Optimal control of hybrid automata: an application to the design of a semiactive suspension*

Daniele Corona, Alessandro Giua, Carla Seatzu

Dip. Ingegneria Elettrica ed Elettronica, Università di Cagliari,
Piazza d'Armi, 09123 Cagliari, Italy.
Email: {daniele.corona,giua,seatzu}@diee.unica.it

January 28, 2004

Abstract

We consider the optimal control of switched linear autonomous systems whose switching sequence is determined by a controlled automaton and 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 an adjacent location, with a cost associated to each switch; the time interval between two consecutive switching times is greater than a fixed quantity. We first show how a state-feedback solution can be computed off-line through a numerical procedure. Then we show how the proposed procedure can be extended to the case of an infinite number of switches.

Finally, we apply this procedure to the design of a semiactive suspension system. In particular, a hybrid model of a quarter-car semiactive suspension system has been considered, where each linear dynamics corresponds to a given value of the damping coefficient f.

1 Introduction

In this paper we focus our attention on a particular class of hybrid systems, namely *switched systems*. The main feature of switched systems is that they may switch between many operating modes, where each mode is governed by its own characteristic dynamical law [1]. In particular, we consider switched linear systems composed by autonomous dynamics and we also take into

*Submitted to *Control Engineering Practice*. Contact author is Alessandro Giua.

account some physical constraints of practical relevance. We first assume that an upper bound on the maximum number N of available switches is imposed. Then we show how, under reasonable assumptions, the proposed procedure can be extended to the case of $N = \infty$ and successfully apply this approach to the design of a semiactive suspension system.

1.1 The optimal control problem

The problem of determining optimal control laws for switched 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 [19, 23], and on the computation of optimal/suboptimal solutions by means of dynamic programming or the maximum principle [5, 12, 15, 20, 25]. Optimal control of discrete-time hybrid systems is tackled in [2].

In [9] we considered the case of switched linear systems composed of stable autonomous dynamics, with pre-assigned switching sequence (thus the only decision variables are the switching times). We proved that the optimal control law is a state-feedback and there exists a numerically viable procedure to compute the switching tables $\mathcal{C}_k$ showing the points of the state space where the next optimal switch should occur when k switches of a sequence of length N are still available. In [4] we generalized this optimization problem by taking both the switching times and the switching sequence as decision variables. The approach we proposed in [4] 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 [4] and extend the state-feedback control technique based on the construction of "switching tables" to also deal with constraints of practical relevance [3].

Constraint 1. The switching sequence is subject to logical constraints of the type: if $i_k = i$ then $i_{k+1} \in \mathrm{succ}(i)$, 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 $\mathrm{succ}(i)$. 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 complexity of the order $\mathcal{O}(r^{n-1}Ns^2)$ (if a switch may occur at no cost) or $\mathcal{O}(r^nNs^2)$ (if a cost is associated to a switch), 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.

Another important contribution of this paper is that we also study the case of an *infinite* number of switches. We show in detail that we can still compute a feedback control law using appropriate switching tables. These tables are computed with the procedure proposed for a finite value of N, provided that N is a large enough number.

1.2 The semiactive suspension design

A semiactive suspension [11, 13, 16, 21] consists of a spring and a damper but, unlike a passive suspension, the value of the damper coefficient f can be controlled and updated. In some types of suspensions, but this case is not considered here, it may also be possible to control the elastic constant λ_s of the spring.

A semiactive suspension is a valid engineering solution — when it can reasonably approximate the performance of the active control — because it requires a low power controller that can be easily realized at a lower cost than that of a fully active one [7, 14]. Note, however, that a semiactive system clearly lacks other important secondary advantages of the fully active one, namely the ability to resist downward static forces due to passenger and baggage loads and to control the altitude of the vehicle.

The optimal control technique known as LQR [18] is probably the simplest way to design an active law for suspension systems and such an idea has been initially proposed by Thompson [24]. In such a case the objective is that of minimizing a given performance index, that consists of a quadratic cost. The control input is the value $u(t)$ of the force generated by the suspension. The optimal law takes the form of a state feedback law with constant gains, i.e., $u(t) = -Kx(t)$, where $x(t)$ is the state of the system.

In this paper we design a semiactive suspension system, assuming that the damping coefficient $f(t)$ may take any value within a finite set $\mathcal{F} = \{f_1, f_2, \cdots, f_s\}$ where $f_1 < f_2 < \cdots < f_s$. The resulting model is a hybrid system where a different location corresponds to each value of f.

The control input is now the discrete switch: we appropriately change the value of f, switching from a location to another one, with the objective of minimizing a given performance index, that consists of a quadratic cost. Even in this case, the optimal law takes the form of a state feedback law: in fact in the paper it is shown that the optimal switch can be triggered by only looking at the current state $x(t)$.

As in [8] we assume that the time required to update the damping coefficient is $\delta_{\min}$. Furthermore, within this time it is only possible to pass to adjacent values of f, i.e., if $f(t) = f_i$ then $f(t + \delta_{\min}) \in \{f_{i-1}, f_i, f_{i+1}\}$.

The results of some numerical simulations show that the proposed semiactive suspension system always provides a good approximation of a fully active suspension system, while producing significant improvements with respect to purely passive suspensions.

The paper is structured as follows. In Section 2 we present the hybrid automata model that describes the class of switching systems we consider in this paper. In Section 3 we state the optimization problem we will solve. In Section 4 we show how the approach presented in [4] may be extended to take into account the presence of the new constraints. The computational complexity of the approach is presented in Section 5. In Section 6 we discuss how the proposed approach can be efficiently used in the case of an infinite number of switches. Finally, in Section 7 we show in detail how the procedure can be applied to design a semiactive suspension system.

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 [17]. 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.

— $act : L \to Diff\_Eq$ 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 $x(\tau^+) = M_{i,j} x(\tau^-)$ [1].

[1]Note that we consider a *linear* state jump. This framework is powerful enough to model several interesting cases: projection, stretching/contraction of the norm, change of coordinates and, obviously, state continuity,

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 [17] 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 $\mathrm{succ}(i) = \{j \in \mathcal{S} \; : \; (l_i, l_j) \in E\}$ 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

$$\begin{array}{ll}
V_N^* \triangleq \min\limits_{I,\mathcal{T}} & \{F(I,\mathcal{T}) \\
& \quad \triangleq \int_0^\infty x'(t) Q_{i(t)} x(t) dt + \sum_{k=1}^N h_k(\tau_k)\Big\} \\
\text{s.t.} & \dot{x}(t) = A_{i(t)} x(t), \;\; x(0) = x_0, \;\; i(0) = i_0 \\
& i(t) = i_k \text{ for } \tau_k \le t < \tau_{k+1}, \qquad k = 0, \ldots, N \\
& \tau_0 = 0, \;\; \tau_{N+1} = +\infty \\
& \tau_{k+1} \ge \tau_k + \delta_{\min}(i_k), \qquad k = 0, \ldots, N \\
& i_k \in \mathrm{succ}(i_{k-1}), \qquad k = 1, \ldots, N \\
& x(\tau_k^+) = M_{i_{k-1},i_k} x(\tau_k^-), \qquad k = 1, \ldots, N \\
& h_k(\tau_k) = H_{i_{k-1},i_k} \text{ if } \tau_k < +\infty, \\
& h_k(\tau_k) = 0 \text{ if } \tau_k = +\infty, \qquad k = 1, \ldots, N
\end{array} \tag{1}$$

The initial state x_0 and the initial location i_0 are given.

The control variables are $\mathcal{T} \triangleq \{\tau_1, \ldots, \tau_N\}$ and $I \triangleq \{i_1, \ldots, i_N\}$, where $\mathcal{T}$ is the set of switching

obtained by using $M = I$ (the identity matrix).

