Simulation for the optimal event-based control of stochastic jump systems

Modeling, control and simulation of jump systems

Simona Mihăiţă · **Stéphane Mocanu**

Abstract This paper presents a continuous time simulation method for stochastic switching systems while applying the event-based control. The main system we have used is a multi-state integrator having a switching behavior, being described by a continuous-time Markov Chain. The objective of the event-based control method is to maintain the continuous system state variable between extreme limits. Control stopping limits have also been taken into consideration. Finally we present the results we have obtained in order to minimize the quadratic energy cost while applying event-based control.

Keywords stochastic switching systems · event-based control

1 Introduction

Lately, stochastic switching systems have been used as a special modeling method for dynamical systems, due to their both continuous-time and discrete switching behavior. These *stochastic switching systems* or *jump systems* have been widely used in transportation systems [1] - [2], automated highway systems [3], communication networks [4], robotics [5], automotive systems [6] and biological systems [7]. The switching between the states is represented by

Simona Mihaita
Gipsa-lab UMR 5216
961 rue de la Houille Blanche BP 46 F - 38402 GRENOBLE Cedex
Tel.: 33 (0)4 76 82 63 45
Fax: 33 (0)4 76 82 64 26
E-mail: Adriana-Simona.Mihaita@gipsa-lab.grenoble-inp.fr

Stephane Mocanu
Gipsa-lab UMR 5216
961 rue de la Houille Blanche BP 46 F - 38402 GRENOBLE Cedex
Tel.: 33 (0)4 76 82 62 56
Fax: 33 (0)4 76 82 64 26
E-mail: Stephane.Mocanu@gipsa-lab.grenoble-inp.fr

continuous-time Markov chains taking values in a finite state space; we can model the stochastic switching systems by using differential equations [8]:

$$\dot{x}(t) = f_{\sigma(t)}(t, x(t), u(t)), t \geq 0 \tag{1}$$

where $\sigma : [0, \infty) \rightarrow \{1, 2, .., N\}$ is the switching signal (a piecewise constant function with a countable number of discontinuities), $\mathbf{x}(t) \in \Re_n$ is the continuous component of the state taking real values, and $\mathbf{u}(t) \in \Re_n$ is the control command applied over the system. The switching signal σ completes the role of changing the dynamic of the system at each time instant t.

Despite their modeling flexibility, the stochastic switching systems are more challenging: analytical solutions are difficult to obtain and few algorithms for numerical simulation exist. Numerical simulation is a powerful analysis technique but the simulation method must be carefully chosen in order to obtain accurate results. For the stochastic switching systems the numerical simulation has to take into consideration both the continuous state aspect of the system as well as the discrete one. Different approaches have considered either the time discretisation which leads to high computing times if a good precision is desired, or applying restrictions to the continuous variables. We have to state that the continuous time evolution of the system has to be sampled in order to obtain precise results. But this procedure can introduce additional transitions at each step which will burden the simulation of the system. Some simulation techniques have also been developed for manufacturing systems [9], chemical systems [10], genetic networks [12], biochemical systems [13]; as well, numerical algorithms for the reachability analysis of stochastic hybrid systems have been provided in [11].

A numerical approach for the optimal stopping control problem of a class of deterministic Markov process with jumps has been presented in ([28], [26]) where a quantization approximation method is used; the quantization method has been recently used in the numerical probability of optimal stochastic control with applications in finance ([29], [30]) and consists in approximating the Markov chain by a quantized process. Although it's a flexible method based on the discretization of the process at certain steps with nice convergence properties, the Markov property is not maintained by the quantization algorithm and the quantized process is generally not markovian. Monte Carlo simulations or linear programming techniques would also seem appropriate to be used but they require assumptions related to the generator of the process which is generally not fulfilled by a Continuous-time Markov chains with jumps.

Our objective is to propose a continuous-time simulation algorithm of a class of stochastic switching systems which takes into consideration the random events that may change the evolution of the system as well as the random transitions between states at uncontrolled time intervals in the simulation. This algorithm would be applied in the scope of minimizing the energy consumed when applying control and offering appropriate performability results. The starting point of our work is an efficient simulation algorithm implemented for the evaluation of the performance level of production lines [14]; it has also been applied for the continuous-time simulation of the packet traffic in communication networks by [15]. We have therefore derived our simulation algorithm for the continuous-time simulation of a stochastic switching system, when applying event-based control.

We consider the *event-based control* as an appropriate control method for the random switching system that we use. Applying the control only when is needed and until certain conditions are met, gives the opportunity of simplifying the sensor system which equip the processus (only a detection threshold is needed instead of a continuous sampling). In [19] it has been proved that for certain types of systems the event-based control command allows to reduce the total energy that the system consumes by comparison to the control command based on periodical sampling (although a general formula hasn't been provided yet).

This *event-based control* method of actuating a system only when certain events appear has become an attractive approach to solve control problems in health care, transportation networks, process industry [16], satellite control [17], or biological systems [18]. Unfortunately there is little theory on the design of an event-based controller [19], [20], [21] and often the problem can be treated as a Markov decision process.

The system model that we use will be provided in the next section, followed by the continuous-time simulation algorithm in Section 3 when event-based control is applied. The main steps are discussed and intermediate results presented. A study case will be also described in Section 4.

2 Base Model

The main model we consider for our study is a particular type of stochastic switching system with piecewise constant variation rates of the continuous-time variable. We call this system a switching integrator with both continuous and discrete behavior for which the event-based control will be applied.

