Optimal hybrid control for switched affine systems under safety and liveness constraints*

D. Corona, C. Seatzu, A. Giua, D. Gromov, E. Mayer, J. Raisch[‡§¶]

Abstract

In this contribution, we propose a two-level hierarchical control strategy to solve the problem of minimising a cost function for a switched affine system under safety and liveness constraints. The constraints are handled on the lower level by a discrete supervisory controller designed using l-complete approximation. Its action can be represented as state invariants added to the plant model. In a bottom-up strategy, we can then synthesise a high-level controller, which computes a set of switching tables using the remaining degrees of freedom to optimise a quadratic performance index.

1 Introduction

Switched affine systems are a particular class of hybrid systems consisting of a number of subsystems, each described by a set of affine differential equations, and a switching law that determines the active subsystem at any instant of time. The problem of optimal control synthesis for this class of systems has been widely investigated during the last decade, and many results can be found in the control and computer science literature. However, existing approaches are not able to take hard safety constraints into account. Clearly, this poses a major problem in many application areas. Supervisory control approaches for switched systems, on the other hand, are able to handle safety constraints but do not attempt to optimise a given performance index. In this contribution, we merge supervisory control and optimal control techniques to provide a method for synthesing a closed loop control strategy which minimises a given cost function under certain safety and liveness constraints.

The proposed approach uses a two-level hierarchical philosophy. The low-level step consists in the design of a discrete supervisory controller which suitably adjusts the discrete input depending on the observed output events. In particular, based on the notion of l-complete approximations

*Published as: D. Corona, C. Seatzu, A. Giua, D. Gromov, E. Mayer, J. Raisch, " Optimal hybrid control for switched linear systems under safety and liveness constraints," *Proc. 2004 CACSD (Taipei, Taiwan)*, Sep 2004.

†D. Corona, C. Seatzu, A. Giua are with the Department of Electrical and Electronics Engineering, University of Cagliari, Italy {daniele.corona,giua,seatzu}@diee.unica.it

‡D. Gromov is with the Lehrstuhl für Systemtheorie technischer Prozesse, Otto-von-Guericke-Universität Magdeburg Dmitry.Gromov@e-technik.uni-magdeburg.de

§E. Mayer is with the Systems and Control Theory Group, Max-Planck-Institut für Dynamik komplexer technischer Systeme, Magdeburg eckart.mayer@mpi-magdeburg.mpg.de

¶J. Raisch is with the Lehrstuhl für Systemtheorie technischer Prozesse, Otto-von-Guericke-Universität Magdeburg and with the Systems and Control Theory Group, Max-Planck-Institut für Dynamik komplexer technischer Systeme, Magdeburg Joerg.Raisch@e-technik.uni-magdeburg.de

[7, 12], [9] and on the supervisory control theory of Ramadge and Wonham [13] we design a discrete supervisor that guarantees safety and liveness constraints.

We show that the action of the resulting supervisor is equivalent to the addition of appropriate state invariants to the hybrid automaton that describes the plant model. At this point the high-level controller may be used to minimize a given performance index, that takes into account the constraints provided at the low-level step. In particular, in this paper the proposed controller implements a feedback control law that is designed using the approach illustrated in detail in [5, 6], that is based on the construction of appropriate "switching tables". Note that this approach is an extension of the procedure firstly presented in [5, 8, 14].

2 Plant Model and Specifications

In this section we first define the class of Hybrid Automata (HA) on which we focus our attention. Then we formally describe the considered optimal control problem.

2.1 Hybrid Automata

A 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, 11]. The hybrid automaton considered here is a structure $H = (L, act, inv, E, Jump)$ defined as follows, in consistency with the current literature definitions (see for instance [2]).

- $L = \{l_1, \cdots, l_s\}$ is a finite set of locations.
- $act : L \to Inclusions$ is a function that associates to each location $l_i \in L$ a differential inclusion.
- $inv : L \to Invariants$ is a function that associates to each location $l_i \in L$ an invariant $inv_i \subseteq \mathbb{R}^n$. An invariant function is $x \in inv_i$. The *true* value of the invariant function allows the continuous state to evolve within the discrete location l_i.
- $E \subset L \times Guards \times L$ 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 switch reaching the location l'.
- A jump relation is $j_e \subset \mathbb{R}^n \times \mathbb{R}^n$ associated to an edge e. When the edge is activated, x is set to x' provided $(x, x') \in j_e$.