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 $i^*(t) = i^*_k$ for $\tau^*_k \leq t < \tau^*_{k+1}$ 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 and N is such that the location l_i may be reached from i_0 in $k \leq N$ steps.*

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

$$\begin{aligned} &\min_{I,\mathcal{T}} \left\{ \sum_{k=0}^{N} x'_k \bar{Q}_{i_k}(\delta_k) x_k + \sum_{k=1}^{N} h_k(\tau_k) \right\} \\ &\text{s.t.} \quad x_{k+1} = M_{i_k,i_{k+1}} \bar{A}_{i_k}(\delta_k) x_k,\ k = 0,\dots,N-1 \\ &\qquad x_0 = x(0),\ \ i_0 = i(0) \\ &\qquad i_k \in \text{succ}(i_{k-1}),\ k = 1,\cdots,N \\ &\qquad \delta_k \geq \delta_{\min}(i_k),\ k = 0,\cdots,N \end{aligned} \tag{2}$$

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 $A'_i Z_i + Z_i A_i = -Q_i$ [10].

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 $i \in \mathcal{S}$ when k switches are still available, it is possible to construct a table $\mathcal{C}_k^i$ that partitions the state space $\mathbb{R}^n$ into s_i regions $\mathcal{R}_j$'s, where $s_i = |\text{succ}(i)| + 1$. Whenever $i_{N-k} = i$ 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 $\mathcal{R}_j$ for a certain $j \in \text{succ}(i)$ we will switch to mode $i_{N-k+1} = j$; 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. On-line computations are burdensome, especially if a disturbance acting on the system may cause the system state to deviate from its expected value.

To prove this result, we show constructively how the tables $\mathcal{C}_k^i$ can be computed using a dynamic programming argument. We first show how the tables $\mathcal{C}_1^i$ $(i \in \mathcal{S})$ for the last switch can be determined. Then, we show by induction how the tables $\mathcal{C}_k^i$ can be computed once the tables $\mathcal{C}_{k-1}^i$ are known.

For sake of brevity, we only give in the rest of this section a short description of the procedure. The complete derivation can be found in [10, 4].

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 $\mathcal{C}_1^i$.

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

$$F_0^*(y,i) = y'\bar{Q}_i(+\infty)y. \tag{5}$$

— 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

$$T_1(y,i,\varrho,j) = y'\bar{Q}_i(\varrho)y + y'\bar{A}_i'(\varrho)M_{i,j}'\bar{Q}_j(+\infty)M_{i,j}\bar{A}_i(\varrho)y. \tag{6}$$

Note that T_1 assumes that we can switch after a time interval $\varrho = 0$, i.e., the constraint about the minimum sojourn 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.,

$$F_0^*(x, i) = \lambda^2 F_0^*(y, i). \tag{7}$$

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

$$F_1^*(x, i) = \lambda^2 \min_{j \in \{\text{succ}(i), i\}} \min_{\varrho \geq 0} \{T_1(y, i, \varrho, j)\}. \tag{8}$$

In general the argument that minimizes (8) may be not unique. To uniquely choose one optimal argument, we impose the following lexicographic ordering. Let (ϱ, j) and (ϱ', j') be two different optimal arguments of (8): we say that $(\varrho, j) \prec (\varrho', j')$ if $\varrho > \varrho'$ or, in case of $\varrho = \varrho'$, $j < j'$. We always chose the optimal argument (ϱ, j) such that $(\varrho, j) \prec (\varrho', j')$ for all other optimal arguments (ϱ', j').

If we denote $(\varrho^*(x, i), j^*(x, i))$ the optimal argument of (8), obviously, being $x = \lambda y$ it also holds that

$$(\varrho^*(x, i), j^*(x, i)) = (\varrho^*(y, i), j^*(y, i)).$$

Thus, the optimal switch from mode i to mode $j^*(y, i)$ should occur after a delay $\varrho^*(y, i)$.

We can say that a vector $x = \lambda y$ belongs to $\mathcal{R}_j$ ($j \in \text{succ}(i)$) if and only if $j = j^*(y, i)$ and $\varrho^*(y, i) = 0$, because in this case the optimal remaining cost can be obtained switching to mode j with no delay.

Finally, $\mathcal{R}_i = \mathbb{R}^n \setminus \cup_{j \in \text{succ}(i)} \mathcal{R}_j$. Since the value of $\varrho^*(x, i)$ does not depend on λ, it immediately follows that these regions are homogeneous[2], 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 $F_1^*(y, i)$ and $\varrho^*(y, i)$ 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

$$F_1^*(x, i) = \min_{j \in \{\text{succ}(i), i\}} \min_{\varrho \geq 0} \{\lambda^2 T_1(y, i, \varrho, j) + H_{i,j}\}. \tag{9}$$

The couple that minimizes the above equation is $(\varrho^*(x, i), j^*(x, i))$, uniquely determined using the previous lexicographic ordering. Thus the optimal switch should occur after a delay $\varrho^*(x, i)$.

We can say that a vector x belongs to $\mathcal{R}_j$ ($j \in \text{succ}(i)$) if and only if $j = j^*(x, i)$ and $\varrho^*(x, i) = 0$. Finally, $\mathcal{R}_i = \mathbb{R}^n \setminus \cup_{j \in \text{succ}(i)} \mathcal{R}_j$. In this case it is not sufficient to compute $F_1^*(y, i)$ and $\varrho^*(y, i)$ for all vectors y that belong to the unitary semi-sphere but we need to grid all the state space.

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

4.2 Computation of the Tables for the Intermediate Switches

We now generalize the previous approach to determine the tables $\mathcal{C}_k^i$, for $k = 2, \ldots, N$.

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

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

$$F_k^*(x, i) = \min_{j \in \{\mathrm{succ}(i), i\}} \min_{\varrho \geq 0} \{T_k(x, i, \varrho, j) + H_{i,j}\}, \tag{10}$$

where

$$T_k(x, i, \varrho, j) = x' \bar{Q}_i(\varrho) x + x_j'(\varrho) \bar{Q}_j(\delta_{min}(j)) x_j(\varrho) + F_{k-1}^*(\bar{A}_j(\delta_{min}(j)) x_j(\varrho), j) \tag{11}$$

and $x_j(\varrho) = M_{i,j} \bar{A}_i(\varrho) x$. Each member of the sum that defines $T_k(x, i, \varrho, j)$ 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 $\mathcal{C}_k^i$, as we did before: if all switching costs are null it is sufficient to sample only along the unitary semi-sphere, otherwise it is necessary to grid all the state space.

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 costs $F_N^*(\cdot, i)$ that are evaluated during the construction of the tables $\mathcal{C}_N^i$, $i \in \mathcal{S}$. We define the cost

$$F_{N+1}^*(x) = \min_{i \in \mathcal{S}} \{x' \bar{Q}_i(\delta_{min}(i)) x + F_N^*(\bar{A}_i(\delta_{min}(i)) x, i)\},$$

where the argument of the minimization 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 $\mathcal{C}_{N+1}$ 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:

$$\mathcal{R}_i = \{x \mid F_N^*(x, i) = F_{N+1}^*(x)\}$$

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.* [9, 10] is $\mathcal{O}(r^{n-1})$ if all switching costs are null, because the table contains two regions that can be determined by solving a one-parameter optimization problem for each vector y on the unitary semi-sphere. On the contrary, if not all switching costs are null the complexity is $\mathcal{O}(r^n)$ because it is necessary to grid all the state space.

For sake of simplicity, let us consider a problem formulation in which all switching costs are null. 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 complexity of computing the tables $\mathcal{C}_k^i$ 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 y on the unitary semi-sphere. Thus the complexity of solving the optimal control problem (1) for a sequence of length N is $\mathcal{O}(Nr^{n-1}\sum_{i=1}^{s}(s_i-1)) \leq \mathcal{O}(Nr^{n-1}s^2)$, because $s_i \leq s$.

Finally, if not all switching costs are null, following the same argument one can immediately show that the complexity is $\mathcal{O}(Nr^n s^2)$. In any 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 $\mathcal{O}(Nr^{n-1}s^N)$ or $\mathcal{O}(Nr^n s^N)$ depending on the presence of switching costs.

