $i_{N-1} = i$, i.e., after $N-1$ switches the current system dynamics is that corresponding to matrix A_i, and the current state vector is y with $||y|| = 1$. We show how to compute the table $\mathcal{C}^i_{N,N}$.

The optimal remaining cost starting from y will consist of two terms: a term due to the time-driven evolution, plus (if the $N-$th switch occurs and $i_N = j$) the switching cost $H_{i,j}$.

— Let us first consider the case in which no switch occurs. The remaining cost starting from y is only due to the time-driven evolution and is

$$T^*_i(y, i) = y' \bar{Q}_i(+\infty) y. \tag{5}$$

— If the system evolves with dynamics A_i for a time ϱ and then a switch to A_j ($j \in \text{succ}(i)$) occurs, the remaining cost starting from y only due to the time-driven evolution (disregarding the switching cost) is

$$\begin{aligned} T_i(y, \varrho, j) = y' \bar{Q}_i(\varrho) y + \\ + y' \bar{A}'_i(\varrho) M'_{i,j} \bar{Q}_j(+\infty) M_{i,j} \bar{A}_i(\varrho) y. \end{aligned} \tag{6}$$

Let us denote $\varrho_i = +\infty$, while for $j \in \text{succ}(i)$ we denote

$$\varrho_{\bar{j}} = \arg \min_{\varrho \ge 0, \bar{j} \in succ_c(i_{\bar{h}})} T_i(y, \varrho, j), \tag{7}$$

the value of ϱ that minimizes (6). The corresponding minimum is

$$T^*_i(y, j) = T_i(y, \varrho_j, j). \tag{8}$$

Note that T_i assumes that we can switch after $\varrho = 0$ time instants, i.e., the constraint about the minimum soujourn time in l_i has already been fulfilled.

Let us now consider any other vector x such that $x = \lambda y$, with $\lambda \in \mathbb{R}$. We can compute for this new vector the equivalent of (5) and (6), i.e.,

$$T^*_i(x, i) = x' \bar{Q}_i(+\infty) x = \lambda^2 T^*_i(y, i) \tag{9}$$

and for $j \in \text{succ}(i)$

$$T_i(x, \varrho, j) = \lambda^2 T_i(y, \varrho, j), \tag{10}$$

Equation (10) is minimized by the same value $\varrho = \varrho_j$ that minimizes (6) and its minimal value is

$$T^*_i(x, j) = \lambda^2 T^*_i(y, j). \tag{11}$$

We discuss separately two cases.

— If all switching costs are null, the optimal remaining cost starting from x and allowing at most one switch is

$$F^*_{i,N}(x) = \lambda^2 \min_{j \in \{\text{succ}(i), i\}} \{T^*_i(y, j)\}, \tag{12}$$

while the value of j that minimizes the previous equation is denoted $j^*(y)$. Thus the optimal switch from mode i to mode j should occur after a delay

$$\delta^*_{i,N}(x) = \delta^*_{i,N}(y) = \varrho_{j^*(y)} \qquad (13)$$

that for $x = \lambda y$ is a function of y but not of λ.

We can say that a vector $x = \lambda y$ belongs to $\mathcal{R}_j$ ($j \in \mathrm{succ}(i)$) if and only if $j = j^*(y)$ and $\delta^*_{i,N}(y) = 0$, because in this case the optimal remaining cost can be obtained switching as soon as we reach x with no delay. Finally, $\mathcal{R}_i = \mathbb{R}^n \setminus \cup_{j\in\mathrm{succ}(i)}\mathcal{R}_j$. Since the value of $\delta^*_{i,N}(\lambda y)$ in this case does not depend on λ, it immediately follows that these regions are homogeneous [1], i.e., if $x \in \mathcal{R}_j$ then $\lambda x \in \mathcal{R}_j$, for all real numbers λ. This property may be exploited in the construction of the table since it is only necessary to compute $F^*_{i,N}(y)$ and $\delta^*_{i,N}(y)$ for all vectors y that belong to the unitary semi-sphere.