2.1 Uncontrolled Switching Integrator

We can describe the uncontrolled stochastic switching integrator by using the following differential equations:

$$\begin{cases} \dot{x}(t) = r_{Z(t)} \\ x(0) = x_0 \end{cases} \tag{2}$$

where $\mathbf{x}(t)$ is the state variable, $x_0 \in \Re$ is the initial state of the system, $Z(t)$ denotes the continuous Markov chain associated to the system and taking values in the finite state space $S = \{1, 2, .., N\}$, and $r_{Z(t)}$ represents the constant variation rate of the continuous variable $\mathbf{x}(t)$ so that $r_i > 0$, $\forall i \in \{1, .., M\}$, and $r_j < 0$, $\forall j \in \{M + 1, .., N\}$. The continuous-time Markov chain is represented by the following transition rate matrix Q :

$$\mathbf{Q} = \begin{pmatrix} -\sum_{j\neq 1}\lambda_{1,j} & \lambda_{1,2} & .. & \lambda_{1,N} \\ \lambda_{21} & -\sum_{j\neq 2}\lambda_{2,j} & .. & \lambda_{2,N} \\ .. & .. & .. & .. \\ \lambda_{N,1} & \lambda_{N,2} & .. & -\sum_{j\neq N}\lambda_{N,j} \end{pmatrix}$$

where $\lambda_{i,j}$ is the transition rate between the state i and j; also the transition probability between i and j is $p_{i,j} = \lambda_{i,j}/\sum_{j\neq i}\lambda_{i,j}$.

A graphical representation for a two-state switching integrator with associated state rates ($r_1 > 0$, $r_2 < 0$), can be seen in Fig. 1. Following the notations used in [24], we denote by σ_1 the event by which the system switches from state 1 to state 2 after a random time depending on the transition rate λ, and σ_2 the event by which the system switches from state 2 to state 1 after a random time depending on the transition rate μ.

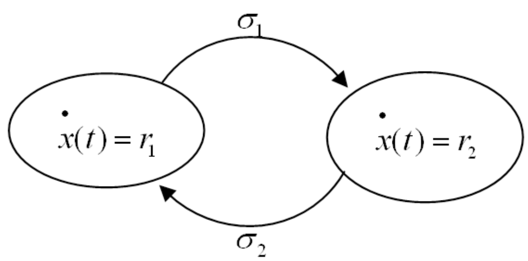

Fig. 1 Hybrid stochastic representation for the uncontrolled two-state switching integrator.

For the above system, when certain conditions are met, a particular type of control is considered, which is the event-based control.

2.2 Event-based Controlled Switching Integrator

Event-based control is often seen as a natural approach to many switching systems as it reacts quickly to disturbances, giving good performance. In [19], Åström showed for some examples that the event-based control can deal with multi-rate, asynchronism and latency which give great difficulties for classical sampled data systems. Few analytical results on the event-based control have been published. Recently, a similar problem has been presented by [26] and [27] based on numerical methods. Our objective is to propose a simulator for discrete event jump systems which can be used as a benchmark and reference for different analytical and numerical stochastic methods. We adopt an event-based controller which we believe it appropriately explores the command objectives for the considered stochastic jump systems.

The main objective is to maintain the system state variable between extreme boundaries: $\mathbf{x}(t) \in [X_{min}, X_{max}]$ while using a minimal energy. In order to define the main objective for energy minimisation, we define the following quadratic cost criterion which will be discussed at the end of this section:

$$J = \lim_{T\to\infty} \left(\frac{1}{T} \int_0^T [q \cdot x^2(t) + r \cdot u^2(t)]dt \right), q \geq 0, r > 0 \tag{3}$$

Every time one of the limits has been reached, control will be applied until $\mathbf{x}(t)$ reaches one of the optimal levels X_H, or X_L. Fig. 2 represents some sample paths of the switching integrator, with or without event-based control, which correspond to the following description of the system evolution:

- *no control* is applied if:
  - the state variable $\mathbf{x}(t)$ is in the no control area: $[X_L, X_H]$ or
  - the state variable is in the upper control interval (X_H, X_{max}) and no control was needed before the current time t or
  - the state variable is in the lower control interval (X_{min}, X_L) and no control was needed before the current time t;
- *high* control is applied if the maximal limit has been reached ($\mathbf{x}(t) = X_{max}$) or if the state vector is still in the upper control interval although high control has been applied before t; this means we have to continue applying the control until $x(t) = X_H$;
- *low* control is applied if the minimal limit has been reached ($\mathbf{x}(t) = X_{min}$) or if the state vector is still in the lower control interval although low control has been applied before t; the control has to be reapplied until $x(t) = X_L$.

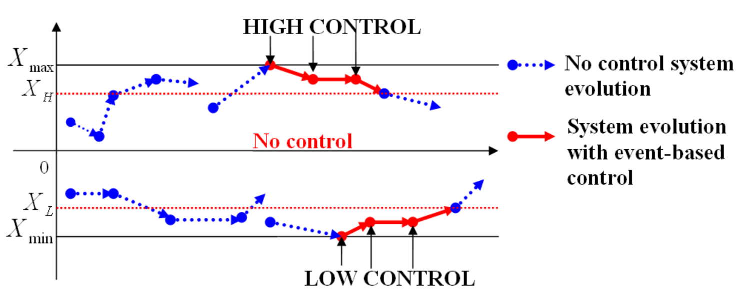

Fig. 2 Some sample paths of the event-based controlled switching integrator.