6 An infinite number of switches

In this section we discuss how, under appropriate assumptions, the above procedure can be efficiently extended to the case of $N = \infty$. In particular, we consider an optimal control problem of the form (1) where

(i) for all $i \in \mathcal{S}$, the linear dynamics A_i is stable;

(ii) no cost is associated to switches, i.e., $H_{i,j} = 0$ for all $i, j \in \mathcal{S}$;

(iii) the state x is continuous, i.e., $M_{i,j} = I_n$ for all $i, j \in \mathcal{S}$, where I_n denotes the n-th order identity matrix.

(iv) for all $i \in \mathcal{S}$, $Q_i > 0$.

Let us preliminary state an obvious monotonicity result.

Property 1 Let $N, N' \in \mathbb{N}$. If $N < N'$ and the switched system evolves along an optimal trajectory, then for any initial state (x_0, i_0),

$$+\infty > V_N^*(x_0, i_0) \geq V_{N'}^*(x_0, i_0).$$

Proof: We first observe that by assumption (i) $V_N^*(x_0, i_0)$ is finite for any $N \geq 1$. Now, we prove the second inequality by contradiction. Assume that $V_{N'}^*(x_0, i_0) > V_N^*(x_0, i_0)$. Then it is obvious that the same evolution that generates $V_N^*(x_0, i_0)$ is also admissible for (1) when a larger value N' of switches is allowed. This leads to a contradiction. □

Proposition 1 For any initial state (x_0, i_0), $x_0 \neq 0$, and $\forall\ \varepsilon' > 0$, $\exists\ \bar{N} = \bar{N}(x_0, i_0)$ such that for all $N > \bar{N}$, $V_N^*(x_0, i_0) - V_{\bar{N}}^*(x_0, i_0) < \varepsilon'$.

Proof: We first observe that by assumptions (iii) and (iv) $V_N^*(x_0, i_0)$ is lower bounded by a strictly positive number. Then, the result trivially follows from the monotonicity property above and the fact that V_N^* is lower bounded. □

Proposition 2 For any initial state (x_0, i_0), $x_0 \neq 0$, and $\forall\ \varepsilon > 0$, $\exists\ \bar{N}$ such that for all $N > \bar{N}$,

$$\frac{V_N^*(x_0, i_0) - V_{\bar{N}}^*(x_0, i_0)}{V_N^*(x_0, i_0)} < \varepsilon.$$

Proof: Under the assumption (ii) the optimal costs are quadratic functions of x_0, i.e., if $x_0 = \lambda y_0$, then $V_N^*(\lambda y_0, i_0) = \lambda^2 V_N^*(y_0, i_0)$ and $V_{\bar{N}}^*(\lambda y_0, i_0) = \lambda^2 V_{\bar{N}}^*(y_0, i_0)$. Moreover, by Proposition 1 $\forall\ (y_0, i_0)$ and $\forall\ \varepsilon' > 0$, $\exists\ \bar{N}(y_0, i_0)$ such that $\forall\ N > \bar{N}(y_0, i_0)$, $V_N^*(y_0, i_0) - V_{\bar{N}}^*(y_0, i_0) < \varepsilon'$. Hence if we define

$$\bar{N} = \max_{i_0 \in \mathcal{S},\ y_0\ :\ ||y_0||=1} \bar{N}(y_0, i_0)$$

it holds that

$$\frac{V_N^*(x_0, i_0) - V_{\bar{N}}^*(x_0, i_0)}{V_N^*(x_0, i_0)} = \frac{\lambda^2 [V_N^*(y_0, i_0) - V_{\bar{N}}^*(y_0, i_0)]}{\lambda^2 V_N^*(y_0, i_0)} \leq \frac{\varepsilon'}{\min\limits_{y_0\ :||y_0||=1} V_N^*(y_0, i_0)} = \varepsilon.$$

□

According to the above result, one may use a given relative tolerance ε to approximate two cost values, i.e.,

$$\frac{V_N^*(x, i) - V_{N'}^*(x, i)}{V_N^*(x, i)} < \varepsilon \qquad \Longrightarrow \qquad V_N^*(x, i) \cong V_{N'}^*(x, i).$$

Hence, we can now prove the main result of this section.

Theorem 1 Given a fixed relative tolerance ε, if $\bar{N}$ is chosen as in Proposition 2 then for all $N > \bar{N}+1$ it holds that $\mathcal{C}_N^i = \mathcal{C}_{\bar{N}+1}^i$.

Proof: By definition $V_k^*(x_0, i_0) = F_k^*(x_0, i_0)$ for all $k \geq 1$, hence from equations (10) and (11) it follows that

$$V_N^*(x_0, i_0) = \min_{j\in\{\mathrm{succ}(i_0), i_0\}} \min_{\varrho\geq 0} \left\{ x_0'\bar{Q}_{i_0}(\varrho)x_0 + x'(\varrho)\bar{Q}_j(\delta_{min}(j))x(\varrho) + V_{N-1}^*(\bar{A}_j(\delta_{min}(j))x(\varrho), j) \right\}$$

where $x(\varrho) = \bar{A}_{i_0}(\varrho)x_0$. Now, being by assumption $N-1 > \bar{N}$, by virtue of Proposition 2 we may approximate

$$V_{N-1}^*(\bar{A}_j(\delta_{min}(j))x(\varrho), j) \cong V_{\bar{N}}^*(\bar{A}_j(\delta_{min}(j))x(\varrho), j)$$

thus

$$\begin{aligned} V_N^*(x_0, i_0) &\cong \min_{j\in\{\mathrm{succ}(i_0), i_0\}} \min_{\varrho\geq 0} \left\{ x_0'\bar{Q}_{i_0}(\varrho)x_0 + x'(\varrho)\bar{Q}_j(\delta_{min}(j))x(\varrho) + V_{\bar{N}}^*(\bar{A}_j(\delta_{min}(j))x(\varrho), j) \right\} \\ &= V_{\bar{N}+1}^*(x_0, i_0). \end{aligned}$$

Therefore, the optimal arguments (ϱ^*, j^*) used to compute $\mathcal{C}_N^i$ and $\mathcal{C}_{\bar{N}+1}^i$ are the same. □

The above result allows one to compute with a finite procedure the optimal tables for a switching law when N goes to infinity. In such a case, in fact, it holds that

$$\mathcal{C}_\infty^i = \lim_{N\to\infty} \mathcal{C}_N^i = \mathcal{C}_{\bar{N}+1}^i.$$

Hence, we only need to use the tables $\mathcal{C}_\infty^i$, $i \in \mathcal{S}$ for all switches.

To construct the tables $\mathcal{C}_\infty^i$ the value of $\bar{N}$ is needed. We do not provide so far any analytical way to compute $\bar{N}$, therefore our approach consists in constructing tables until a convergence criterion is met.

Finally, we recall that under the assumptions (i) to (iv), the system, optimally controlled with an infinite number of switches, is stable as proved in [10].

7 Semiactive suspension design

In this section we show how the proposed methodology can be successfully applied to the design of a semiactive suspension system.

7.1 Dynamical models of the suspension system

We consider a quarter car suspension system and derive two different dynamical models. The first one is a two-degrees of freedom fourth order dynamical model that takes into account the dynamics of the tire. The second one is a one-degree of freedom second order dynamical model that neglects the effect of the tire.

While the second order model allows one to study the filtering properties of the suspension in terms of passenger comfort, it does not describe the interaction of the tire with the suspended mass and the ground, and thus it cannot be used to evaluate other important features such as road holding and rideability.

From a tutorial point of view, however, the reduced order model is extremely useful, because it is possible to give a geometrical representation of the optimal switching regions, thus providing a more intuitive explanation of the proposed approach. This is the main reason that led us to consider both models.

