Optimal Feedback Switching Laws
for Autonomous Hybrid Automata*

D. Corona, A. Giua, C. Seatzu

Dip. Ingegneria Elettrica ed Elettronica, Università di Cagliari
Piazza d’Armi, 09123 Cagliari, Italy
daniele.corona,giua,seatzu@diee.unica.it

Abstract

We define a new class of hybrid systems called Autonomous Hybrid Automata that
can be seen as a generalization of the class of switched systems we have considered in
previous works. In this new model there are two types of edges: a controllable edge
represents a mode switch that can be triggered by the controller; an autonomous edge
represents a mode switch that is triggered by the continuous state of the system as it
reaches a given threshold. We show how to solve an infinite time horizon quadratic
optimization problem with a numerically viable procedure for such a class of Hybrid
Automata; the optimal control law is a state-feedback.

*Published as: D. Corona, A. Giua, C. Seatzu, "Optimal Feedback Switching Laws for Autonomous
1 Introduction

The optimal control of switched and hybrid systems has been widely investigated in the last years and many results can be found in the recent literature.

For continuous-time hybrid systems, most of the literature is focused on the study of necessary conditions for a trajectory to be optimal \([8, 19]\), and on the computation of optimal/suboptimal solutions by means of dynamic programming or the maximum principle \([5–7, 11, 13, 16, 17, 20]\). For determining the optimal feedback control law some of these techniques require the discretization of the state space in order to solve the corresponding Hamilton-Jacobi-Bellman equations. In \([10]\) the authors use a hierarchical decomposition approach to break down the overall problem into smaller ones. In so doing, discretization is not involved and the main computational complexity arises from a higher-level nonlinear programming problem.

In \([2,3,9,18]\) we considered continuous-time switched systems where each subsystem has a dynamics of the type \(\dot{x}(t) = A_i x(t)\), and considered a particular optimization problem, with an infinite horizon quadratic cost function and a fixed number \(N\) of allowed switches. We showed that in this setting the optimal control law is a state-feedback and there exists a numerically viable procedure to compute off-line the switching regions, i.e., the points of the state space where the \(k\)-th switch should occur (for \(k \in \{1, \ldots, N\}\)). The proposed approach was based on discretization. However, we showed that — if no cost is associated to a switch — the structure of these regions is homogenous and thus they can be computed by a discretization of the unitary semi-sphere (as opposed to a discretization of all the state space) and they can also be parametrized.

In this paper we show how to solve the same optimal control problem extending the class of considered plant models from switched systems to more general hybrid automata. While in a switched system all switches are assumed to be controllable (i.e., they can be triggered by the controller) in a hybrid automaton there may also exist autonomous switches that are internally triggered by the crossing of a given threshold. This type of autonomous switches have also been considered by Xu and Antsaklis \([21]\) in a recent work.

As in \([2,3,18]\) we assume that the decision variables to solve the optimization problem are the controlled switching instants \(\tau_1, \ldots, \tau_N\) — here \(\tau_k\) denotes the time instant in which the \(k\)-th controlled switch occurs — and the location indices \(i(\tau_1), \ldots, i(\tau_N)\) — here \(i(\tau_k)\) denotes the index of the location reached when the \(k\)-th controlled switch occurs.

In this paper we show how the considered optimal control problem can be solved with a state-feedback control law still based on the off-line computation of appropriate switching regions. To this aim we formally define the class of Autonomous Hybrid Automata (AHA) whose main feature is that it has a continuous and discrete autonomous behavior. In fact we consider that: a) there is no continuous control input; b) a subset of edges may fire autonomously, depending upon a set of constraints (guards) on the space state \(\mathbb{R}^n\).
Many real world situations motivated our study. As a trivial example consider a circuit containing a diode where the voltage threshold $x_1(t) < 0$ denotes the condition where the diode behaves as an open circuit. It also permits to model several cases where the continuous evolution of the system must be restricted to a safe or specification region.