With the above specifications the differential equation for the controlled system [2] becomes:

$$\begin{cases} \dot{x}(t) = r_{Z(t)} + u_{Z(t)}(x(t)) \\ x(0) = x_0 \end{cases} \tag{4}$$

where

$$u_{Z(t)}(x(t)) = \begin{cases} 0 & \text{,if } C_1 \\ -QH_l & \text{,if } C_2,\ \forall l \in \{1, 2..N\} \\ +QL_m & \text{,if } C_3,\ \forall m \in \{1, 2..N\} \end{cases}$$
$$\begin{cases} C_1: & (x(t) \in [X_L, X_H]) \vee \\ & (x(t) \in (X_H, X_{max}) \wedge u(Z(t-\Delta t)) = 0) \vee \\ & (x(t) \in (X_{min}, X_L) \wedge u(Z(t-\Delta t)) = 0) \\ C_2: & (state = l) \wedge [(x(t) = X_{max}) \vee \\ & (x(t) \in (X_H, X_{max}) \wedge u(Z(t-\Delta t)) \neq 0)] \\ C_3: & (state = m) \wedge [(x(t) = X_{min}) \vee \\ & (x(t) \in (X_{min}, X_L) \wedge u(Z(t-\Delta t)) \neq 0)] \end{cases} \tag{5}$$

The QH_l notation denotes the *high* control applied when the Markov chain is in state l, QL_m is the *low* control applied when being in state m, while Δt represents an infinitesimal time interval. It is important to specify that for the control to take place, QH_l and QL_m will be chosen such that:

$$\begin{cases} r_l - QH_l < 0 & ,\forall QH_l > 0,\ l \in \{1, ..N\} \\ r_m + QL_m > 0 & ,\forall QL_m > 0,\ m \in \{1, ..N\} \end{cases}$$

In the view of the above discussion, the system described by [4] can also be modeled through the simple automaton of Figure 3. As well, in order to specify the way in which transitions from one state to another occur in [4], we need to define the events that can take place. We will denote by "*uncontrollable Markov chain events*" the events that are independent of the extreme boundaries or stopping control limits and cause the system to switch between states having the same control type (no control, high or low control); these events are represented by $\{\sigma_1, \sigma_2\}$. We denote by "*event-based control events*" the events that can change the type of control being applied to the system. We use the notation $x \uparrow X_{max}$ and $x \downarrow X_{min}$ to indicate an event that causes $\mathbf{x}(t)$ to reach the maximal or minimal limit respectively; $x \downarrow X_H$ and $x \uparrow X_L$ when high or low control needs to be stopped. These transitions have also been discontinuously marked in Figure 3.

2.3 Quadratic minimisation criterion

The behavior of the controlled system described for example by Fig. 2 can be seen as an alternance between two renewal processus having two renewal points X_H and X_L (periodically the system will start either in X_H or in X_L). The analytical analysis of such a system is difficult but this perspective allows us to define the quadratic minimisation criterion: the global energy minimisation of the system resides on the energy minimisation on each renewal period (a renewal period it is defined for example by the free evolution of the system between X_H and X_{max}, followed by the event-based control evolution between X_{max} and X_H; all the possible combinations - four in the above case - need to be taken into consideration).

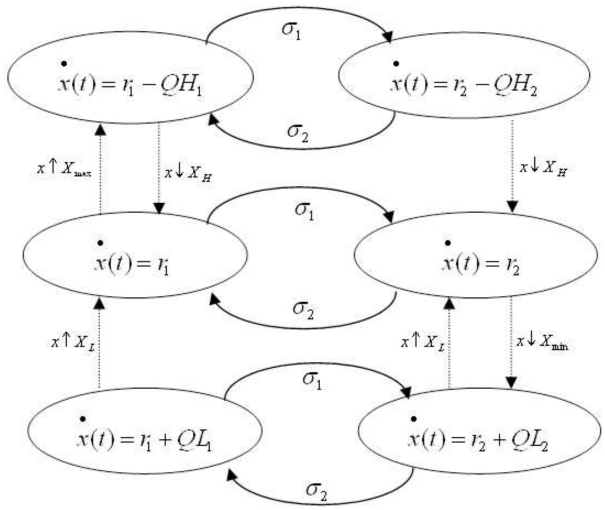

Fig. 3 Hybrid stochastic representation for a two-state switching integrator with event-based control.

As the renewal period is unknown due to a lack of analytical treatment of the problem (numerical approaches have been discussed in Section 1), we choose by the simulation method to minimise the mean statistical energy on time unit which can be expressed by the formula 3. On the other hand this formula can be computed directly during the simulation. The analytical treatment of the optimal control command is difficult even for a small system.

The quadratic energy criterion we have provided can be seen as a dynamic programming problem where the cost is evaluated over time due to the fact that the present choice of the control command $\mathbf{u}_{Z(t)}$ influences the future system evolution and the switching between the states (which is an energy consuming act as well). When analyzing the quadratic cost in 3, $\mathbf{u^2(t)}$ represents the energy consumed to apply the control while the $\mathbf{x^2}(t)$ member can be associated to the second order performability moment (M_2) which, in the performability analysis, it represents the variance of the state variable. Analytical studies for the evaluation of the system performance like the one presented in [22]-[23] compute the first performability moment (M_1) which actually represents the mean accumulated transition rate (or the mean accumulated reward) during the simulation; the second order moment is given by recursive differential equations using the first order moment. Our goal here is not to present the analytical method for computing these moments but to simply calculate them during the simulation in order to appreciate the system performability and the efficiency of our simulation method; comparisons between the simulated and the theoretical moments will be given in the following section.