7.1.1 The fourth order dynamical model

Let us now consider the completely active suspension system of a quarter car with two degrees of freedom schematized in Figure 1.a. We used the following notation:

- M_w is the equivalent unsprung mass consisting of the wheel and its moving parts;
- M_s is the sprung mass, i.e., the part of the whole body mass and the load mass pertaining to only one wheel;
- λ_t is the elastic constant of the tire, whose damping characteristics have been neglected. Note that this is in line with almost all researchers who have investigated synthesis of active suspensions for motor vehicles as the tire damping is minimal;
- $x_1(t)$ is the deformation of the suspension with respect to (wrt) the static equilibrium configuration, taken as positive when elongating;
- $x_2(t)$ is the vertical absolute velocity of the sprung mass M_s;
- $x_3(t)$ is the deformation of the tire wrt the static equilibrium configuration, taken as positive when elongating;
- $x_4(t)$ is the vertical absolute velocity of the unsprung mass M_w;
- $u(t)$ is the control force produced by the actuator.

It is readily shown that the state variable mathematical model of the system under study is given by [7]

$$\dot{x}(t) = \overline{A}x(t) + \overline{B}u(t) \tag{12}$$

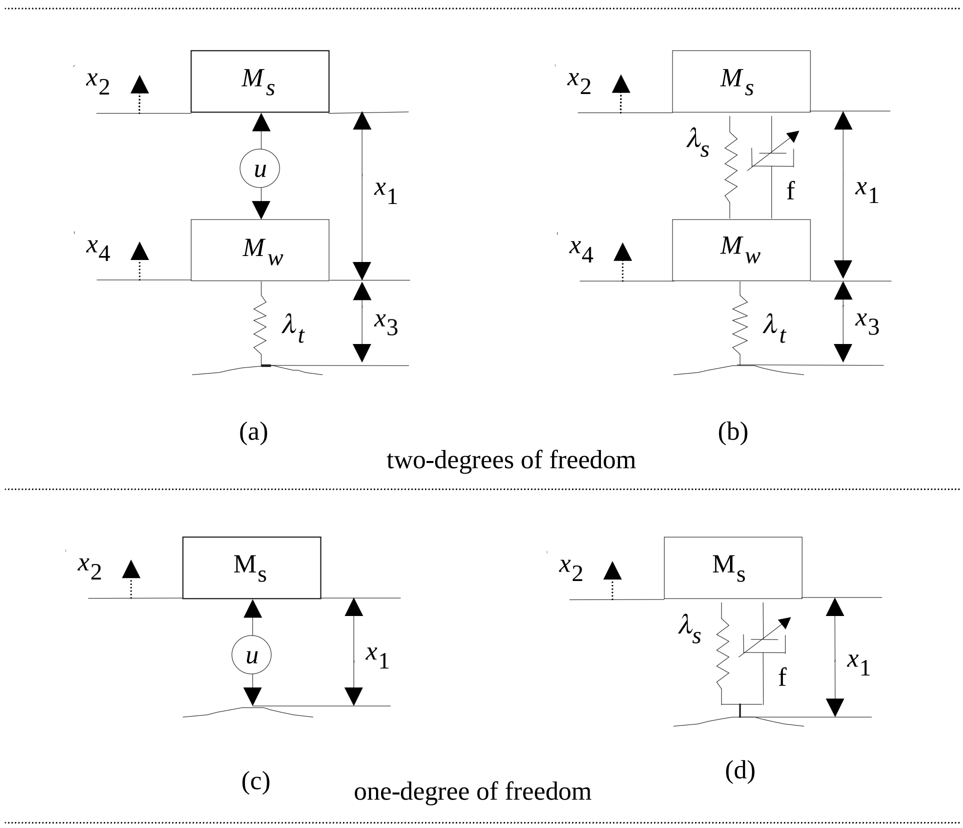

Figure 1: *Scheme of the two degrees-of-freedom suspension: (a) active suspension; (b) semiactive suspension. Scheme of the one degree-of-freedom suspension: (c) active suspension; (d) semiactive suspension.*

where $x(t) = [x_1(t),\ x_2(t),\ x_3(t),\ x_4(t)]^T$ is the state, and the constant matrices $\overline{A}$ and $\overline{B}$ have the following structure:

$$\overline{A} = \begin{bmatrix} 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\lambda_t/M_w & 0 \end{bmatrix}, \qquad \overline{B} = \begin{bmatrix} 0 \\ 1/M_s \\ 0 \\ -1/M_w \end{bmatrix}.$$

Now, let us consider Figure 1.b that represents a conventional semiactive suspension composed of a spring and a damper with adaptive characteristic coefficient $f = f(t)$.

The effect of this suspension is equivalent to that of a control force

$$u_s(t) = -\begin{bmatrix} \lambda_s & f(t) & 0 & -f(t) \end{bmatrix} x(t). \tag{13}$$

Note that, as f may vary, $u_s(t)$ is both a function of $f(t)$ and of $x(t)$. It is immediate to verify that the state variable mathematical model of the semiactive suspension is still given by equation (12) where $u(t)$ is replaced by $u_s(t)$. Therefore, in such a case the system dynamics is regulated by the following state equation:

$$\dot{x}(t) = Ax(t) = \begin{bmatrix} 0 & 1 & 0 & -1 \\ -\lambda_s/M_s & -f(t)/M_s & 0 & f(t)/M_s \\ 0 & 0 & 0 & 1 \\ \lambda_s/M_w & f(t)/M_w & -\lambda_t/M_w & -f(t)/M_w \end{bmatrix} x(t). \tag{14}$$

7.1.2 The second order dynamical model

If the dynamics of the tire is completely neglected, the suspension system of a quarter car can be schematized as shown in Figures 1.c and d. More precisely, figure c provides the scheme of a completely active suspension system, while figure d provides the scheme of a semiactive suspension system, where the physical meaning of all variables is the same as in the two-degrees of freedom case.

The state variable mathematical model of the active system is still given by a linear equation of the form (12), where the state is $x(t) = [x_1(t),\ x_2(t)]^T$, and the constant matrices $\overline{A}$ and $\overline{B}$ have the following structure:

$$\overline{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad \overline{B} = \begin{bmatrix} 0 \\ 1/M_s \end{bmatrix}.$$

The effect of the semiactive suspension is equivalent to that of a control force

$$u_s(t) = -\begin{bmatrix} \lambda_s & f(t) \end{bmatrix} x(t). \tag{15}$$

Thus, the system dynamics of a semiactive suspension is regulated by the following state equation:

$$\dot{x}(t) = Ax(t) = \begin{bmatrix} 0 & 1 \\ -\lambda_s/M_s & -f(t)/M_s \end{bmatrix} x(t). \tag{16}$$

7.2 Semiactive suspension design

Now, let us discuss in detail how the proposed methodology can be successfully used to design a semiactive suspension system.

As already said in the Introduction, we assume that the value of the damping coefficient f may take values within a finite set $\mathcal{F} = \{f_1, f_2, \cdots, f_s\}$ where $f_1 < f_2 < \cdots, f_s$. We select the value of f in $\mathcal{F}$ so as to minimize a given performance index, consisting of a quadratic cost depending on the time evolution.

Moreover, we assume that:

(A1) the state is measurable;

(A2) whenever f is updated, its value remains the same within a given time interval $\delta_{\min}$, that does not depend on the current value of f;

(A3) if at time t the damping coefficient is updated to $f(t) = f_i \in \mathcal{F}$, and at least one more switch is available, then at time $t + \delta_{\min}$ the value of f may either remain the same or it may switch to an "adjacent" value, namely,

$$f(t+\delta_{\min}) \in \begin{cases} \{f_i, f_{i+1}\} & i = 1 \\ \{f_{i-1}, f_i, f_{i+1}\} & i = 2, \cdots, s-1 \\ \{f_{i-1}, f_i\} & i = s \end{cases} \tag{17}$$

Note that in a first approximation, assumption (A2) enables us to take into account the fact that the damping coefficient f cannot be updated at an arbitrarily high frequency. Clearly, the amplitude of the time interval $\delta_{\min}$ depends on the particular physical damper. As an example, in the case of a solenoid valve damper [8, 22], under the above assumption (A2) an admissible value is $\delta_{\min} = 7$ ms [8]. If the assumption (A3) is removed, and we assume that the value of f may arbitrarily change from any value to any other one, a larger $\delta_{\min}$ should be considered, e.g., $\delta_{\min} = 30$ ms [8].