The controller is discrete and may intervene in the system only if certain conditions are verified. Such conditions are formally described by the definition of an invariant set in $\mathbb{R}^n$ where the controller has complete decision freedom within a set of edges that can activate, in order to drive the system to the equilibrium point. Our purpose is to show how to construct an optimal quadratic feedback controller, taking also into account the possibility of autonomous switches of the system.

Moreover, we show in Section 5, that under appropriate structural hypotheses on the guards of the autonomous edges, the problem can be simplified because the switching regions can be computed by a discretization of the unitary semisphere. This particular case allows a reduction of the computational cost order from $O(r^n)$ to $O(r^{n-1})$ ($r$ is the number of samples in each direction). Obviously such advantage is significant for low values of $n$ (i.e., up to 4).

The paper is structured as follows. In Section 2 we formally define the model AHA. In Section 3 the optimization problem we consider is stated and some relevant properties of AHA are discussed. In Section 4 we show how the approach of [2, 3] can be extended to deal with the more general framework of AHA. In Section 5 we illustrate a particular class of the AHA. Finally a complete example is discussed in Section 6.

2 The considered model

In this section we first recall the general definition of Hybrid Automaton. Then we define a particular class of HA, named Autonomous Hybrid Automaton, on which we focus our attention in this paper.

2.1 Hybrid Automata

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 [1,14]. More precisely, a hybrid automaton is a structure $H = (L, act, inv, E, Jump)$ defined as follows [1,14].

- $L = \{l_1, \cdots, l_s\}$ is a finite set of locations.
- $act : L \rightarrow Inclusions$ is a function that associates to each location $l_i \in L$ a differential inclusion of the form $\dot{x} \in act_i(x)$ where $act_i(x)$ is a set-valued map. If $act_i(x)$ is a singleton then it is a differential equation.
• \( \text{inv} : L \to \text{Invariants} \) is a function that associates to each location \( l_i \in L \) an invariant \( \text{inv}_i \subseteq \mathbb{R}^n \). An invariant function is \( x \in \text{inv}_i \). The invariant function constraints the behavior of the automaton state during the continuous evolution within a given subset of \( \mathbb{R}^n \). In other words, if the evolution of the continuous state within location \( l_i \) would produce a violation of the invariant function, then the system cannot continue evolving within location \( l_i \).

• \( E \subset L \times \text{Guards} \times L \) is the set of edges. An edge \( e = (l, g_e, l') \in E \) is an edge from location \( l \) to \( l' \) and guard \( g_e \subseteq \mathbb{R}^n \). The edge is enabled when the current location is \( l \) and the current continuous state is \( x \in g_e \): it may fire reaching the new location \( l' \).

• A jump relation is \( j_e \subseteq \mathbb{R}^n \times \mathbb{R}^n \) associated to an edge \( e \). When the edge fires, \( x \) is set to \( x' \) provided \( (x, x') \in j_e \). When \( j_e \) is the identity relation, the continuous state does not change.

The state of the HA is the pair \((i, x)\) where the index \( i \) identifies the discrete location \( l_i \in L \) and \( x \in \mathbb{R}^n \) is the continuous state. The hybrid automaton starts from some initial state \((i_0, x_0)\). The trajectory evolves with the location remaining constant and the continuous state \( x \) evolving within the invariant function at that location, and its first derivative remains within the differential inclusion at that location. When the continuous state satisfies the guard of an edge from location \( l_i \) to \( l_k \) at time \( \tau \), a switch can be made to location \( l_k \). During the jump at time \( \tau \), the continuous state may get initialized from \( x(\tau^-) \) to a new value \( x(\tau) \). The new state is the pair \((i_1, x(\tau))\). The continuous state now moves within the new invariant function with the new differential inclusion, followed some time later by another switch, and so on.

We now define in detail the particular class of HA considered in this paper, that we call \textit{Autonomous Hybrid Automata} (AHA).

An AHA is a structure \( H = (L, \text{act}, \text{inv}, E, M) \) that satisfies the following assumptions.