Due to the stochastic aspect of the problem, the infinite horizon and the fact that the evolution rates (r_i) have not the same sign, a direct dynamical programming approach of solving a Hamilton-Jacobi-Bellman equation for our system is difficult. A solution based on an approximation model and a non-linear optimisation have been proposed in [25]. On the other hand, in this article our goal is to solve and analyse the model using the continuous-time simulation adapted to the discrete-events that change the evolution of the system.

3 Continuous-time Simulation

The simulation of the stochastic switching systems is an important technique to understand and analyze the system behavior. Compared to the discrete simulations, the continuous-time simulation helps reducing the execution time and memory space. As stated in the Introduction, we have adapted the simulation algorithm for continuous tandem production lines provided by [14] to the control conditions described in the previous section.

In the following we present the adapted event-based control algorithm for the multi-state system integrator [2]. Due to the random occurrence of the system transitions between the states, statistics have been also computed at certain regular times. A statistics period is chosen upon which the statistics on the state variable and on the time moment will be computed. This method will be helpful in the validation step afterwards.

Let us define the main parameters that we are going to use for the simulation. Let t denote the time and $Z(t)$ the system state at time t, where:

$$Z(t) = \begin{cases} iNC \text{ ,for } x(t) \in [X_L, X_H] \text{ and } u_{Z(t)} = 0, i \in \{1,..,N\} \\ iHC \text{ ,for } x(t) \in (X_H, X_{max}] \text{ and } u_{Z(t)} = -QH_i \\ iLC \text{ ,for } x(t) \in [X_{min}, X_L) \text{ and } u_{Z(t)} = +QL_i \end{cases} \quad (6)$$

Let N be the number of states of the system, X be the continuous state variable, X_{st} the statistics on the state variable and $r_i > 0$, $r_j < 0$ the positive and respectively the negative variation rates associated to the states $i \in \{1,..,M\}$, $j \in \{M+1,..,N\}$. When the control is needed, corresponding control measures will be applied so that: $r_k - QH_k < 0$ respectively $r_k + QL_k > 0$, $\forall k \in \{1,..,N\}$. We shall also define the set of all possible events for a state s: $E = \{\sigma_s, \{NH_{ss}\}, \{HN_{ss}\}, \{NL_{ss}\}, \{LN_{ss}\}\}$, where each of these events are defined in Table 1.

Symbol	Events represented
$\{\sigma_s\}$	Exit from state s to another state having the same control type
$\{NH_{ss}\}$	Switch from *no control* state s to *high* control state s
$\{HN_{ss}\}$	Switch from *high* control state s to *no* control state s
$\{NL_{ss}\}$	Switch from *no* control state s to *low* control state s
$\{LN_{ss}\}$	Switch from *low* control state s to *no* control state s

Table 1 List of all possible events

We will also consider λ_s to be the transition rate from the source state $s \in \{1, .., N\}$ to the next state determined by the next-event which has been chosen, T_f to be the simulation length and $State$ the discrete state history during the simulation taking values into the following set $\{iNC, iHC, iLC\}, i \in \{1, 2, .., N\}$. The random behavior of the system gives different results for each simulation. Therefore many simulations are needed on longer simulation time intervals, using the same data set; let N_r be the number of simulations that will be applied over one data set. We also have to state the fact that the $\{\sigma_s\}$ events are characteristic only to the Markov chain with random switching between the states, while $\{NH_{ss}\}$, $\{HN_{ss}\}$, $\{NL_{ss}\}$, $\{LN_{ss}\}$ are triggered only by the changes of the state variable $\mathbf{x}(t)$ (when the extreme boundaries or the control stopping points are reached).

For our system we have also taken into consideration the fact that each event is associated with a clock representing the time of the next occurrence of that event. When the clock runs down to zero, the event takes place bringing changes in the system state, according to the above description. The set of all next occurring times associated to the above events is $T = \{T_{\sigma_s}, T_{NH_{ss}}, T_{HN_{ss}}, T_{NL_s}, T_{LN_s}\}$. T_{σ_s} are specific to the Markov chain transitions and denote the exit times from a source state s; they are random samples of the exponential distribution of λ_s. Each time the system changes the type of control, the T_{σ_s} will be updated so that $T_{\sigma_s} = T_{\sigma_s} - T_{sw}$, where $T_{sw} \in \{T_{NH_{ss}}, T_{HN_{ss}}, T_{NL_{ss}}, T_{LN_{ss}}\}$.

$T_{NH_{ss}}$ is the notation for the next occurrence time to pass from *no* control to *high* control in state s; it is the time to reach the maximal limit X_{max} in state s and it can be written as $(X_{max} - X(t))/|r_i|$. $T_{HN_{ss}}$ stands for the time to pass from *high* control to *no* control and can be written as $(X(t) - X_H)/|r_s - QH_s|$; it is actually the time to reach X_H and stop the *high* control. $T_{NL_{ss}}$ denotes the time it takes to switch from *no* control to *low* control; it can be written as $(X(t) - X_{min})/|r_s|$ and represents the time to reach the minimal limit X_{min}. Finally, $T_{LN_{ss}}$ represents the time to reach X_L and stop the *low* control; it can be written as: $(X_L - X(t))/|r_s + QL_s|$.