Under the assumptions (A1) to (A3), the considered optimal control problem can be written as:

$$\begin{array}{lll} V_N^* \triangleq & \min\limits_{I,\mathcal{T}} \left\{ F(I,\mathcal{T}) \triangleq \displaystyle\int_0^\infty x'(t) Q_{i(t)} x(t) dt \right\} & \\ & & \\ \text{s.t.} & \dot{x}(t) = A_{i(t)} x(t), \;\; x(0) = x_0, \;\; i(0) = i_0 & \\ & i(t) = i_k \text{ for } \tau_k \le t < \tau_{k+1}, & k = 0, \ldots, N \\ & \tau_0 = 0, \;\; \tau_{N+1} = +\infty & \\ & \tau_{k+1} \ge \tau_k + \delta_{\min}, & k = 0, \ldots, N \\ & i_k \in \text{succ}(i_{k-1}), & k = 1, \ldots, N \\ & x(\tau_k^+) = x(\tau_k^-), & k = 1, \ldots, N \end{array} \tag{18}$$

where matrices $A_{i(t)}$ are uniquely defined given the value of f according to equations (14) or (16), depending on the considered dynamical model. More precisely, to each value of f in $\mathcal{F}$ it corresponds a matrix $A(f(t))$ that specifies the discrete state (location) of the hybrid system.

Note that the optimal control problem (18) is a particular case of (1) where no cost is associated to switches, the state x is continuous, and the minimum permanence time in discrete locations is the same for all locations.

Moreover, given the assumption (A3), the automaton showing all the allowed switches takes the structure of a birth-death process and is shown in Figure 2.

In the following we present the results of some numerical simulations carried out on both the second order and the fourth order dynamical system. In particular, we first assume that a finite number N of switches is available, then we remove such an assumption, thus allowing the system to perform an infinite number of switches.

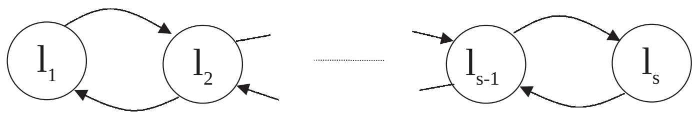

Figure 2: *The hybrid automaton that defines the mode switching.*

7.3 Application example

The proposed procedure has been applied to the quarter car suspension shown in Figure 1, with values of the parameters taken from [11], namely, $M_w = 28.58$ Kg, $M_s = 288.90$ Kg, $\lambda_s = 14345$ N/m, and $\lambda_t = 155900$ N/m.

We assume that the damping coefficient f may take values within the finite set $\mathcal{F}[\text{ Ns/m}] = \{800, 1500, 2300, 3000\}$, while the minimum permanence time is equal to $\delta_{\min} = 7$ ms.

7.4 Simulations on the second order model

We first present the results of some numerical simulations carried out on the second order dynamical model of the suspension system.

A different weighting matrix is associated to each discrete location, or equivalently to each value of f. In particular, we assume that

$$Q_{i(t)} = Q(f(t)) = \text{diag}\{1,\ 0\} + 0.8 \cdot 10^{-9} \cdot [\lambda_s\ \ f(t)]^T \cdot [\lambda_s\ \ f(t)].$$

In such a way, by virtue of equation (15), we can perform a significant comparison, in terms of performance index, among the proposed semiactive suspension and an active suspension system, considered as a target, and obtained by solving an LQR problem where $Q = \text{diag}\{1,\ 0\}$ and $R = 0.8 \cdot 10^{-9}$. Note that the numerical values of the weighting matrices Q and R are the same as those already considered in [11].

Simulation 1: $N = 6$.

We first assume that a finite number $N = 6$ of switches is available. We evaluate off-line the $N \times s$ switching tables. A state space discretization of $r = 100$ points along the unitary semisphere and a minimum local search over three time constants were considered sufficiently fine.

We assume that the initial state is $x_0 = [0.1\ 0]^T$ and $i_0 = 1$.

The state trajectory that minimizes the performance index is depicted in Figure 3, where the circle indicates the initial state and the squares indicate the values of the state at the switching times. We found out $\mathcal{T}^*(\text{s}) = \{0.096, 0.1370, 0.222, 0.473, 0.482, 0.646\}$, $\mathcal{I}^* = \{1, 2, 3, 4, 3, 2, 3\}$, and $J^* = 1.419 \cdot 10^{-3}$.

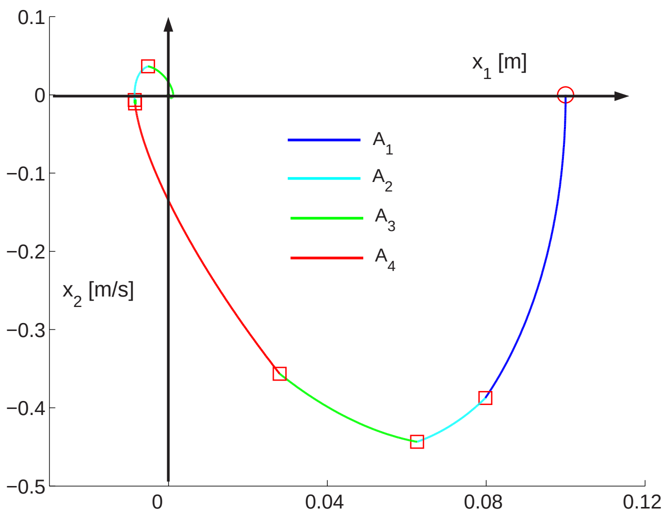

Figure 3: *The results of Simulation 1: the state trajectory.*

Figure 4 shows, among the 24 tables constructed, only the 6 ones used by the controller during the evolution of the system. The system initially evolves in location l_1. When the minimum permanence time $\delta_{\min}$ has elapsed, the controller must keep checking the color in table $\mathcal{C}_6^1$ (see Figure 4) corresponding to the current state x. According to this color the controller decides whether to remain in l_1 or to switch to the adjacent location l_2. In this case, no switch occurs until a time $\tau_1 = 0.096$ s has elapsed, when the continuous state reaches the cyan area relative to location l_2. Now the controller will wait for the minimum permanence time and then consider table $\mathcal{C}_5^2$. The same procedure is repeated until all the available switches are performed.

Note that, given the structure of the automaton, while the switching tables associated to discrete locations l_2 and l_3 may have up to 3 colors, the tables associated to locations l_1 and l_4 may have at most two different colors.

To better appreciate the performance of the proposed semiactive suspension it is necessary to look at the time evolution of the sprung mass displacement. This curve is reported in Figure 5.a where we can also visualize the evolution of the fully active suspension considered as a target, and that of a completely passive suspension obtained using a value of $f = 1918$ Ns/m [6].

In Figure 5.b we have reported the different values of the damping coefficient f during the simulation.

In Table 1 we compare the values of the quadratic performance index obtained using the active suspension (considered as a target), the semiactive suspension in the case of $N = 6$ ($i_0 = 1$ in all cases), and the passive suspension system obtained using $f = 1918$ Ns/m.