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_0} to l_{i_1}, a switch can be made to location l_{i_1}. During the jump at time τ, 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.

In this paper we make the following assumptions.

(A1) The differential inclusion is an affine equation of the form $\dot{x} = A_i x + b_i$.

(A2) The state evolution is continuous, thus the jump relation is equal to the identity.

2.2 Safety specification

We also assume that some constraints on the admissible state trajectory are given. In particular, these constraints are expressed via a discrete automaton that is based on a partition of the state space $X \subset \mathbb{R}^n$. To each element of this partition we associate an output signal Y_d. A discrete automaton SP_Y is used to restrict the set of the admissible sequences of output signals.

In the following we call *safety constrains* the constraints that originate from the structure of the hybrid automaton HA, and the constraints on the output sequences given by the discrete automaton SP_Y. An example of this specification will be given in Section V.

2.3 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 the above class of hybrid systems we want to solve the following optimal control problem:

$$
\begin{aligned}
V_N^* &\triangleq \min_{I,\mathcal{T}} \{F(I,\mathcal{T}) \\
&\triangleq \int_0^\infty (x(t)-x_{eq})^T Q_{i(t)} (x(t)-x_{eq}) dt\} \\
& \\
\text{s.t.}\quad & \dot{x}(t) = A_{i(t)} x(t) + b_{i(t)} \\
& 0 = \tau_0 \leq \cdots \leq \tau_k \leq \cdots \leq \tau_{N+1} = +\infty \\
& \qquad\qquad \text{(controlled switching times)} \\
& i(0) = i_0 \quad \text{(initial location)} \\
& x(0) = x_0 \quad \text{(initial state)} \\
& i(\tau_k) \in \ succ(i(\tau_k^-)) \\
& \qquad\qquad \text{(location reached after the} \\
& \qquad\qquad\quad k-\text{th controlled switch)} \\
& x(\tau_k) = x(\tau_k^-) \\
& \qquad\qquad \text{(state reached after the} \\
& \qquad\qquad\quad k-\text{th controlled switch)} \\
& x(t) \in X \subset \mathbb{R}^n, \ \ \forall\, t \geq 0
\end{aligned}
\tag{1}
$$

Here x_{eq} is the target state, $succ(i)$ is the set of locations reachable from l_i with only one switch; $X \subset \mathbb{R}^n$ is a physical constraint on the continuous state.

Note that in order to obtain a finite optimal cost, we make the following additional assumptions:

(A3) there should exist a location l_i such that $x_{eq} = -A_i^{-1} b_i$ with A_i strictly hurwitz;

(A4) the number of available switches N should be great enough to reach, from the initial location l_{i_0}, the location l_i that satisfies assumption (A3).

The control variables in this problem are the sequence of controlled switching times $\mathcal{T} \triangleq \{\tau_1, \ldots, \tau_N\}$, and the sequence of location indices associated with switches $I \triangleq \{i(\tau_1), \ldots, i(\tau_N)\}$. In this paper we propose a solution to the above optimal control problem that is based on two hierarchical levels.

3 The low-level task

The low-level step consists in the definition of the invariant sets that guarantee that the discrete output sequences obey the imposed specification (SP_Y). The resulting hybrid automaton does not posses blocking states, i.e. it guarantees the liveness of the overall system. The resulting system is considered as an input for the top-level optimisation procedure.

The hybrid plant model is converted to a purely discrete one via the l-complete approximation approach [9,12]. Subsequently, Ramadge and Wonham's supervisory control theory [13] is implemented to synthesise a least restrictive supervisor. If the hybrid plant is interpreted as a hybrid automaton, attaching the supervisor is equivalent to adding invariants to this automaton.

3.1 Ordered set of discrete abstractions

Let us now restrict our attention to the class of switched affine systems that evolve in discrete time and generate discrete-valued outputs. The sampling interval is denoted by Δt. Furthermore we assume that all processes in our overall system are synchronised (i.e. they operate on a common time scale). The model of the plant is then described by the set of time-invariant difference equations

$$\begin{aligned} x(t_{k+1}) &= f_{\psi(t_k)}(x(t_k)), \\ y_d(t_{k+1}) &= q_y(x(t_{k+1})) \end{aligned} \tag{2}$$

where $k \in \mathbb{N}_0$ is the time index, $t_k \in \mathbb{T} = \{k\Delta t\}_{k \in \mathbb{N}_{\not{\!\!0}}}$; $x \in X \subset \mathbb{R}^n$, $\{f_{\psi^{(i)}}(x) = A_i x + b_i : \psi^{(i)} \in \Psi\}$ is a family of affine state transition maps from X into $\mathbb{R}^n$ that is parameterised by some finite index set $\Psi = \{\psi^{(1)}, \ldots, \psi^{(\alpha)}\}$; $\psi : \mathbb{T} \to \Psi$ is a switching signal which can be interpreted as a discrete control input.