The main steps of the simulation are presented in the following:

1. INITIALIZE SYSTEM PARAMETERS

The following parameters will be initialized: N, T_f, N_r, $X(1)$, $X_{st}(1)$, $r_i > 0$, $r_j < 0$, $i \in \{1, ..M\}, j \in \{M+1, ..N\}$, the statistics period d_{st}, the control parameters $QH_l > 0$, $QL_m > 0$, $l, m \in \{1, ..N\}$, the initial history state $State(1)$, the initial event time instant T_{sim}; the statistics time instant T_{st} and its initial index $L_{st} = 1$; the *low* or *high* energy consumed in order to apply the event-based control: $En_{QH} = 0$, $En_{QL} = 0$. We will also initialise the next event occurrence times starting state $s = State(1)$: $T = \{T_{\sigma_s}, T_{NH_{ss}}, T_{HN_{ss}}, T_{NL_s}, T_{LN_s}\}$.

2. CHOOSE NEXT EVENT

From all the above event possibilities for one state $s = State(j)$, the next event that will be applied is the one with the smallest associated occurrence time.

$$\begin{aligned}&\text{While} \quad T_{sim}(j) \le T_f \quad // \text{ choose the next possible event}\\&\quad \Delta t = min\{T_{\sigma_s}, T_{NH_{ss}}, T_{HN_{ss}}, T_{NL_{ss}}, T_{LN_{ss}}\}\\&\quad next_{ev} = \text{next possible event corresponding to} \Delta t\\&\quad next_{ev} \in \{\{\sigma_s\}, \{NH_{ss}\}, \{HN_{ss}\}, \{NL_{ss}\}, \{LN_{ss}\}\}\end{aligned} \tag{7}$$

3. ANALYZE NEXT EVENT

Depending on the current state of the system, we analyze the next possible event of the system which has been chosen, and its associated occurrence time. When $next_{ev} = \{\sigma_s\}$ the system switches from the current state s to a destination state d having the same control type. The state d will be randomly chosen by comparing the transition probabilities between s an the other states of the system $(p_{s,i}, \{s, i \in Z(t)\})$ having the same control type $(p_{i,i} = 0)$. The next occurrence time T_{σ_s} will become a sample of the exponential distribution of the sojourn time in the next state d. On the other hand, when we change the control type for the current state s, then $T_{\sigma_s} = T_{\sigma_s} - \Delta t$.

case $next_{ev} = \{\sigma_s\}$
  $X(j) \in [X_L, X_H]$
    $State(j) = mNC, m \in \{1, .., N\}$ //*No control*
      $State(j+1) = dNC$ // Random choice upon $p_{mNC,nNC}, n \in \{1, ..N\}$
      Compute $T_{\sigma_d}, T_{NH_{dd}}, T_{NL_{dd}}$
  $X(j) \in (X_H, X_{max})$
    $State(j) = mNC, m \in \{1, .., N\}$ //*No control*
      $State(j+1) = dNC$ // Random choice upon $p_{mNC,nNC}, n \in \{1, ..N\}$
      Compute $T_{\sigma_d}, T_{NH_{dd}}, T_{NL_{dd}}$
    $State(j) = mHC, m \in \{1, .., N\}$ //continue the *High control*
      $State(j+1) = dHC$ // Random choice upon $p_{mNC,nNC}, n \in \{1, ..N\}$
      Compute $T_{\sigma_d}, T_{HN_{dd}}$
  $X(j) = X_{max}$ // apply the *High control*
    $State(j) = mHC, m \in \{1, .., N\}$
      $State(j+1) = dHC$ // Random choice upon $p_{mNC,nNC}, n \in \{1, ..N\}$
      Compute $T_{\sigma_d}, T_{HN_{dd}}$
  $X(j) \in (X_{min}, X_L)$
    $State(j) = mNC, m \in \{1, .., N\}$ //*No control*
      $State(j+1) = dNC$ // Random choice upon $p_{mNC,nNC}, n \in \{1, ..N\}$
      Compute $T_{\sigma_d}, T_{NH_{dd}}, T_{NL_{dd}}$
    $State(j) = mLC, m \in \{1, .., N\}$ // continue the *Low control*
      $State(j+1) = dLC$ // Random choice upon $p_{mNC,nNC}, n \in \{1, ..N\}$
      Compute $T_{\sigma_d}, T_{LN_{dd}}$
  $X(j) = X_{min}$ // apply the *Low control*
    $State(j) = mLC, m \in \{1, .., N\}$
      $State(j+1) = dLC$ // Random choice upon $p_{mNC,nNC}, n \in \{1, ..N\}$
      Compute $T_{\sigma_d}, T_{LN_{dd}}$
case $next_{ev} = \{NH_{ss}\}$ // switch to *High control*
  $X(j) \in (X_{min}, X_{max})$
    $State(j) = mNC, m \in \{1, .., N\}$
      $State(j+1) = mHC$
      Update $T_{\sigma_s} = T_{\sigma_s} - \Delta t$, Compute $T_{NH_{ss}}$

case $next_{ev} = \{NL_{ss}\}$ // switch to *Low control*
$X(j) \in (X_{min}, X_{max})$
$State(j) = mNC, m \in \{1, .., N\}$
$State(j+1) = mLC$
Update $T_{\sigma_s} = T_{\sigma_s} - \Delta t$, Compute $T_{NL_{ss}}$
case $next_{ev} = \{HN_{ss}\}$ // stop the *High control*
$X(j) \in (X_H, X_{max}]$
$State(j) = mHC, m \in \{1, .., N\}$
$State(j+1) = mNC$
Update $T_{\sigma_s} = T_{\sigma_s} - \Delta t$, Compute $T_{HN_{ss}}$
case $next_{ev} = \{LN_{ss}\}$ // stop the *Low control*
$X(j) \in [X_{min}, X_L)$
$State(j) = mLC, m \in \{1, .., N\}$
$State(j+1) = mNC$
Update $T_{\sigma_s} = T_{\sigma_s} - \Delta t$, Compute $T_{LN_{ss}}$