The results of Table 1, together with the results of other numerical simulations that have not been reported here for sake of brevity, enable us to conclude that the proposed semiactive

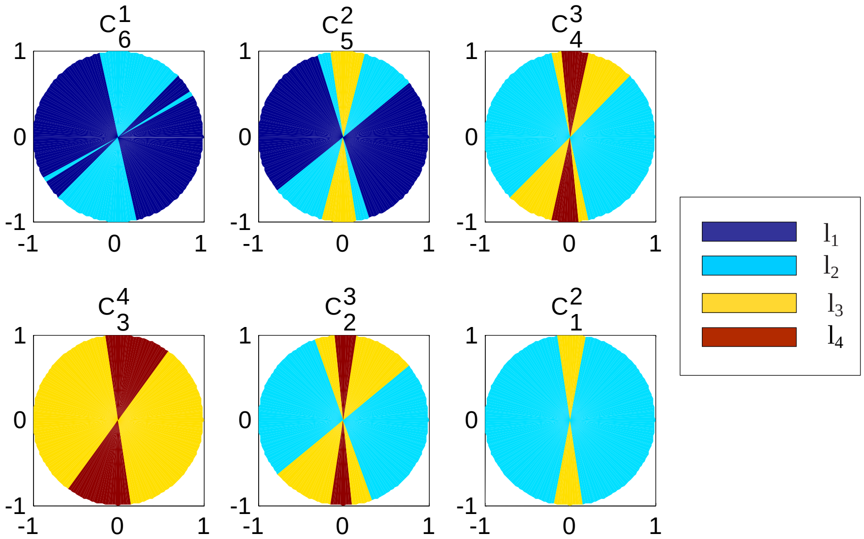

Figure 4: *Tables used by the controller to compute the state evolution in Figure 3.*

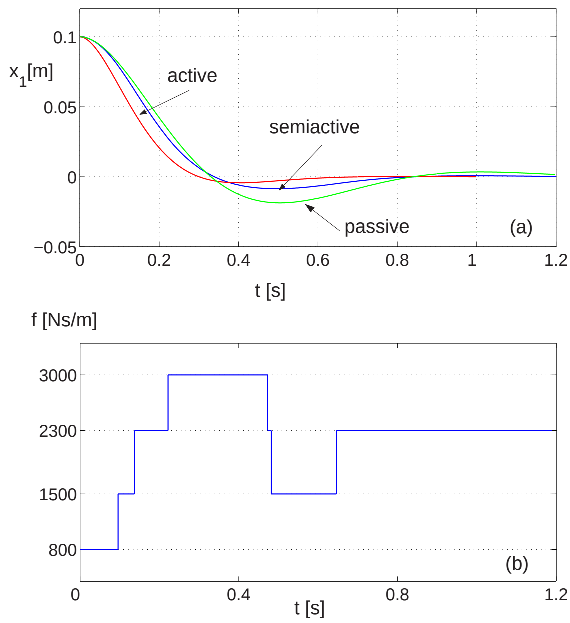

Figure 5: *The results of Simulation 1: (a) the time evolution of the sprung mass displacement; (b) the different values of f used by the semiactive suspension.*

x_0	semiactive ($N = 6$)	semiactive ($N = \infty$)	active	passive
$[0.100\ \ 0.000]^T$	$1.419 \cdot 10^{-3}$	$1.419 \cdot 10^{-3}$	$1.278 \cdot 10^{-3}$	$1.546 \cdot 10^{-3}$
$[0.045\ \ 0.090]^T$	$3.960 \cdot 10^{-4}$	$3.959 \cdot 10^{-4}$	$3.294 \cdot 10^{-4}$	$4.189 \cdot 10^{-4}$
$[-0.015\ \ 0.100]^T$	$1.493 \cdot 10^{-5}$	$1.492 \cdot 10^{-5}$	$1.437 \cdot 10^{-5}$	$1.905 \cdot 10^{-5}$
$[-0.057\ \ 0.080]^T$	$3.719 \cdot 10^{-4}$	$3.717 \cdot 10^{-4}$	$3.506 \cdot 10^{-4}$	$4.114 \cdot 10^{-4}$

Table 1: *Different values of the performance index in the case of some numerical simulations carried out on the second order model.*

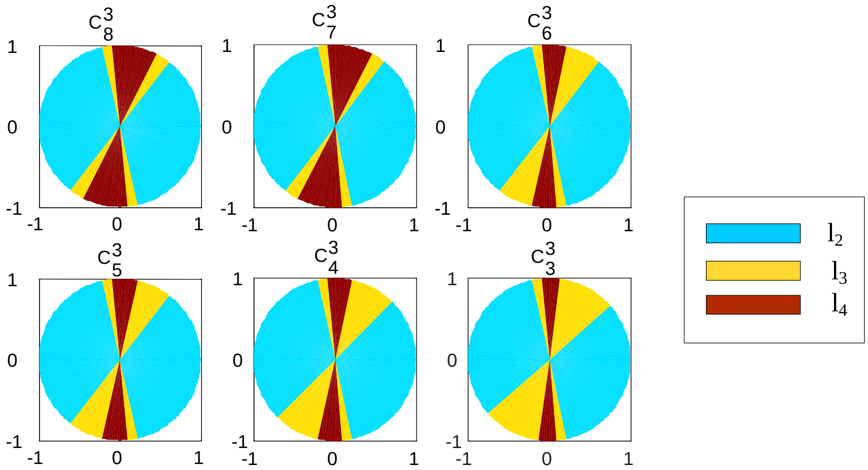

Figure 6: *The first 6 switching tables for location l_3 and $N = 8$.*

suspension exhibits a behaviour that is intermediate between that of the passive suspension and the considered active one, even if a small number of switches is allowed.

Simulation 2: $N = \infty$.

As already discussed in Section 6, for a sufficiently large value of N, the tables relative to the first switches always converge to the same one, only depending on the discrete location $l \in L$.

As an example, assume $N = 8$ and consider the discrete location l_3. The tables relative to the first 6 switches, namely $\mathcal{C}_k^3$, $k = 3, \cdots, 8$, are reported in Figure 6. We may observe that, as the number of available switches increases, i.e., k goes from 3 to 8, the tables converge. In particular, in this case the tables relative to the first two switches, namely $\mathcal{C}_8^3$ and $\mathcal{C}_7^3$, are the same. Now, if we consider a larger value of N, e.g. $N = 9$ (10), and look at the tables relative to location l_3, we may observe that $\mathcal{C}_9^3$ ($\mathcal{C}_{10}^3$) tables coincide with $\mathcal{C}_8^3$ and $\mathcal{C}_7^3$. Using the notation introduced in Section 6, we denote these tables as $\mathcal{C}_\infty^3$.

Analogous considerations may be repeated for all the other discrete locations.

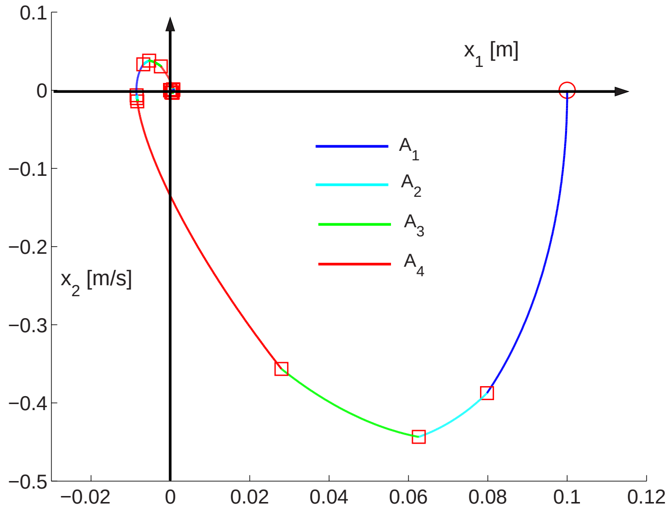

Figure 7: *The results of Simulation 2: the state trajectory.*

Now, let us consider the optimal control problem (18) with no bound on the maximum number of available switches.

By virtue of the above convergence properties, this problem can be solved by using only the tables $\mathcal{C}^i_\infty$, for $i \in \mathcal{S}$, as described in Section 6. These tables are not reported here for sake of brevity.

Assume that the initial state is still equal to $x_0 = [0.1\ 0]^T$ and $i_0 = 1$.

The state trajectory that minimizes the performance index is reported in Figure 7 where the circle indicates the initial state and the squares indicate the values of the state at the switching times. It can be easily observed that this trajectory is practically coincident with that in Figure 3. This clearly occurs because after the first 6 switches, the system has practically reached the origin. As a consequence, the optimal value of the performance index J^* is practically the same, as it can be read in Table 1.

In Figure 8.a we have reported the sprung mass displacement of the semiactive suspension together with that of the fully active suspension considered as a target, and that of a completely passive suspension [6]. In Figure 8.b we can see the different values of the damping coefficient f during the numerical simulation. Note that the periodicity of the switching sequence is a consequence of the particular example (second order system, rotating dynamics), but it is not a general result.