(A1) The activity \( \text{act} : L \to \text{DiffEq} \) is a function that associates to each location \( l_i \in L \) an autonomous linear time-invariant differential equation of the form \( \dot{x} = \text{act}_i(x) = A_ix \).

(A2) The jump relation is defined by a function \( M : E \to \mathbb{R}^{n \times n} \) that associates to each edge \( e = (l_i, g_e, l_k) \in E \) a constant matrix in \( \mathbb{R}^{n \times n} \). When the discrete state switches from \( l_i \) to \( l_k \) at time \( \tau \), the continuous state \( x \) is set to \( x(\tau^-) = M_{i,k}x(\tau^-) \).

(A3) For each discrete location \( l_i \in L \), the set of its output edges \( E_i \) can be partitioned in two different sets, namely

\[
E_i = E_{i,c} \cup E_{i,a}
\]  
(1)
depending on the associated guards. More precisely,

\[
E_{i,c} = \{ e \in E_i \mid g_e = inv_i \}
\]  \hspace{1cm} (2)

and

\[
E_{i,a} = \{ e \in E_i \mid g_e \cap inv_i = \emptyset \}
\]  \hspace{1cm} (3)

(A4) All guards associated to edges within the set \(E_{i,a}\) are disjoint sets. Formally:

\[
\forall e, e' \in E_{i,a} \text{ with } e \neq e', \ g_e \cap g_{e'} = \emptyset,
\]  \hspace{1cm} (4)

Moreover, we assume:

\[
inv_i \cup \left( \bigcup_{e \in E_{i,a}} g_e \right) = \mathbb{R}^n.
\]  \hspace{1cm} (5)

We call this HA autonomous because there is no continuous control input and the autonomous edges are uncontrollable.

Note that the assumptions on the structure of the guards of an AHA have several implications.

• Firstly, given an edge \(e = (l_i, g_e, l_h) \in E_{i,a}\) from location \(l_i\) if the continuous state is \(x \in g_e\), then a switch to \(l_h\) should immediately occur. In fact, according to (3) \(x \not\in inv_i\) and the system cannot remain in location \(l_i\). We may call the edge \(e \in E_{i,a}\) autonomous (or equivalently uncontrollable) and the set \(R_{i,a} = \bigcup_{e \in E_{i,a}} g_e\) the autonomous (or equivalently uncontrollable) region.

• Whenever the continuous state reaches the guard \(g_e\), thus activating the edge \(e\), the discrete autonomous behavior of the system is deterministic, because no other switch may occur. In fact, if there exist another output edge \(e'\) from location \(l_i\) (be it controlled or autonomous), then by assumption A4 it holds \(g_e \cap g_{e'} = \emptyset\).

• On the contrary, if the continuous state \(x\) evolve at a given discrete location \(l_i\), within the invariant set \(inv_i\), and there exist an output edge \(e = (l_i, g_e, l_q) \in E_{i,c}\) then the system may either switch to the location \(l_q\) or may keep evolving within location \(l_i\).

We assume that the choice is made by a discrete control agent.

**Example 1** Let us consider the AHA whose graph is reported in Figure 1 where dashed arrows have been used to denote edges associated to autonomous switches, while continuous arrows have been used to denote edges associated to controllable switches.
In this particular $\mathbb{R}^2$ case, guards and invariants of the automaton are homogeneous. In such a case they may be easily described [15] as quadratic forms of $x$. In particular, we assume that the guards associated to autonomous switches are\footnote{To avoid a heavy notation we denote here $g_{i,j}$ the guard associated to edge $e_{i,j}$.}

$$g_{1,2} = \{ x \in \mathbb{R}^2 \mid x^T G_{1,2} x \geq 0 \}, \quad G_{1,2} = \left[ \begin{array}{cc} -0.2 & 0.6 \\ 0.6 & -1 \end{array} \right],$$

$$g_{1,3} = \{ x \in \mathbb{R}^2 \mid x^T G_{1,3} x \geq 0 \}, \quad G_{1,3} = \left[ \begin{array}{cc} 1 & 1.25 \\ 1.25 & 1 \end{array} \right],$$

and

$$g_{2,3} = \{ x \in \mathbb{R}^2 \mid x^T G_{2,3} x \geq 0 \}, \quad G_{2,3} = \left[ \begin{array}{cc} -3 & 0.5 \\ 0.5 & 0 \end{array} \right],$$

where $g_{1,2} \cap g_{1,3} = \emptyset$, thus verifying assumption (A4).