4. UPDATE THE SYSTEM

Once the next event has been chosen and analyzed, the next simulation time and the next state variable will be updated according to the event which has been chosen. r_s is the variation rate associated to the current state s which can take different values: $\{r_m, (r_m - QH_m), (r_m + QL_m)\}, m \in \{1, .., N\}$; (for example if the *high control* is applied in state s then the variation rate becomes $r_s - QH_s$). The consumed energies are computed when we apply the *high* or *low* control in state s.

$T_{sim}(j+1) = T_{sim}(j) + \Delta t$; *//advance the simulation time*
$X(j+1) = X(j) + r_s \cdot \Delta t$; *//update the state variable*
$En_{QH} = En_{QH} + QH_s^2 \cdot \Delta t$; *//compute the consumed energies*
$En_{QL} = En_{QL} + QL_s^2 \cdot \Delta t$;

5. COMPUTING STATISTICS AT REGULAR TIMES

Random switching in the behavior of the system gives random continuous state variables which can badly influence the accuracy of the results. A sampling method is needed on the final state variable X at regular time intervals, given by the statistics period we have considered d_{st}, and not at each event occurrence.

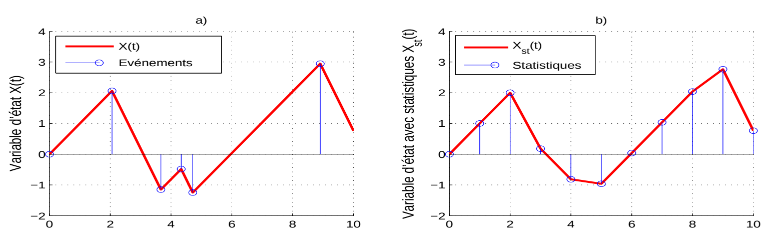

Fig. 4 a) State variable $X(t)$ b) State variable with statistics $X_{st}(t)$

If two consecutive events take place during the same statistic period, no action will be taken. An example of applying the statistics with $d_{st} = 1$ is shown in Figure 4 and the steps listed below.

$d = floor[(T_{sim}(j+1) - T_{st}(L_{st}))/d_{st}];$
$if \quad d \geq 1 \quad //compute\ consecutive\ statistics$
$\quad k = 1$
$\quad while \quad k \leq d$
$\quad\quad T_{st}(L_{st}+k) = T_{st}(L_{st}) + k \cdot d_{st};$
$\quad\quad X_{st}(L_{st}+k) = X(j) + r_s * (T_{st}(L_{st}+k) - T_{sim}(j));$
$\quad\quad k = k+1;$
$\quad end \quad while$
$\quad L_{st} = L_{st} + k - 1;$
$else \quad //no\ action\ will\ be\ taken;$
$end \quad if;$
$j = j+1; \quad //advance\ the\ simulation$
$end \quad while \quad //see$ [7]

An important aspect of applying the sampling can be noticed when computing the performability moments ($M1$ and $M2$) of the considered system. Due to the state rates associated to the Markov chain, one would use the sampled state variable in the study of the transient performance analysis of fault-tolerant systems. When comparing the first and second order moments of cumulative performability we observe that the sampling improves the data accuracy. This is rather obvious as the random switching between the states due to uncontrollable Markov events causes a non uniform evolution of the state variable $\mathbf{x(t)}$ (events can occur either consecutively after small event times or rather at long time intervals if the sojourn time spent in one state is unexpectedly long). The sampling would ensure homogeneity and improve the quality of the data.

In Figure 5, graphs in a) and b) show the evolution of the first and respectively second order moment of performability without applying the sampling procedure (both simulation and analytical results are represented on the same plot area) for a two state system having the following parameters: $r_1 = 3$, $r_2 = -2$, $\lambda_{12} = 0.4$, $\lambda_{21} = 0.6$, $X_{max} = 10$, $X_{min} = 0$, $X_H = 8$, $X_L = 4$, $T_f = 1000$. On the other hand graphs in c) and d) show the two performability moments when the sampling procedure is applied over the simulation (with $d_{st} = 1$). By comparing for example a) and c) it is obvious to see that the sampling method improves data accuracy and provides better results.

The numerical comparison between the two types of moments with or without sampling is also given in Table 2. The *Error M1* represents the error between the simulation and analytical results for the first performability moment. It is obvious to see the high difference and bad accuracy when no sampling is applied over the state vector of the system due to the random events that can change the system evolution.