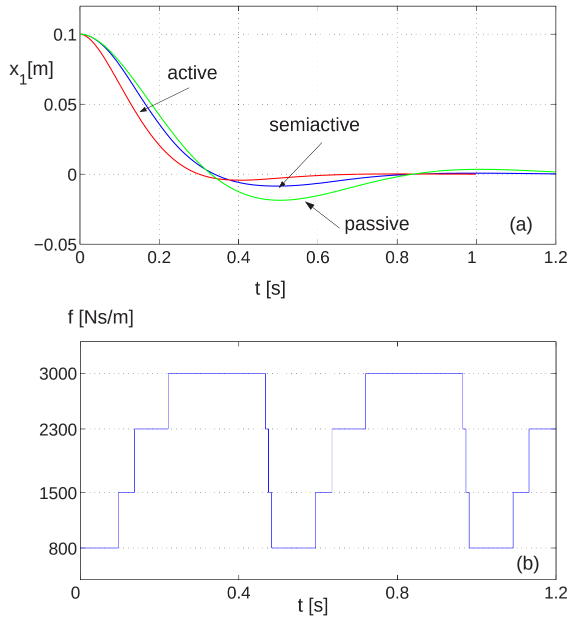

Figure 8: *The results of Simulation 2: (a) the time evolution of the sprung mass displacement; (b) the different values of f used by the semiactive suspension.*

7.5 Simulations on the fourth order model

Now, let us present the results of some numerical simulations carried out on the fourth order dynamical model of the suspension system.

As in the previous case, a different weighting matrix is associated to each discrete location, or equivalently to each value of f. In particular, we assume that

$$Q_{i(t)} = Q(f(t)) = \mathrm{diag}\{1,\ 0,\ 10,\ 0\} + 0.8 \cdot 10^{-9} \cdot [\lambda_s \ \ f(t) \ \ 0 \ \ -f(t)]^T \cdot [\lambda_s \ \ f(t) \ \ 0 \ \ -f(t)].$$

In such a way, by virtue of equation (13), we can perform a significant comparison, in terms of performance index, among the proposed semiactive suspension and an active suspension system, considered as a target, and obtained by solving an LQR problem where $Q = \mathrm{diag}\{1,\ 0,\ 10,\ 0\}$ and $R = 0.8 \cdot 10^{-9}$ [11].

Now, let us consider the most realistic case of $N = \infty$.

As already explained above, we first compute the $N \times s$ switching tables for a "sufficiently" large value of N until we observe that there exists a $k < N$ such that for all $i \in \mathcal{S}$, $\mathcal{C}_k^i = \mathcal{C}_{k+1}^i = \cdots = \mathcal{C}_N^i$. In this case we took $N = 6$ and we observed that the convergence occurs for $k = 5$. Thus, we can reasonably assume $\mathcal{C}_\infty^i = \mathcal{C}_6^i$, $i = 1, \cdots, 4$.

These switching tables are not reported here because a significant graphical representation is not possible[3].

[3]See the appendix for a brief description of the algorithm that allowed the numerical construction of the tables

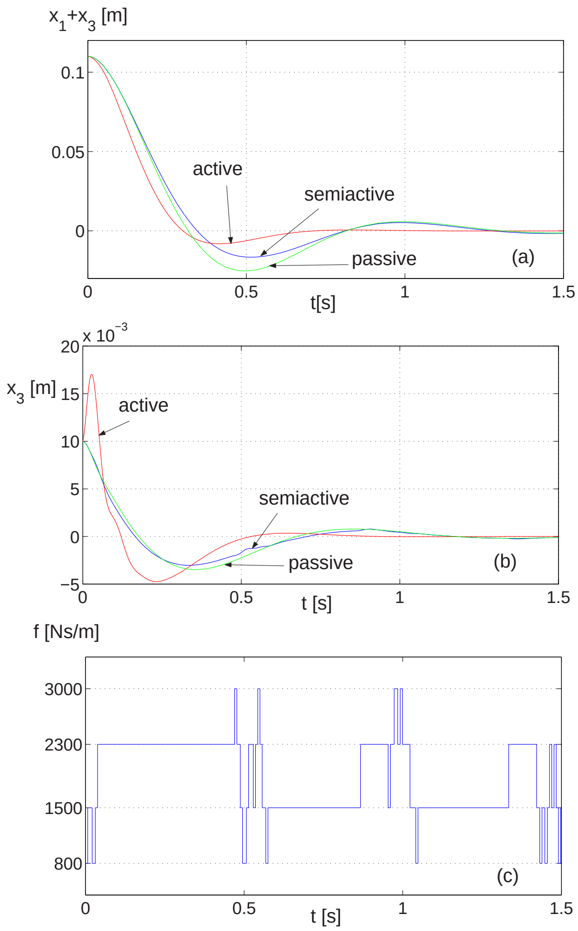

Figure 9: *The results of the simulation carried out on the fourth order model: (a) the time evolution of the sprung mass displacement ($x_1 + x_3$); (b) the unsprung mass displacement x_3; (c) the different values of f used by the semiactive suspension.*

Assume that the initial state is $x_0 = [0.1 \;\; 0 \;\; 0.01 \;\; 0]^T$ and $i_0 = 1$.

In Figures 9.a and b we have reported the sprung mass and the unsprung mass displacement of the semiactive suspension together with that of the fully active suspension considered as a target, and that of a completely passive suspension [6]. In particular, by looking at plot (a) that shows the most significant variable, we can conclude that the semiactive system guarantees better performance than the passive one. In fact, in such a case, the behaviour of the semiactive suspension system in terms of the sprung mass displacement, is quite similar to that obtained using the purely active system. Finally, in Figure 9.c we can see the different values of the damping coefficient f during the numerical simulation.

A comparison among the semiactive, the active and the passive suspension in terms of performance index is given in Table 2. Note that in this table we have also reported the results of other numerical simulations carried out for different values of the initial state x_0. We may conclude

in $\mathbb{R}^4$.

x_0	semiactive	active	passive
$x_0 = [0.100\ \ 0\ \ 0.010\ \ 0]^T$	$1.775 \cdot 10^{-3}$	$1.591 \cdot 10^{-3}$	$1.829 \cdot 10^{-3}$
$x_0 = [-0.050\ \ 0.300\ \ -0.005\ \ 0.010]^T$	$2.423 \cdot 10^{-4}$	$2.374 \cdot 10^{-4}$	$2.976 \cdot 10^{-4}$
$x_0 = [0.050\ \ 0.300\ \ 0.005\ \ 0.010]^T$	$1.011 \cdot 10^{-3}$	$8.200 \cdot 10^{-4}$	$1.052 \cdot 10^{-3}$
$x_0 = [0.010\ \ -0.300\ \ 0.010\ \ 0.100]^T$	$1.678 \cdot 10^{-4}$	$1.164 \cdot 10^{-4}$	$2.175 \cdot 10^{-4}$
$x_0 = [0\ \ 0.400\ \ 0.010\ \ 0.300]^T$	$3.513 \cdot 10^{-4}$	$3.109 \cdot 10^{-4}$	$4.312 \cdot 10^{-4}$
$x_0 = [-0.080\ \ -0.100\ \ 0.012\ \ 0.400]^T$	$1.144 \cdot 10^{-3}$	$8.903 \cdot 10^{-4}$	$1.151 \cdot 10^{-3}$

Table 2: *The results of the numerical simulations carried out on the fourth order model.*

that the proposed semiactive suspension provides an intermediate performance between that of the passive suspension and that of the purely active one.

8 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 times is bounded from below. For this class we have shown that it is possible to extend the results presented in [4] based on the construction of "switching tables" to solve an optimal control problem with a state-feedback. We have also shown how the same approach can be used to deal with the case of an infinite number of available switches.

The proposed approach has been applied to design a semiactive suspension system for road vehicles. A hybrid model of the quarter-car semiactive suspension system has been considered, where each linear dynamics corresponds to a given value of the damping coefficient f. The results of some numerical simulations are presented and the comparison with both passive and active suspensions is also shown.