Consequently, by assumption (A4), the invariant sets may be defined as

$$\text{inv}_1 = \mathbb{R}^2 \setminus (g_{1,2} \cup g_{1,3}),$$

$$\text{inv}_2 = \mathbb{R}^2 \setminus g_{2,3},$$

$$\text{inv}_3 = \mathbb{R}^2,$$

while the guards associated to controllable switches are

$$g_{2,1} = \text{inv}_2,$$

$$g_{3,1} = g_{3,2} = \text{inv}_3.$$

The above set of guards and invariants are shown in Figure 2.

\section{Optimal Control Problem}

In this paper we deal with the problem of designing an optimal control policy for an autonomous hybrid automaton $H = (L, \text{act}, \text{inv}, E, M)$ as defined in the previous section.
Before giving a formal definition of the problem it is helpful to introduce some additional notions.

3.1 Preliminary definitions

Let $s = |L|$ be the number of discrete locations and $S \equiv \{1, 2, \cdots, s\}$. Let us define the set

$$\text{succ}_c(i) = \{j \in S : (l_i, inv_i, l_j) \in E_{i,c}\}$$

which denotes the indices associated to the locations reachable from $l_i$, by firing a controllable transition. In the same way we define the set

$$\text{succ}_a(i) = \{j \in S : (l_i, g_{i,j}, l_j) \in E_{i,a}\}$$

which denotes the indices associated to the locations reachable from $l_i$, by firing an autonomous transition.

Indicating by $\bar{A}(t) = e^{At}$ the state transition matrix, we now introduce a notation that turns out to be useful in the problem description and in the development of the minimization algorithm.

**Definition 1 (Sequence of autonomous switches)** Given a state $(i_0, x_0)$ of an AHA we define the sequence $\sigma(i_0, x_0) = \{(i_0, \theta_0), (i_1, \theta_1), \ldots, (i_h, \theta_h)\}$ where $i_k$ is the index of the $k$–th location visited from location $l_{i_0}$ and firing only autonomous edges of the AHA, while $\theta_k \geq 0$ is the time spent in location $l_{i_k}$. Formally the $\theta_k$’s are time intervals such that for $k = 0, \ldots, h$ it holds:

$$x_{k+1} = M_{i_k,i_{k+1}}\bar{A}_{i_k}(\theta_k)x_k$$

$$\forall t \in [0, \theta_k) \quad \bar{A}_{i_k}(t)x_k \in inv_{i_k}$$

$$\bar{A}_{i_k}(\theta_k)x_k \in g_{i_k}$$

$$e_k = (t_{i_k}, g_{i_k}, t_{i_{k+1}}) \in E_{i,a}$$

with $\theta_h = +\infty$.  

\[ (6) \]
Note that the interval $\theta_k$ is the time it takes, once entered in location $l_{i_k}$, to reach the guard of the autonomous edge leading to location $l_{i_{k+1}}$. Therefore $\theta_k = 0$ implies that $x_k \notin \text{inv}_{i_k}$.

**Definition 2 (Bounded automata)** We say that an AHA is bounded if there exists an integer $\hat{h} < +\infty$ such that for all states $(i, x)$ it holds $|\sigma(i, x)| \leq \hat{h}$. ■

Note that this property implies that the automaton is not allowed to evolve autonomously for an infinite number of switches, thus avoiding classical undesired behaviors such as Zeno [12] or instability [4].