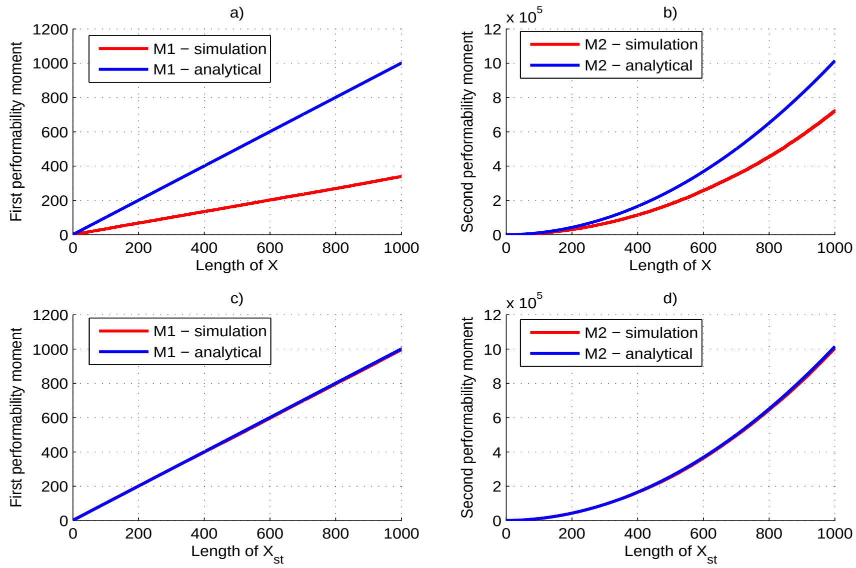

Fig. 5 a) First moment of performability (M1) - without sampling b) Second moment of performability (M2) - without sampling c) First moment of performability (M1) - with sampling d) Second moment of performability (M2) - with sampling.

$Figure5$	$Type$	$ErrorM1[\%]$	$ErrorM2[\%]$
a) and b)	No sampling	51.64	66.2
c) and d)	With sampling	0.09	0.16

Table 2 Difference between performability moments for a two state switching system

6. SIMULATION END

At the end of the simulation we compute the output measures: the mean state variable (X_m), the variance of the statistics state variable X_{st} (V_{X_s}), the energy consumed in order to apply the event-based control during the simulation length (En_{tot}).

$$\begin{aligned}&X_m = mean(X);\\ &V_{X_{st}} = \sum_{i=1}^{T_f} X_{st}(i)^2/T_f;\\ &En_{tot} = (En_{QH} + En_{QL})/T_f.\end{aligned}$$

The above simulation algorithm for a two-state switching integrator can be easily followed using the graphical representation provided in Figure 3. On the other hand, Figure 6 shows the difference between the random behavior of a two state system, with and without the event-based control, for the following data entry: $T_f = 300$, $r_1 = 1, r_2 = -2, \lambda = 0.4, \mu = 0.6, QH_1 = QH_2 = 1.5, QL_1 = QL_2 = 2.5$, $[X_{min}, X_{max}] = [-10, 10]$ and $[X_L, X_H] = [-5, 5]$.

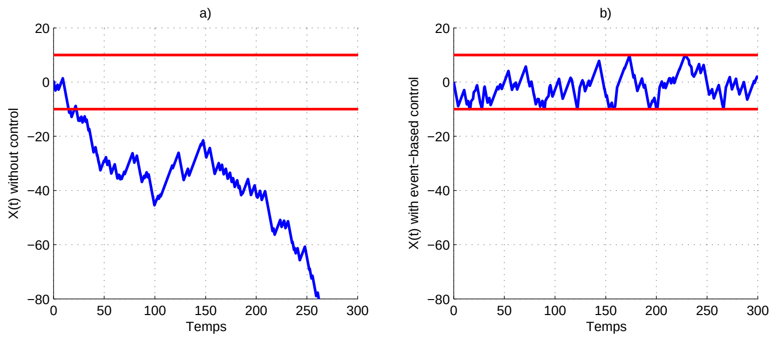

Fig. 6 (a) Uncontrolled Switching Integrator (b) Event-based controlled Switching Integrator over $[X_{min}, X_{max}] = [10, -10]$ with $[X_L, X_H] = [5, -5]$.

The random behavior of the system gives different results each time the simulation runs. Therefore many replications using the same data set are needed but on a longer simulation time. Means of the output measures will be computed over the replications. This will allow a proper analysis and result interpretation which will be given in Section 4.

4 Results

This section provides a study case example for the simulation method presented above. We consider a two-state switching integrator, for which the event-based control algorithm can provide a minimal consumed energy for applying the control. The simulations have been made in Matlab 7.9.0.529 on a four quad core machine, having 3GHz and 4GB RAM. We choose the initial parameters which define the system: the simulation length $T_f = 5000$(time units), the number of replications $N_r = 50.000$, the variation rates associated to the states ($r_1 = 5$, $r_2 = -5$), the transition rates from one state to another ($\lambda = 0.4, \mu = 0.6$) and the control area: $X_{min} = -100$, $X_{max} = 100$.

We variate the control parameters QH_1, QH_2, QL_1, QL_2 so that they respect the conditions of the event-based control. Taking into consideration the fact that the system has many parameters which can vary, we consider for simplicity that $QH_1 = QH_2 \in \{5.1, 5.2, ..30\}$, $QL_1 = QL_2 \in \{5.1, 5.2, ..30\}$). As well, different variations for the stopping control limits have been done ($X_H \in \{0, 10, 20, 30\}$, $X_L \in \{0, -10, -20, -30\}$). Figure 7 is a graphical representation of the total energy cost [3] obtained by applying the event-based control over the considered system. A minimal energy value is obtained, which corresponds to the control parameter $QH_1 = 14.9$ and the stopping control limit $X_H = 0$. A similar energy evolution is obtained for switching integrators having different variation rates and different stopping control limits.