$y_d \in Y_d^{\mathbb{T}}$ is a discrete-valued measurement signal. The set of output symbols, Y_d, is assumed to be finite: $Y_d = \{y_d^{(1)}, \ldots, y_d^{(\beta)}\}$, and $q_y : X \to Y_d$ is the output map. Without loss of generality, the latter is supposed to be surjective (*onto*). The output map partitions the state space into a set of disjoint subsets $Y^{(i)} \subset X$, $i = 1, \ldots, \beta$, i.e.

$$\bigcup_{i=1}^{\beta} Y^{(i)} = X,$$

$$Y^{(i)} \cap Y^{(j)} = \emptyset \quad \forall i \neq j \, .$$

To implement supervisory control theory, the hybrid plant model is approximated by a purely discrete one. This is done using the method of l-complete approximation [9, 12], which is described in the following paragraphs.

Denote the behaviour of the hybrid plant model by $\mathcal{B}_{plant}$, i.e. $\mathcal{B}_{plant} \subseteq (\Psi \times Y_d)^{\mathbb{T}}$ is the set of all pairs of (discrete valued) input/output signals $w = (\psi, y_d)$ that (2) admits. In general, a time-invariant system with behaviour $\mathcal{B}$ is called l-complete if $w \in \mathcal{B} \Leftrightarrow \sigma^t w|_{[t_0,t_l]} \in \mathcal{B}|_{[t_0,t_l]}$ $\forall t \in \mathbb{T}$, where σ is the backward shift operator and $w|_{[t_0,t_l]}$ denotes the restriction of the signal w to the domain $[t_0, t_l]$ [15]. Hence, for l-complete systems we can decide whether a signal belongs to the system behaviour by looking at intervals of length l. Clearly, an l-complete system can be represented by a difference equation in its external variables with lag l. The hybrid plant model (2) is, except for trivial cases, not l-complete. For such systems, the notion of *strongest l-complete approximation* has been introduced in [9]: a time-invariant dynamical system with behaviour $\mathcal{B}_l$ is called strongest l-complete approximation for $\mathcal{B}_{plant}$ if

$$
\begin{array}{ll}
(i) & \mathcal{B}_l \supseteq \mathcal{B}_{plant}, \\
(ii) & \mathcal{B}_l \text{ is } l\text{-complete}, \\
(iii) & \mathcal{B}_l \subseteq \tilde{\mathcal{B}}_l \text{ for any other } l\text{-complete } \tilde{\mathcal{B}}_l \supseteq \mathcal{B}_{plant},
\end{array}
$$

i.e. if it is the "smallest" l-complete behaviour containing $\mathcal{B}_{plant}$. Obviously, $\mathcal{B}_l \supseteq \mathcal{B}_{l+1}$ $\forall l \in \mathbb{N}$, hence the proposed approximation procedure may generate an ordered set of abstractions. Clearly, $w \in \mathcal{B}_l \Leftrightarrow w|_{[t_0,t_l]} \in \mathcal{B}_{plant}|_{[t_0,t_l]}$. For $w|_{[t_0,t_l]} = (\psi^{i_0}, \ldots, \psi^{i_l}, y_d^{i_0}, \ldots, y_d^{i_l})$ this is equivalent to

$$
\begin{aligned}
&f_{\psi^{(i_{l-1})}}\left(\cdots f_{\psi^{(i_1)}}\left(f_{\psi^{(i_0)}}\left(q_y^{-1}(y_d^{(i_0)})\right) \cap \left(q_y^{-1}(y_d^{(i_1)})\right)\right)\right. \\
&\left.\ldots (q_y^{-1}(y_d^{i_{(l-1)}})\right) \cap q_y^{-1}(y_d^{(i_l)}) := X(w|_{[t_0,t_l]}) \neq \emptyset.
\end{aligned}
\tag{3}
$$

Note that for a given string $w|_{[t_0,t_l]}$, $X(w|_{[t_0,t_l]})$ represents the set of possible values for the continuous state variable $x(t_l)$ and that (3) does not depend on $\psi^{(i_l)}$. For switched affine systems evolving on discrete time $\mathbb{T}$, (3) can be checked *exactly.* For switched affine system evolving on continuous time and special classes of nonlinear systems, $X(w|_{[t_0,t_l]})$ can be safely overapproximated, hence (3) can be checked "conservatively" (e.g. [7, 10]). This will still lead to an l-complete approximation but, in general, not a strongest l-complete approximation.