**Remark 1** If the graph of an AHA does not have cycles composed of only autonomous edges, then it is bounded. ■

**Proof:** The fact that no cycle composed of autonomous edges exists, is a sufficient (but not necessary) condition to imply that the bound $\hat{h}$ given in Definition 2 is less or equal to the length of the longest directed path containing only autonomous edges. □

As an example, we can immediately conclude that the automaton in figure 1 is bounded because it does not contain any cycle of autonomous edges.

In this paper we will only consider bounded AHA.

We shall now introduce a piecewise constant time function associated to the sequence $\sigma(i_0, x_0)$.

**Definition 3 (Index trajectory)** The index trajectory corresponding to a given sequence $\sigma(i_0, x_0) = \{(i_0, \theta_0), \ldots, (i_h, \theta_h)\}$ is:

$$\varphi_{\sigma}(t) = i_k, \quad \text{if} \quad t \geq \sum_{j=0}^{k-1} \theta_j, \sum_{j=0}^{k} \theta_j$$

(7)

■

**Example 2** Suppose that from a given AHA state $(i, x)$ it has been computed the following sequence $\sigma(i, x)$:

$$\sigma(i, x) = \{(1, 2), (3, 1.5), (2, 2.5), (4, +\infty)\}$$

Then the associated function $\varphi_{\sigma}(t)$ is displayed in Figure 3. ■

### 3.2 Optimal Control Problem

The optimal control problem is based on the assumption that the discrete controller has at most $N$ (fixed a priori) controllable switches available.
We assume that a positive semi-definite matrix $Q_i$ is associated to each discrete location $l_i \in L$. For such a class of hybrid systems we want to solve the following optimal control problem:

$$V^*_N \triangleq \min_{I,T} \left\{ F(I, T) \triangleq \int_0^\infty x^T(t)Q_i(t)x(t)dt \right\}$$

subject to:

$$x(t) = A_i(t)x(t)$$

$$0 = \tau_0 \leq \cdots \leq \tau_k \leq \cdots \leq \tau_{N+1} = +\infty$$

(initial state)

$$i(0) = i_0$$

(initial location)

$$x(0) = x_0$$

(location reached after the $k$-th controlled switch)

$$i(\tau_k) \in \text{succ}_c(i(\tau_k^-))$$

(state reached after the $k$-th controlled switch)

$$x(\tau_k) = M_{i(\tau_k^-),i(\tau_k)} x(\tau_k^-)$$

$$\sigma_k = \sigma(i(\tau_k),x(\tau_k))$$

(auton. sequence)

$$i(\tau_k + \theta) = \varphi_{\sigma_k}(\theta) \text{ for } \theta \in [0, \tau_{k+1} - \tau_k)$$

(auton. index trajectory)

Here function $i(t)$ is composed of $N + 1$ blocks delimited by the instants $\tau_k$’s where the controlled switches occur. Each block is a piecewise constant function: steps internal to the interval $[\tau_k, \tau_{k+1})$ correspond to autonomous switches.

The control variables in this problem are the sequence of controlled switching times $T \triangleq \{\tau_1, \ldots, \tau_N\}$, and the sequence of location indices associated with controllable switches $I \triangleq \{i(\tau_1), \ldots, i(\tau_N)\}$.

We want now to characterize those control problems such that the optimal cost is finite.

**Definition 4 (Ultimate stability)** A location $l_i \in L$ of a bounded AHA is ultimately stable if $\forall x \in \text{inv}_i$ the associated sequence $\sigma(i, x)$ reaches a final dynamics $A_{i_h}$ (that may depend on $x$) such that $A_{i_h}$ is strictly Hurwitz. \( \blacksquare \)
**Proposition 1** A bounded AHA can be stabilized by a switching control law if from every location \( l_i \) not ultimately stable there exists at least a controlled edge leading to an ultimately stable location.

**Proof:** We show that from any initial state \( (i_0, x_0) \) it is possible to steer the continuous state to the origin. In fact from the initial state we can wait until the last location \( l_{i_h} \) of the sequence \( \sigma(i_0, x_0) \) is reached. Obviously if \( A_{i_h} \) is not Hurwitz then \( l_{i_h} \) is not ultimately stable, hence by assumption there exists a controllable switch that leads to an ultimately stable location.