— Assume that not all $H_{i,j}$ (this is the cost of switching from mode i to mode j) are null and let us define $H_{i,i} = 0$. Taking into account the switching cost, the optimal remaining cost starting from x and allowing at most one switch is

$$F^*_{i,N}(x) = \min_{j\in\{\mathrm{succ}(i),i\}} \{T^*_i(x,j) + H_{i,j}\}, \qquad (14)$$

while the value of j that minimizes the previous equation is denoted $j^*(x)$. Thus the optimal switch should occur after a delay

$$\delta^*_{i,N}(x) = \varrho_{j^*(x)}. \qquad (15)$$

We can say that a vector $x = \lambda y$ belongs to $\mathcal{R}_j$ ($j \in \mathrm{succ}(i)$) if and only if $j = j^*(x)$ and $\delta^*_{i,N}(x) = 0$. Finally, $\mathcal{R}_i = \mathbb{R}^n \setminus \cup_{j\in\mathrm{succ}(i)}\mathcal{R}_j$. In this case it is not sufficient to compute $F^*_{i,N}(y)$ and $\delta^*_{i,N}(y)$ for all vectors y that belong to the unitary semi-sphere but we also have to take into account the norm λ of a vector $x = \lambda y$ (at least for small values of λ: for λ large enough the effect of the switching cost becomes negligible).

Note that in order to compute the switching regions $\mathcal{R}_j$ and to determine the optimal remaining cost $F^*_{i,N}(x)$, we only need to compute the values $\varrho_i(j)$ with $|\mathrm{succ}(i)|$ one-parameter optimizations (see equations (6) and (7)) for all y on the unitary semi-sphere. The corresponding values of $T^*_i(y,i)$ and $T^*_i(y,j)$ can be obtained applying equations (5) and (8), while to determine if a vector $x = \lambda y$ belongs to $\mathcal{R}_j$ and to compute the corresponding optimal remaining cost we only need to apply equations (14) and (15).

4.2 Computation of the Tables for the Intermediate Switches

We now generalize the previous approach to determine the tables $\mathcal{C}^i_{k,N}$, for $k = 1,\ldots,N-1$.

Assume that: (a) we have already computed the tables $\mathcal{C}^j_{k+1,N}$ for all $j \in \mathcal{S}$; (b) for each vector x and each mode $j \in \mathcal{S}$ we know the optimal cost $F^*_{j,k+1}(x)$ for the remaining time-driven evolution that starts from x with dynamics A_j and allows $N-k$ more switches.

With the same argument of the previous subsection we can write that

$$F^*_{i,k}(x) = \min_{j\in\{\mathrm{succ}(i),i\}} \{T^*_i(x,j) + H_{i,j}\}, \qquad (16)$$

where $T^*_i(x,i) = x'\bar{Q}_i(+\infty)x$, while for $j \in \mathrm{succ}(i)$

$$T^*_i(x,j) = \min_{\varrho\geq 0} \; \{x'\bar{Q}_i(\varrho)x + x'_j(\varrho)\bar{Q}_j(\delta_{min}(j))x_j(\varrho) + F^*_{j,k+1}(\bar{A}_j(\delta_{min}(j))x_j(\varrho))\;\},$$

where $x_j(\varrho) = M_{i,j}\bar{A}_i(\varrho)x$. Each member of the sum that defines $T^*_i(x,j)$ has the following physical meaning: the first one is the cost of the evolution in the current location l_i for a time ϱ, the second one is the cost of the minimum permanence $\delta_{min}(j)$ in the successive location l_j, the third one is the optimal remaining cost from point $\bar{A}_j(\delta_{min}(j))x_j(\varrho)$ to infinity and its value has been determined at the previous step of the algorithm. We are thus able to compute the table $\mathcal{C}^i_{k,N}$, as we did before.

4.3 Computation of the Table for the Initial Mode

An additional degree of freedom that one may want to exploit is that of choosing the initial location, i.e., we assume that only the initial continuous state $x(0) = x_0$ is given.