As both input and output signal evolve on finite sets, Ψ and Y_d, $\mathcal{B}_l$ can be realised by a (nondeterministic) finite automaton. In [9, 12], a particularly intuitive realisation is suggested, where the approximation state variable stores information on past values of ψ and y_d. More precisely, the automaton state set can be defined as

$$
X_d := \bigcup_{j=0}^{l-1} X_{d_j}
$$

where $X_{d_0} = Y_d$
and $X_{dj} = \{(\psi^{(i_0)}, \ldots, \psi^{(i_{j-1})}, y_d^{(i_0)}, \ldots, y_d^{(i_j)})\}$ such that $\exists \psi^{(i_j)} \in \Psi : (\psi^{(i_0)}, \ldots, \psi^{(i_j)}, y_d^{(i_0)}, \ldots, y_d^{(i_j)}) \in \mathcal{B}_l|_{[t_0,t_j]}$.

As the states $x_d^{(j)}$ of the approximation realisation are strings of input and output symbols, we can associate $x_d^{(j)}$ with a set of continuous states, $X(x_d^{(j)})$, in completely the same way as in (3).

Note that the transition function $\delta : X_d \times \Psi \to 2^{X_d}$ follows immediately from $\mathcal{B}_l$ and that we can associate $y_d^{(i_j)}$ as the unique output for each discrete state $x_d^{(j)} \in X_d$. The resulting (non-deterministic) Moore-automaton $M_l = (X_d, \Psi, Y_d, \delta, \mu, X_{d_0})$ with state set X_d, input set Ψ, output set Y_d, transition function δ, output function μ, and initial state set X_{d_0} is then a realisation of $\mathcal{B}_l$. Note that the state of M_l is instantly deducible from observed variables.

To recover the framework of supervisory control theory [13] as closely as possible, we finally convert M_l into an equivalent automaton without outputs, $G_l = (\tilde{X}_d, \Psi \times Y_d, \tilde{\delta}, \tilde{X}_{d_0})$, where Ψ represents the set of controllable events and Y_d the set of uncontrollable events.

3.2 Specification and supervisor design

Safety requirements can often be formalised as a set of acceptable pairs of input/output signals. In many applications we have independent specification behaviours for both inputs and outputs, $\mathcal{B}_\Psi \subseteq \Psi^{\mathbb{T}}$, $\mathcal{B}_{Y_d} \subseteq Y_d^{\mathbb{T}}$, which are assumed to be m_Ψ and m_Y-complete. They can hence be realised by finite automata $SP_\Psi = (S_\Psi, \Psi, \delta_\Psi, S_{\Psi 0})$ and $SP_Y = (S_Y, Y_d, \delta_Y, S_{Y0})$.

An example for the specification of the outputs is represented in Figure 2. Let's suppose that the set of output symbols can be represented as a union of three sets: $Y_d = Y_d^+ \bigcup Y_d^0 \bigcup Y_d^-$, where Y_d^+, Y_d^0, and Y_d^- are the sets of symbols that correspond to safe, conditionally safe, and unsafe regions of the continuous state space. Thus, the specification requires that the state can belong to the conditionally safe region not more than two time intervals. After that the system should stay at least one time interval within the safe region producing the corresponding output symbol.

The overall specification is then easily obtained by forming the shuffle product of SP_Ψ and SP_Y (e.g. [3]),

$$SP = (S, \Psi \cup Y_d, \delta_{SP}, S_0)$$

where $S = S_\Psi \times S_Y$, $S_0 = S_{\Psi 0} \times S_{Y0}$. SP realises the concurrent behaviour of SP_Ψ and SP_Y.

Given an approximating automaton G_l and a specification automaton SP, supervisory control theory checks, whether there exists a nonblocking supervisor and, if the answer is affirmative, provides a least restrictive supervisor SUP via "trimming" of the product of G_l and SP. Hence the state set of the supervisor, X_{SUP}, is a subset of $\tilde{X} \times S$.