Note that this proposition is a sufficient (but not necessary) condition for the existence of a stabilizing control law. In order to make the problem (8) solvable with finite cost \( V^*_N \), we assume that all considered AHA satisfy Proposition 1.

Finally, in order to express in a more compact way the following results, we recall that for a linear time invariant system of dynamics \( A \) an integral like

\[
J = \int_{\tau}^{\tau + \Delta \tau} x^T(t) Q x(t) dt
\]

with \( Q \geq 0 \) is a quadratic form

\[
J = x^T(\tau) \tilde{Q}(\Delta \tau) x(\tau)
\]

that can be computed numerically or analytically as in [9].

## 4 State feedback control law

In this section we show that the optimal control law for the optimization problem above 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 controllable switch from location \( l_{i_k} \) to \( l_{i_{k+1}} \), or equivalently from linear dynamics \( A_{i_k} \) to \( A_{i_{k+1}} \), should occur.

In particular, we show that for a given location \( l_i \in L \) and for a given controllable switch \( k \in 1, \ldots, N \) it is possible to construct a table \( C^i_k \) that partitions the invariant space \( \text{inv}_i \) into \( s_i \) regions \( R_j \)'s, where \( s_i = |\text{succ}_c(i)| + 1 \), i.e., we can write

\[
\text{inv}_i = R_i \cup \left( \bigcup_{j \in \text{succ}_c(i)} R_j \right).
\]

Whenever \( i(\tau_k + \theta) = i \) we use table \( C^i_k \) to determine if a switch should occur: as soon as the state reaches a point in the region \( R_j \) for a certain \( j \in \text{succ}_c(i) \) a controllable switch will occur and we switch to mode \( i(\tau_{k+1}) = j \); finally, no switch will occur while the system’s state belongs to \( R_i \).
Note that we have presented similar results when dealing with a different class of hybrid systems, namely the switched linear systems [2,3]. However, in that case all switches were assumed to be controllable. In this paper we extend that result to a more general class of hybrid systems, where also autonomous switches may occur.

To avoid repeating the derivation already presented in previous works, we simply show how the tables for the last switch can be computed using the cost function associated to an autonomous evolution. The tables for the intermediate switches can also be constructed using the same dynamic programming arguments given in [2,3].

4.1 Computation of the tables for controllable switches

Consider a state \((i, x)\) and let \(\sigma(i, x) = \{(i_0, \theta_0), \ldots, (i_h, \theta_h)\}\) (where \(i_0 = i\)) be the corresponding sequence of autonomous switches. Let us evaluate the following function:

\[
J_\sigma(i, x, \varrho) = \int_0^\varrho x^T(t)Q_{\varphi_\sigma(t)}x(t)dt = \sum_{k=0}^{h-1} x_k^T \bar{Q}_{i_k}(\theta_k)x_k + x_h^T \bar{Q}_{i_h}(\varrho - \sum_{k=0}^{h-1} \theta_k)x_h
\]

where \(x_0 = x, x_{k+1} = M_{i_k,i_{k+1}} \bar{A}_{i_k}(\theta_k)x_k\) and where \(0 \leq \bar{h} \leq h\) is an integer value that depends on \(\varrho\) through the following inequalities:

\[
\sum_{k=0}^{\bar{h}-1} \theta_k \leq \varrho < \sum_{k=0}^{\bar{h}} \theta_k
\]

The function in (11) represents the cost of the evolution of the system, starting from state \((i, x)\) and only subject to autonomous switches, for a time \(\varrho\).

We will first explain how to build the table of the last controlled switch and then proceed recursively for the others. Assume that \(i_N = i\), i.e., after \(N-1\) controlled switches the current AHA state is \((i, x)\). We show how to compute the table \(C_N^i\). First of all we must create \(\sigma(i, x) = \{(i_0, \theta_0), \ldots, (i_h, \theta_h)\}\).