To decide the optimal initial location l_{i_0} we may use the knowledge of the cost $F^*_{i,1}(\cdot)$ that is evaluated during the construction of the table $\mathcal{C}^i_{1,N}$. We define the cost

$$F^*_{i,0}(x) = x'\bar{Q}_i(\delta_{min}(i))x + F^*_{i,1}(\bar{A}_i(\delta_{min}(i))x),$$

which is the optimal global cost over the infinite time horizon starting from point x and constrained to location l_i for at least a $\delta_{min}(i)$ amount of time. Thus we construct a new table $\mathcal{C}_{0,N}$ showing a partition of the state space $\mathbb{R}^n$ into s regions $\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_s$.

Each region in this table is defined as follows:

$$\mathcal{R}_i = \{x \mid (\forall j \in \mathcal{S}) F^*_{i,0}(x) \leq F^*_{j,0}(x)\}$$

i.e., if the initial state belongs to region $\mathcal{R}_i$ we must choose $i_0 = i$ to minimize the total cost.

5. COMPUTATIONAL COMPLEXITY

We discuss here the computational complexity involved in the construction of the tables following the approach sketched in the previous section.

If the state space is $\mathbb{R}^n$ and we take r samples along each direction, then the computational complexity for constructing each table using the algorithm given by Giua *et al.* (2001*a*; 2001*b*) is $\mathcal{O}(r^{n-1})$ because these regions can be determined by solving a one-parameter optimization problem for each

[1] A term also used to define the special form of these regions is *conic*.

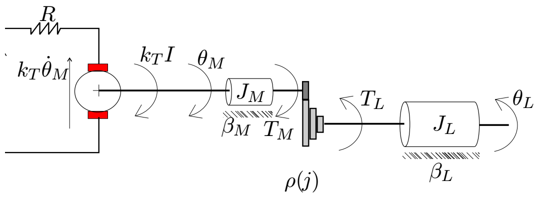

Fig. 1. Servomechanism model with controllable gear ratio.

vector x on the unitary semi-sphere. Thus the complexity of solving the optimal control problem for a pre-assigned sequence of length $N+1$ is $\mathcal{O}(Nr^{n-1})$, because for each switch a new table must be determined.

Using the algorithm given in the previous section, for each switch it is necessary to compute s tables, one for each discrete location. Furthermore the cost of computing the tables $\mathcal{C}^i_{k,N}$ is equal to $\mathcal{O}((s_i-1)r^{n-1})$. In fact each table contains s_i regions that can be determined solving s_i-1 one-parameter optimization problems for each vector x on the unitary semi-sphere. Thus the complexity of solving the optimal control problem (1) for a sequence of length N is $\mathcal{O}(Nr^{n-1}\sum_{i=1}^{s}(s_i-1)) \le \mathcal{O}(Nr^{n-1}s^2)$, because $s_i \le s$. Thus, even in the worst case the complexity is quadratic in the number of possible locations.

If we solve by brute force an optimal control problem of the form (1) by investigating all admissible switching sequences (they are $(s-1)^N$ in the worst case) the complexity becomes $\mathcal{O}(Nr^{n-1}s^{N+1})$.

6. A SERVOMECHANISM WITH GEAR-BOX

As an example we consider the following servomechanism system. It consists of a DC-motor, a gear-box with selectable gear ratios, and a mechanical load. The dynamics of the system is described by the relations $V = RI + k_T\dot{\theta}_M$, $J_M\ddot{\theta}_M = k_T I - \beta_M\dot{\theta}_M - T_M$, $\dot{\theta}_M = \rho(j)\dot{\theta}_L$, $T_L = \rho(j)T_M$, $J_L\ddot{\theta}_L = -\beta_L\dot{\theta}_L + T_L$ where V is the applied armature voltage, I the armature current, R the armature resistance, θ_M, θ_L the angular position of the motor and load shafts, respectively, T_M the torque developed by the motor, k_T the motor constant, J_M, J_L, β_M, β_L the equivalent moment of inertia and viscous friction coefficient of the motor and load, respectively, and $\rho(j)$ the gear ratio, $j \in \{1,2,3\}$. The above relations can be easily rewritten as the linear differential equation

$$\left[J_L + \rho^2(j)J_M\right]\ddot{\theta}_L + \left[\beta_L + \rho^2(j)\left(\frac{k_T^2}{R} + \beta_M\right)\right]\dot{\theta}_L = \rho(j)\frac{k_T}{R}V.$$