The functioning of the resulting supervisor is very simple. At time t_k it "receives" a measurement symbol which triggers a state transition. In its new state $x_{sup}^{(j)}$, it enables a subset $\Gamma(x_{sup}^{(j)}) \subseteq \Psi$ and waits for the next feedback from the plant. As shown in [9], the supervisor will enforce the specifications not only for the approximation, but also for the underlying hybrid plant model (2).

In the following, we will be interested in the special case of *quasi-static* specifications. To explain this notion, let $p_{app} : X_{SUP} \to \tilde{X}$ denote the projection of $X_{SUP} \subseteq \tilde{X} \times S$ onto its first component. If p_{app} is injective, the specification automaton is called quasi-static with respect to the approximation automaton G_l.

Proposition 1. *S is quasi-static with respect to G_l if*

$$l \geq max(m_\Psi, m_Y).$$

3.3 Closed loop model

We now interprete the hybrid plant model (2) as a hybrid automaton with locations $\psi^{(1)}, \ldots, \psi^{(\alpha)}$ and attach the supervisor SUP. For the case of quasi-static specifications, each supervisor state $x_{sup}^{(i)}$ corresponds exactly to a state $\tilde{x}_d^{(i)} = p_{app}(x_{sup}^{(i)})$ of the approximating automaton, which, in turn, can be associated with a set $X(\tilde{x}_d^{(i)}) = X(p_{app}(x_{sup}^{(i)}))$.

Attaching the supervisor to the hybrid plant automaton therefore boils down to adding invariants to each location

$$inv(\psi^{(j)}) = \bigcup_{\substack{i,\, p_{app}(x_{sup}^{(i)}) \in \tilde{X}_{d_{l-1}} \\ \psi^{(j)} \in \Gamma(x_s^{(i)})}} X(p_{app}(x_{sup}^{(i)})),$$

where $\tilde{X}_{d_{l-1}} = X_{d_{l-1}}$. Union of all invariants forms the refined, safe state space that contains only safe points, i.e. points for which exists at least one sequence of control symbols such that the resulted behaviour satisfies the specification.

The resulting hybrid automaton is guaranteed to obey the specification but retains degrees of freedom, which can be used in a separate optimal control layer.

4 The high-level task

The high-level task requires the solution of an optimal control problem of the form (1) with the additional two constraints:

(C1)

$$x(t) \in inv_{i(t)} \quad \forall\, t \geq 0$$

i.e., an invariant set is associated to each discrete location;

(C2)

$$\tau_k - \tau_{k-1} \geq \Delta, \quad \forall\, k = 1, \ldots, N+1$$

i.e., a minimum permanence time is imposed in each location.

The first constraint is given by the low level step, and the second constraint comes from the synchronization between the two control levels.

In this paper we build on the results presented in [5] and extend the state feedback control technique based on the construction of ”switching tables” to also deal with HA with *invariant sets* and affine dynamics [14].

The main advantage of the proposed procedure may be briefly summarized as follows.

- It is guaranteed to find the optimal solution to problem (1) enriched with constraints (C1) and (C2).
- It provides the global optimal solution, i.e., the tables may be used to determine the optimal state feedback control law for all initial states.

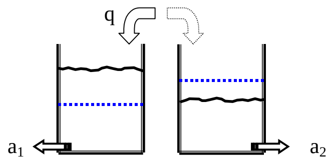

Figure 1: Schematic view of the physical system considered for the example.

The optimal control law can be computed as follows. For a given location $l_i \in L$ and for a given switch $k \in \{1, \ldots, N\}$, it is possible to construct a table $\mathcal{C}_k^i$ that partitions the invariant set inv_i into up to $s_i = succ(i) + 1$ regions $\mathcal{R}_j$'s. Whenever $i_{k-1} = i$ we use table $\mathcal{C}_k^i$ to determine if a switch should occur: as soon as the state reaches a point in the region $\mathcal{R}_j$ we will switch to mode $i_k = j$ provided that the minimum permanence time Δ has elapsed; on the contrary no switch will occur while the system state belongs to $\mathcal{R}_i$.

Switching tables can be computed off-line using dynamic programming arguments. To avoid repeating the derivation already presented in previous works we will not provide here details on their construction. We address the interested reader to [5, 14].

Remark 1. Note however that in the considered problem an important caution should be taken when constructing the switching tables and in particular, when considering the successors of the current location l_i. In fact, let us consider the edge $e = (l_i, g_e, l_j)$. It may be activated when $x \in inv_j$ but two different cases may occur.