- Consider first the case in which no controlled switch occurs. The remaining cost starting from \(x\), due to the time-driven evolution and only subject to autonomous switches is

\[
T_0^i(x, i) = J_\sigma(i, x, +\infty).\]

- If the system evolves without performing controlled switches for a time \(\varrho\) and then a controlled switch to \(l_j\) occurs, the remaining cost starting from \(x\) due to the time-driven evolution is

\[
T_l(x, \varrho, j) = J_\sigma(i, x, \varrho) + T_j^*(\bar{x}, j).
\]
\[- \bar{x} = M_{\bar{h}},j \tilde{A}_{\bar{h}} (\varrho - \sum_{k=0}^{\bar{h}-1} \theta_k) x_{\bar{h}} \text{ is the destination point after } \bar{h} \text{ autonomous switches.}\]

The minimization of function (14) has to be performed over \( \varrho \) and over \( j \in \text{succ}_c(i_{\bar{h}}) \) (and note that \( \bar{h} \) depends on \( \varrho \)). This minimization problem can be written as

\[
\min_{0 \leq h \leq \bar{h}} \min_{j \in \text{succ}_c(i_{\bar{h}})} \min_{\varrho \in \mathcal{I}_{\bar{h}}} T_i(x, \varrho, j), \tag{15}
\]

where \( \mathcal{I}_{\bar{h}} \) is the time interval defined by the inequalities in (12).

Let us denote by \( \varrho^*(i, x) \) and \( j^*(i, x) \) the values of \( \varrho \) and \( j \) that minimize (15). We may now indicate

\[
T_i^*(x, j^*(i, x)) = T_i(x, \varrho^*(i, x), j^*(i, x)) \tag{16}
\]

We now show how these data are used to construct the tables for the last controllable switch.

In presence of autonomous switching regions the state space available for controllable partitions is only the \( \text{inv}_i \). Such subspace will be then partitioned into \( \mathcal{R}_j \) regions according to the following criterion:

- \( x \in \mathcal{R}_i \) if \( \varrho^*(i, x) > 0 \); this physically means that the optimal strategy is to remain for a non zero time \( \varrho \) in location \( l_i \);
- \( x \in \mathcal{R}_{j^*(i, x)} \) if \( \varrho^*(i, x) = 0 \); this physically means that the optimal strategy is to immediately switch to location \( l_{j^*} \).

Once the table for the last switch is constructed, it is simple to build all the others following the principle of dynamic programming and solving problem (15) recursively over the total number of allowed controllable switches as in [2].

5 The homogeneous case

We present now a particular class of AHA where the structure of the guards and invariants is homogeneous. Firstly we recall that a guard \( g_e \) is homogeneous if

\[
(\forall x \in g_e, \ \forall \lambda \in \mathbb{R}) \ \lambda x \in g_e.
\]

Such case is meaningful because it allows one to describe guards of the form \( x^T(t)Zx(t) \geq 0 \), where \( x(t) \) is the continuous state of the hybrid system, i.e., guards given by quadratic forms. A physical example of this is given by an electric system whose threshold \( x_1(t)x_2(t) > 0 \) (here \( x_1(t) \) and \( x_2(t) \) are voltage and current, resp.) denotes the condition where the system behaves as a power generator.

Moreover, as we show in the following remark, in such conditions the computational complexity of the off-line to compute the switching regions is reduced.
Remark 2 For each state \((i, x)\) of an AHA with homogeneous guards, \(\sigma(i, x)\) is a homogeneous function with respect to its second variable, i.e., \(\forall \lambda \in \mathbb{R} \setminus \{0\}\)

\[
\sigma(i, x) \equiv \sigma(i, \lambda x).
\]

This obvious fact implies that the residual cost \(J_\sigma(i, x, \varrho)\) given in Section 4.1 can be calculated only in the points \(y\) of the unitary semisphere. In fact, knowing \(J_\sigma(i, y, \varrho)\), clearly \(J_\sigma(i, x, \varrho) = \lambda^2 J_\sigma(i, y, \varrho), x = \lambda y\).