We assume that V can be generated by one of the following PD controllers $V = -k_1(h)\theta_L - k_2(h)\dot{\theta}_L$, $h \in \{1,2\}$, where $h=1$ corresponds to a smooth control action, while $h=2$ corresponds to an aggressive one. By setting $x \triangleq [\theta_L \ \ \dot{\theta}_L]'$, the overall model can be represented as the autonomous switched linear system $\dot{x} = A(h,j)x$

$$\dot{x} = A(h,j)x = \begin{bmatrix} 0 & 1 \\ a_{21}(h,j) & a_{22}(h,j) \end{bmatrix} x$$

where

$$a_{21}(h,j) = -\frac{\rho(j)(k_T/R)k_1(h)}{J_L+\rho^2(j)J_M};$$

$$a_{22}(h,j) = -\frac{\beta_L+\rho^2(j)\big((k_T^2/R)+\beta_M\big)+\rho(j)(k_T/R)k_2(h)}{J_L+\rho^2(j)J_M}.$$

Equivalently, we write $\dot{x} = A(l_i)x$, where $i \triangleq 1 + (h-1) + 2(j-1)$, $A(l_i) \triangleq A(h,j)$, $i \in \{1,\dots,6\}$. We assume that (i) the gear shift is sequential, i.e., only transitions $1 \leftrightarrow 2$, $2 \leftrightarrow 3$ are allowed; (ii) a gear can be shifted only when the smooth control is active, in order to avoid power losses.

The automaton showing all the allowed transitions is depicted in Fig. 2. The parameters of the system are reported in the table below.

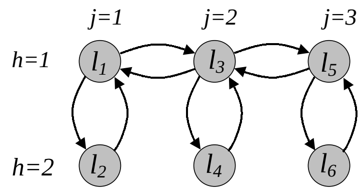

Fig. 2. The hybrid automaton that defines the mode switchings.

Symbol	Value (MKS)	Physical meaning
J_M	1	motor inertia
β_M	0.2	motor friction coefficient
R	50	resistance of armature
k_T	15	motor constant
J_L	50	nominal load inertia
β_L	10	load friction coefficient
ρ	1,2,3	gear ratios
k_1	3.2, 31.6	proportional action
k_2	3.5, 32.1	derivative action

6.1 *Numerical simulations*

We considered the following numerical values: the maximum number of switches is $N=5$; the state x is a continuous function (i.e. $M_{i,j}$ is the identity matrix for any l_i, l_j); no cost is associated to any switch (i.e. $H_{i,j}=0$ for any l_i, l_j); the minimum permanence time in every location is $\delta_{min} = 0.2\ s$; the initial state of the system is $x_0 = [-1.4 \ \ 1.5]'$, and the initial discrete location is l_1.

We assumed $Q_1 = Q_3 = Q_5 = \mathrm{Diag}\{1,\ 2\}$ and $Q_2 = Q_4 = Q_6 = \mathrm{Diag}\{3,\ 6\}$. We evaluate offline the $N \times |L|$ switching tables, each of them containing up to $|1 + succ(\cdot)|$ colors. A space discretization of $r = 101$ points along the unitary semisphere and a local minimum search over three time constants were considered sufficiently fine.

The state trajectory that minimizes the performance index is depicted in Fig. 3, where the circle indicates the initial state and the squares indicate the values of the state at the switching instants. We found out $\mathcal{T}^* = \{0.20, 0.40, 1.47, 4.0,\ 4.2\}$, $I^* = \{1,3,5,6,5,3\}$, and $J^* = 4.75$.

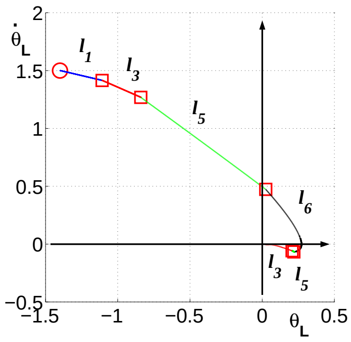

Fig. 3. The system evolution for $\theta_L(0) = -1.4$, $\dot{\theta}_L(0) = 1.5$, and initial location l_1.