1. The continuous state $x \in inv_i \bigcap inv_j$. In this case the discrete controller has the degree of freedom between keeping the evolution in l_i or switching to l_i.

2. The continuous state $x \notin inv_i$. The evolution cannot continue in l_i thus the discrete controller must leave location l_i.

This implies that the set of "admissible" successors also depends on the current continuous state. ■

5 Numerical example

As an example of the described procedure, the following problem is considered. This problem was inspired by [4]. A physical system is composed of two cylindric tanks, equipped with inflow pipe and subject to leakage (see Figure 1).

The continuous variables of this system are the levels of the fluid in each tank, namely $x = [x_1, x_2]$.

The physical dimension of the tanks imposes a minimum and a maximum value of the fluid level, i.e., $x \in X = [0, 30] \times [0, 20]$.

The inflow pipe is capable of a flow rate $q = 60$, but it may only assume quantized positions taken from a finite set $\mathcal{Q} = \{b_1, \ldots, b_5\}$ where $b_1 = [q, q]$, $b_2 = [0, q]$, $b_3 = [q, 0]$, $b_4 = [0, 0]$ and $b_5 = [q/2, q/2]$.

An approximated model of the system can be given in terms of a differential equation $\dot{x} = Ax + b_i$,

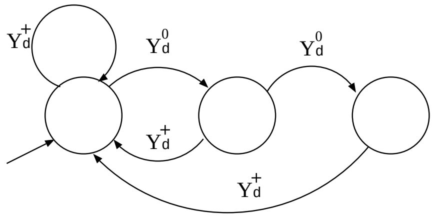

Figure 2: Specification for the outputs

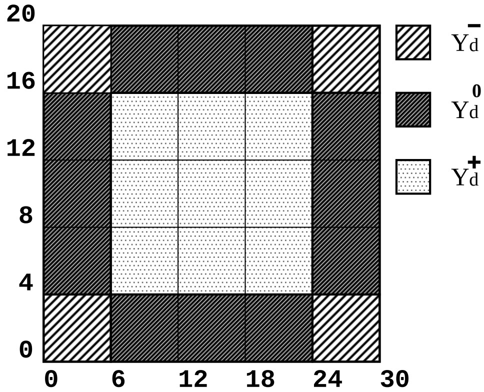

Figure 3: Partitioning of the state space

$i = 1 \dots 5$, where $A = \text{Diag}\{-2, -3\}$ represents the linearized fluid loss due to static height pressure.

The resulting system can be modeled as a hybrid automaton composed of 5 locations, as depicted in Figure 4. The structure of the automaton takes into account the order in which the different inflow rates can be changed.

The specification of the outputs is represented in Figure 2. Let us suppose that the set of output symbols can be represented as a union of three sets: $Y_d = Y_d^+ \bigcup Y_d^0 \bigcup Y_d^-$, where Y_d^+, Y_d^0, and Y_d^- are the sets of symbols that correspond to safe, conditionally safe, and unsafe regions of the continuous state space, respectively. For our particular case these regions are shown in Figure 3. Thus, the specification requires that the state can belong to the conditionally safe region for no longer than two time intervals. After that, the system should stay at least one time interval within the safe region producing the corresponding output symbol.

In the low-level step the hybrid system is converted into a purely discrete one using l-complete approximation approach described above. The parameter l is chosen to satisfy the requirements of Proposition 1. Then, supervisory control theory is applied to synthesise a least restrictive nonblocking supervisor. The obtained supervisor is represented as a set of invariants which are attached to the resulting hybrid automaton. These invariants restrict the behaviour of the overall system to guarantee the safety and liveness conditions. As an example, the invariant set of location l_1 is reported in Figure 5.

The high-level step requires the solution of an optimal control problem of the form (1) with the additional constraints (C1) and (C2).

The weighting matrices Q_i are indicated for each location in Figure 4 where I denotes the identity matrix.

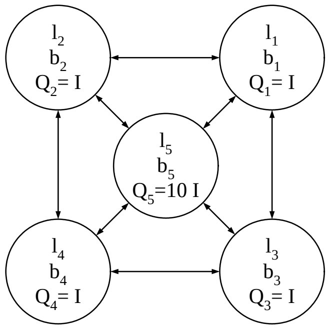

Figure 4: The HA modeling the considered affine system. The double arrows indicate that both switching directions are allowed.