Appendix

In this appendix we will show some basic steps that allowed us to apply the procedure of the tables construction in $\mathbb{R}^4$, used to carry out the simulations described in section 7.5.

The main computational effort is the discretization of the state space. However the structure of the problem, i.e., quadratic performance index with infinite time horizon and null switching costs, lead us to switch conveniently to polar coordinates. Moreover the computation is carried out only on the unitary "hyper" semi-sphere, thus reducing to a third order discretization.

Let us first construct the relation between polar and cartesian system in $\mathbb{R}^n$. The n polar coordinates are composed of 1 radius ρ and $n-1$ angles. Given a point $x = [x_1, x_2, \cdots, x_n]$, clearly $\rho_n = \|x\|$. Indicate with θ_n the angle formed by vector x and the "hyper" plane $x_n = 0$; assign $x_n = \rho_n \sin(\theta_n)$. Now consider the equation of the "hyper" sphere $\sum_{i=1}^{n} \|x_i\|^2 = \rho_n^2$, that, according to the previous assignments, becomes:

$$\sum_{i=1}^{n-1} \|x_i\|^2 = \rho_n^2(1 - \sin^2(\theta_n)) = \rho_{n-1}^2 \tag{19}$$

where $\rho_{n-1} = \rho_n \cos(\theta_n)$. We now proceed in the same manner considering the "hyper" sphere of equation (19), in $\mathbb{R}^{n-1}$. This can be repeated until the space is reduced to $\mathbb{R}^2$. We obtain the set:

$$\begin{cases} x_n = \rho_n \sin(\theta_n) \\ x_{n-1} = \rho_{n-1} \sin(\theta_{n-1}) \\ \vdots \\ x_3 = \rho_3 \sin(\theta_3) \\ x_2 = \rho_2 \sin(\theta_2) \\ x_1 = \rho_2 \cos(\theta_2) \end{cases}$$

where $\rho_i = \rho_{i+1} \cos(\theta_i)$ for $i = n-1 \cdots 2$. To describe $\mathbb{R}^n$, variables must range in: $\rho_n \in [0, +\infty)$, $\theta_3, \cdots \theta_n \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and $\theta_2 \in [0, 2\pi)$.

In our particular problem $n = 4$ and to describe the unitary "hyper" semi-sphere we could restrict $\rho_4 = 1$ and $\theta_4 \in [0, \frac{\pi}{2}]$, by virtue of the properties mentioned above. To get rid of indexes let us call $\theta_4 = \xi$, $\theta_3 = \varphi$ and $\theta_2 = \vartheta$.

Note that a uniform discretization for each angle brings to areas with high density of points (think of the grid on the earth surface at the poles) but we needed an equally spaced grid. We gave the following criteria, namely of constant arc length:

1. define nominal values of discretization N_ϑ, N_φ, N_ξ; since $\vartheta \in [0, 2\pi)$, $\varphi \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and $\xi \in [0, \frac{\pi}{2}]$ we choose $N_\vartheta = 2N_\varphi = 4N_\xi$ proportional to the respective range of each variable;

2. discretize ξ uniformly, i.e., $\xi_i = i\frac{\pi}{2N_\xi}$, $i = 0, \cdots, N_\xi$;

3. denoted by round$(\cdot)$ a function that approximates to the closest integer, for every ξ_i define $\bar{N}_\varphi = \text{round}(N_\varphi \cos(\xi_i))$ and discretize φ uniformly, i.e., $\varphi_j = -\frac{\pi}{2} + j\frac{\pi}{\bar{N}_\varphi}$, $j = 0, \cdots, \bar{N}_\varphi - 1$;

4. for every ξ_i and φ_j define $\bar{N}_\vartheta = \text{round}(N_\vartheta \cos(\xi_i)\cos(\varphi_j))$ and discretize ϑ uniformly, i.e., $\vartheta_k = k\frac{2\pi}{\bar{N}_\vartheta}$, $k = 0, \cdots, \bar{N}_\vartheta - 1$;

With such criteria we obtained a grid of $N \simeq \frac{N_\xi N_\varphi N_\vartheta}{3}$. The algorithm of the tables construction is based on the calculation of a cost function J in each point of this grid. To contain the level of discretization and to guarantee a significant accuracy on J, an interpolation criteria is required. When a point x isn't in the grid, we use the cost values in the $H \leq 8$ points of the grid around x, namely $x_1, \cdots, x_H$. Thus, defined $d_i = \|x - x_i\|^{-1},\ i = 1 \cdots H$ we approximate $J(x) = \dfrac{\sum_1^H d_i J(x_i)}{\sum_1^H d_i}$.

The trade-off value $N_\xi = 15$ was chosen, giving a discretization of 8581 points. Running in MATLAB environment on a pentium III 450 MHz the computational time per region is about 60 hours. Note that these burdensome calculations are performed off-line.

References

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

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

[3] A. Bemporad, D. Corona, A. Giua, and C. Seatzu. Optimal state-feedback quadratic regulation of linear hybrid automata. In *Conf. on Analysis and Design of Hybrid Systems*, pages 407–412, St. Malò, France, 2003.

[4] A. Bemporad, A. Giua, and C. Seatzu. Synthesis of state-feedback optimal controllers for continuous time switched linear systems. In *Proc. 41th IEEE Conf. on Decision and Control*, pages 3182–3187, Las Vegas, USA, December 2002.

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

[6] G. Corriga, A. Giua, and G. Usai. An H_2 formulation for the design of a passive vibration-isolation system for cars. *Vehicle Systems Dynamics*, 26:381–393, 1996.

[7] G. Corriga, S. Sanna, and G. Usai. An optimal tandem active-passive suspension for road vehicles with minimum power consumption. *IEEE Trans. Industrial Electronics*, 38(3):210–216, 1991.

[8] A. Giua, M. Melas, C. Seatzu, and G. Usai. Design of a predictive semiactive suspension system. *Vehicle Systems Dynamics*, 41(4), 2003. (to appear).

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

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

[11] A. Giua, C. Seatzu, and G. Usai. Semiactive suspension design with an optimal gain switching target. *Vehicle Systems Dynamics*, 31:213–232, 1999.

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

[13] E. Göring, E.C. von Glasner, and P. Schützner R. Povel. Intelligent suspension systems for commercial vehicles. pages 1–12, Lyon, France, 1993.

[14] A. Hac. Suspension optimization of a 2-dof vehicle model using stochastic optimal control technique. *J. of Sound and Vibration*, 100(3):343–357, 1985.

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

[16] K.J. Kitching, D.J. Cole, and D. Cebon. Performance of semi-active damper for heavy vehicles. *ASME Journal of Dynamic Systems Measurement and Control*, 122:498–506, 2000.

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

[18] K. Ogata. *Modern Control Engineering.* Prentice Hall International Editions, 1990.

[19] B. Piccoli. Necessary conditions for hybrid optimization. In *Proc. 38th IEEE Conf. on Decision and Control*, pages 410–415, Phoenix, USA, December 1999.

[20] P. Riedinger, F. Kratz, C. Iung, and C. Zanne. Linear quadratic optimization for hybrid systems. In *Proc. 38th IEEE Conf. on Decision and Control*, pages 3059–3064, Phoenix, USA, December 1999.

[21] V. Roberti, B. Ouyahia, and A. Devallet. Oleopneumatic suspension with preview semi-active control law. In *Proc. Int. Cong. MV2, Active Control in Mechanical Engineering*, Lyon, France, 1993.

[22] D. Sammier, O. Sename, and L. Dugard. Commande par placement de poles de sospensions automobiles. In *Conf. Int. Francophone d'Automatique*, Nantes, France, July 2002. (in french).

[23] H.J. Sussmann. A maximum principle for hybrid optimal control problems. In *Proc. 38th IEEE Conf. on Decision and Control*, pages 425–430, Phoenix, USA, December 1999.

[24] A.G. Thompson. An active suspension with optimal linear state feedback. *J. of Sound and Vibration*, 5:187–203, 1976.

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