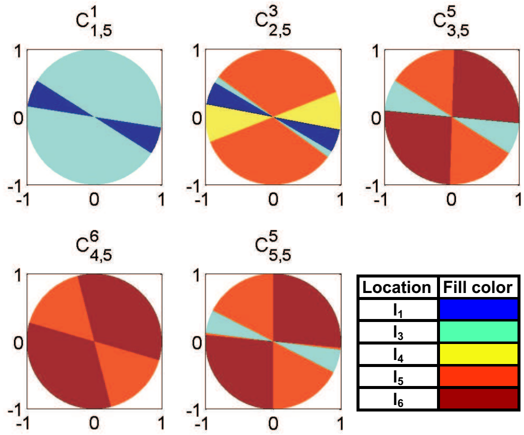

Fig. 4. Tables used by the controller to compute the state evolution in Fig. 3.

Fig. 4 shows, among the 30 tables constructed, only the 5 ones used by the controller during the evolution of the system. The system initially evolves for the minimum time in locations l_1. When this time has elapsed, the controller must keep checking the color in table $\mathcal{C}^1_{1,5}$ (see figure 4) corresponding to the current state x. According to this color the controller decides to remain in l_1 or to switch to an adjacent location. In this example an immediate switch to l_3 takes place, since the current state is in the cyan area. Now the controller will wait for the minimum time and then consider table $\mathcal{C}^3_{2,5}$. The same procedure is repeated until all available switches are performed.

7. CONCLUSIONS

We have considered a special class of autonomous linear switched systems where: a) the allowed mode switches are described by an automaton where to each state is associated a dynamics, and to each transition a switch; b) the interval between two consecutive switching instants is bounded from below. For this class we have shown that it is possible to extend the results presented in (Bemporad *et al.*, 2002*a*) based on the construction of "switching tables" to solve an optimal control problem with a state-feedback.

REFERENCES

Antsaklis, P.J. (Ed.) (2000). Special issue on Hybrid Systems. *Proceedings of the IEEE* **88** *(7)*.

Bemporad, A., A. Giua and C. Seatzu (2002*a*). Synthesis of state-feedback optimal controllers for continuous time switched linear systems. In: *Proc. 41th IEEE Conf. on Decision and Control*. Las Vegas, USA. pp. 3182–3187.

Bemporad, A., F. Borrelli and M. Morari (2002*b*). On the optimal control law for linear discrete time hybrid systems. In: *Hybrid Systems: Computation and Control*. LNCS 2289. Springer Verlag. pp. 105–119.

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

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

Giua, A., C. Seatzu and C. Van Der Mee (2001*b*). Optimal control of switched autonomous linear systems. In: *Proc. 40th IEEE Conf. on Decision and Control*. Orlando, USA. pp. 2472–2477.

Gokbayrak, K. and C.G. Cassandras (1999). A hierarchical decomposition method for optimal control of hybrid systems. In: *Proc. 38th IEEE Conf. on Decision and Control*. Phoenix, USA. pp. 1816–1821.

Hedlund, S. and A. Rantzer (1999). Optimal control of hybrid systems. In: *Proc. 38th IEEE Conf. on Decision and Control*. Phoenix, USA. pp. 3972–3976.

Nicollin, X., A. Olivero, J. Sifakis and S. Yovine (1993). An approach to the description and analysis of hybrid systems. In: *Hybrid Systems*. LNCS 736. Springer Verlag. pp. 149–178.

Piccoli, B. (1999). Necessary conditions for hybrid optimization. In: *Proc. 38th IEEE Conf. on Decision and Control*. Phoenix, USA. pp. 410–415.

Riedinger, P., F. Kratz, C. Iung and C. Zanne (1999). Linear quadratic optimization for hybrid systems. In: *Proc. 38th IEEE Conf. on Decision and Control*. Phoenix, USA. pp. 3059–3064.

Sussmann, H.J. (1999). A maximum principle for hybrid optimal control problems. In: *Proc. 38th IEEE Conf. on Decision and Control*. Phoenix, USA. pp. 425–430.

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