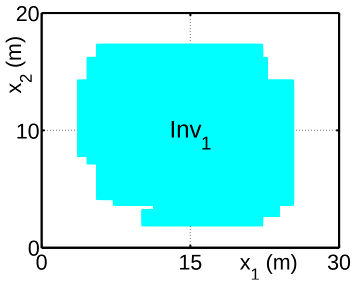

Figure 5: Invariant region for location l_1.

The maximum number of switches is $N = 3$.

The target state is set equal equal to $x_{eq} = [15, 10]$, that satisfies assumption (A3) for $i = 5$.

The off-line part of this procedure consists in the construction of $5 \times 3 = 15$ tables, one per each location and per number of available switches. A state space discretization is a grid of 125×125 points. The minimum search algorithm works on a time domain of $t = 5s$ with time step $0.1s$. The latter value was chosen to guarantee an appropriate synchronization between the two levels. The offline calculation effort for this step of the problem took approximately 2 hours, on a common commercial laptop with average up to date performances. For sake of brevity we only report some of these tables (depicted in Figure 6), i.e., those tables used by the controller during the simulation ran for the initial continuous state $x = [21, 3]$ and the initial location l_1. The trajectory obtained for this particular value of the initial state is plotted in Figure 7. The optimal switching sequence and switching times are $I = \{l_1, l_2, l_4, l_5\}$ and $\mathcal{T} = \{0.121, 0.221, 0.321\}$, and the optimal cost is $V_3^* = 19.47$.

Note that, due to the minimum permanence time within each location, it may occur that the switching from one discrete location to another, does not necessarily occur as soon as the state trajectory exits the current region. This is the case of the last switching point of the trajectory reported in Figure 7.

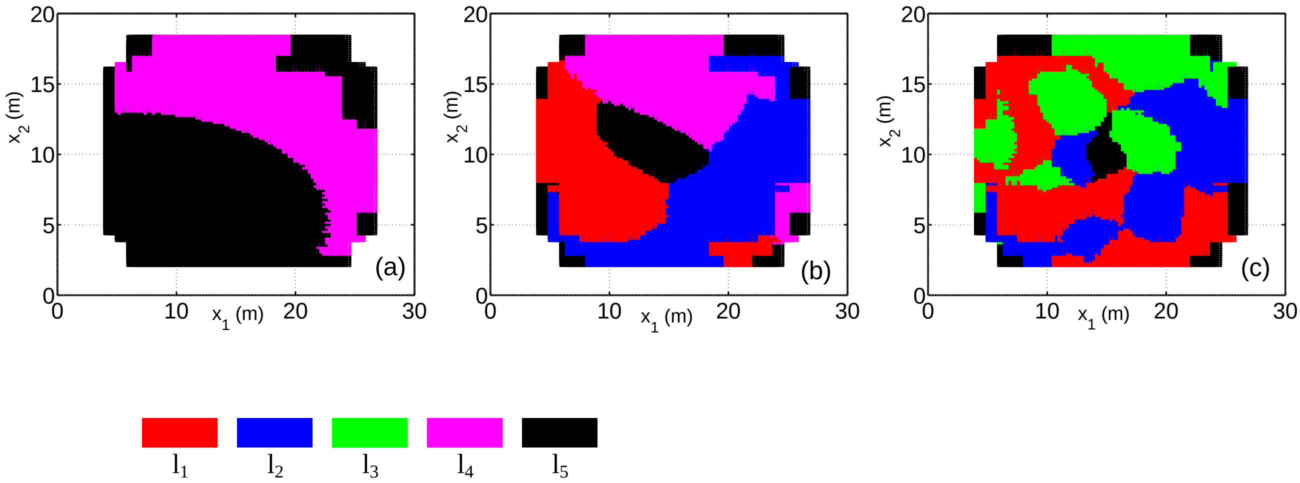

Figure 6: Switching tables used by the controller during the simulation described in the example. (a) is the table used when 1 switch is available, (b) when 2 switches are available, (c) when 3 switches are available.

6 Conclusion

In this paper two approaches based respectively on discrete approximation of continuous systems and optimal control of switched systems, were successfully casted and merged to the framework of a hybrid automaton. The proposed procedure consists in two hierarchical levels of control. The low-level is responsible of gross safety specification and constraints the behavior of HA. More precisely, it constructs the invariant sets that govern the permanence in each location of the HA. The high-level controller applies the already existing results on optimal control of switched systems to the resulting HA. Both methods develop the major computational effort offline, and they result in a feedback control law.