As a consequence a discretization of the all invariant set \(inv_i\) is no longer required, because all the necessary information to construct the optimal switching tables can be calculated along the unitary semisphere. Hence this special case reduces the computational complexity of the construction of table \(C_{i,k,N}^\ast\) [2] from \(O((s_i - 1)r^n)\) for the general AHA, to \(O((s_i - 1)r^{n-1})\), where we indicate by \(s_i\) the number of controllable edges of location \(l_i\), \(r\) is the discretization sampling along each direction, \(n\) is the state space dimension.

6 An example in the homogeneous case

Let us consider again the AHA in Example 1 whose structure is shown in Figure 1. This automaton is also homogeneous, thus it allows to perform calculations along the unitary semisphere. Let us assume that the activity functions at the discrete locations are defined by the following matrices:

\[
A_1 = \begin{bmatrix}
-1.85 & -1 \\
1 & 0
\end{bmatrix} \quad A_2 = \begin{bmatrix}
0 & 1 \\
-0.74 & -1.29
\end{bmatrix} \quad A_3 = \begin{bmatrix}
-2.75 & -2.84 \\
1 & 0
\end{bmatrix}
\]

Moreover, we assume that all jumps are coincident with the identity relation, i.e., \(M_{i,j} = I\), for all \(i, j\) with \(i \neq j\), where \(I\) denotes the second order identity matrix.

Finally we assume that weighting matrices as well are coincident with the identity matrix.

To solve the resulting optimal control problem, we first evaluate off-line the \(N \times |L|\) controllable switching tables, using the procedure presented in the previous subsection.

In this particular case 9 tables have been constructed (3 for every switch).

A space discretization of 101 points along the unitary semi-sphere and a local minimum search within five time constants have been considered sufficiently fine. Provided such tables, the controller/supervisor is ready (and fast) to estimate the optimal strategy in real time mode subject to the constraints of the automaton. The state trajectory that minimizes the performance index is depicted in Figure 4, where the black squares indicate the controllable switches and the red stars indicate the autonomous switches.
Finally, we found out the following values of the switching (both controllable and autonomous) instants $T$, of the optimal sequence $I$, and of the optimal cost $J$:

$$T = \{0.05, 0.11, 0.78, 0.96, 1.505, +\infty\}$$

$$I = \{3 \Rightarrow 1 \rightarrow 3 \Rightarrow 2 \rightarrow 3 \Rightarrow 2 \rightarrow 3\}$$

$$J = 62.15$$

In the subset $I$ the arrow $\Rightarrow$ indicates a controllable switch, and the arrow $\rightarrow$ indicates an autonomous switch.

The system initially sojourns in location $l_3$ then the supervisor switches to $l_1$. Tables indicate that it is worth waiting until the autonomous threshold with location $l_3$, in order to go directly to $l_2$ in zero time. Now it is better to remain in $l_2$ until the autonomous boundary is reached before using the third controllable switch, which takes place during the evolution in location $l_3$. From now on the system evolves independently towards zero, performing a finite number of autonomous switches, due to the assumptions of section 2.1.

7 Conclusions

We have defined a new class of hybrid systems called Autonomous Hybrid Automata that can be seen as a generalization of the class of switched system we have considered in [2,3,9]. In this new model there are two types of edges: a controllable edge represents a mode switch that can be triggered by the controller; an autonomous edge represents a mode switch that is triggered by the continuous state of the system as it reaches a given threshold.

We have shown how the special structure of autonomous hybrid automata allows one to solve an infinite horizon quadratic optimization problem with a numerically viable...
procedure; the optimal control law takes the form of a state-feedback.

As the modelling power of Autonomous Hybrid Automata is not rich enough to encompass all cases of practical interest, our future work will focus on extending the presented approach to a larger class of hybrid automata.

References