References

[1] R. Alur and D.L. Dill. A theory of timed automata. In *Theoretical Computer Science*, 1994.

[2] E. Asarin, O. Bournez, T. Dang, O. Maler, and A. Pneuli. Effective synthesis of switching controllers for linear systems. *Proceedings of the IEEE*, 88(7), 2000.

[3] C.C. Cassandras and S. Lafortune. *Introduction to discrete event systems*. Kluwer Academic Publishers, 1999.

[4] C. Chase, J.Serrano, and P.J.Ramadge. Periodicity and chaos from switched flow systems: contrasting examples of discretely controlled continuous systems. *IEEE Trans. Automatic Control*, 38, 1993.

[5] D. Corona, A. Giua, and C. Seatzu. Optimal control of hybrid automata: an application to the design of a semiactive suspension. *Control Engineering Practice*, 2004.

[6] A. Giua D. Corona and C. Seatzu. Optimal feedback switching laws for autonomous hybrid automata. In *Proc. IEEE Int. Sym. on Intelligent Control*, Taipei, Taiwan, September 2004.

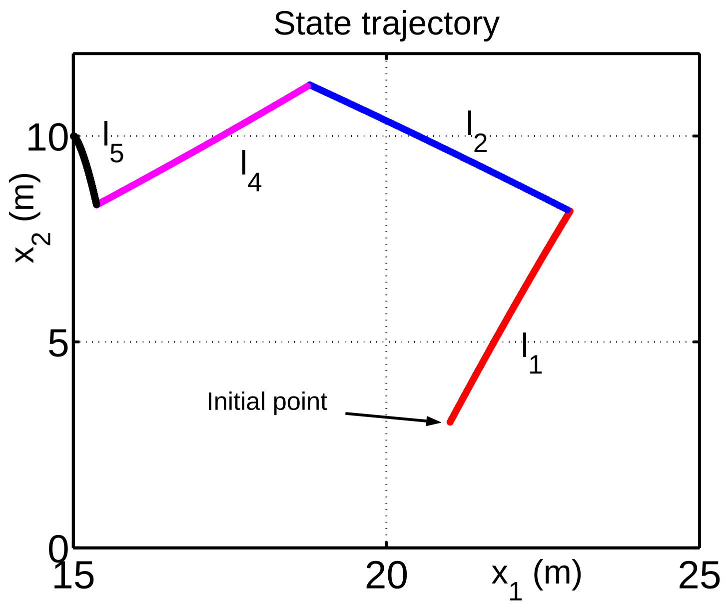

Figure 7: State space trajectory and discrete location sequence.

[7] D. Franke, T. Moor, and J. Raisch. Supervisory control of switched linear systems. *at–Automatisierungstechnik.* Special Issue on Hybrid Systems I: Analysis and Control, 48(9), 2000.

[8] A. Giua, C. Seatzu, and C. Van Der Mee. Optimal control of switched autonomous linear systems. In *Proc. 40th Conf. on Decision and Control*, Orlando, FL, USA, 2001.

[9] T. Moor and J. Raisch. Supervisory control of hybrid systems within a behavioural framework. *Systems & Control Letters*, 38:157–166, 1999. Special issue on *Hybrid Control Systems.*

[10] T. Moor and J. Raisch. Abstraction based supervisory controller synthesis for high order monotone continuous systems. In S. Engell, G. Frehse, and E. Schnieder, editors, *Modelling, Analysis and Design of Hybrid Systems*, volume 279 of *Lecture notes in computer science*, pages 247–265. Springer-Verlag, 2002.

[11] X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. An approach to the description and analysis of hybrid systems. In *Hybrid Systems, LNCS 736, Springer Verlag*, 1993.

[12] J. Raisch and S. O'Young. Discrete approximation and supervisory control of continuous systems. *IEEE Transactions on Automatic Control*, 43(4):569–573, April 1998.

[13] P.J. Ramadge and W.M. Wonham. The control of discrete event systems. *Proc. IEEE, Special Issue on Discrete Event Dynamic Systems*, 77(1), January 1989.

[14] C. Seatzu, D. Corona, A. Giua, and A. Bemporad. Optimal control of continuous-time switched affine systems. *IEEE Transactions on Automatic Control,* conditionally accepted, 2004.

[15] J.C. Willems. Models for dynamics. *Dynamics reported*, 2, 1989.