Another study case that we present is that of a larger system, having four states, and different transition rates associated to these states. Our objective is

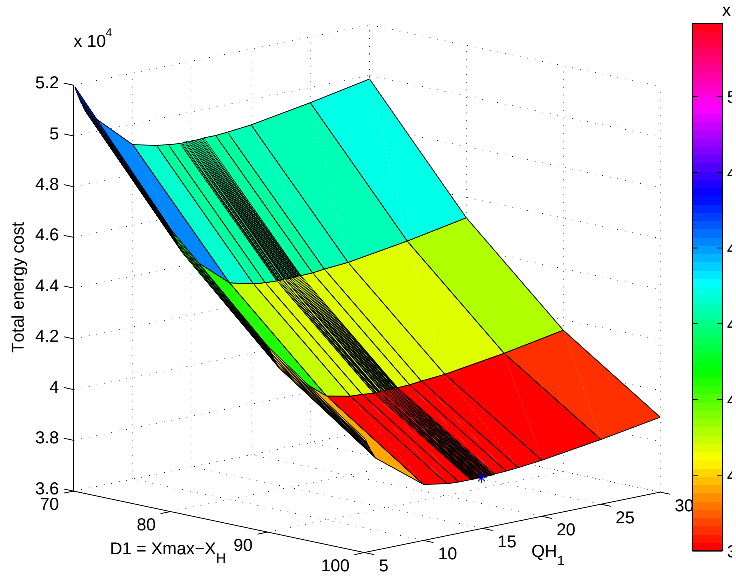

Fig. 7 Energy variation for the considered switching integrator.

to validate the method for larger systems in terms of simulating time, minimal energy obtained and optimal control parameters (although the system has only four states, varying all the above parameters can considerably increase the computation time). We have considered different control measures and different variations on the control stopping limits in order to simulate the total sets of possible variations on the system. The data we have considered is represented by the following: $r_1 = 7$, $r_2 = -4$, $r_3 = 5$, $r_4 = -2$, over the control interval $[X_{min}, X_{max}] = [0, 1]$; the control measures and the stopping control limits have been varied over the following sets: $QH_1 \in \{7.1, 7, 2...., 8\}$, $QH_2 = QH_3 = QH_4$, $QL_1 \in \{4.1, 4.2, .., 5\}$, $QL_2 = QL_3 = QL_4$, $X_H \in \{0.9, 0.8, 0.7, .., 0.3\}$, $X_L \in \{0.1, 0.2, ...0.7\}$. A graphical representation of the above variations is shown in Figure 8. The number of variations on (X_H, X_L) have the following meaning: the variation 1 on (X_H, X_L) is represented by: $X_H = 0.9, X_L = 0.1$, the variation 2 is : $X_H = 0.9, X_L = 0.2$, etc. As well the number of variations on (Q_H, Q_L) is presented by the following pairs: variation 1 by $Q_H = 7.1, Q_L = 4.1$, variation 2 by $Q_H = 7.1, Q_L = 4.2$, until variation 100 by $Q_H = 8, Q_L = 5$. Although not seisable on the figure at a first glance, the minimal energy cost was obtained for the following optimal values: $X_H = 0.8$, $X_L = 0.4$, $QH = 7.2$, $QL = 4.8$.

An important observation is that the computing time for larger systems is bigger than the one obtained for a simple two state system, and of course influenced by the number of variations that we apply to the control parameters

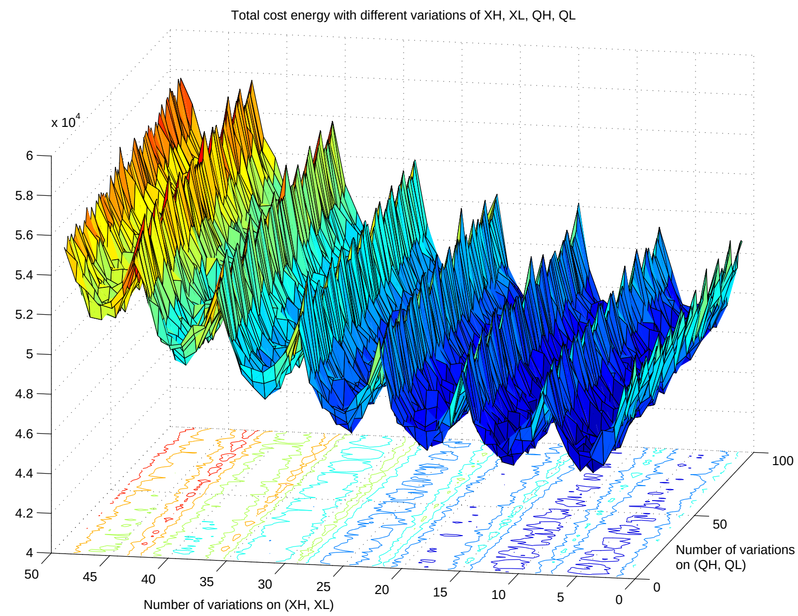

Fig. 8 Variations on the energy cost for a 4 state switching integrator.

and stopping limits. As a consequence, we also evaluate the adaptability of our event-based control algorithm in terms of computing time, determined by three main parameters: the simulation length(T_f), the number of simulations (N_r) and the number of the system states (N). More explicitly, we fix two of the three parameters and we evaluate how the computing time evolves with the third one. The control parameters and the stopping control limits have also been considered to be fix. Table 3 shows the computing time (T_c) for a two-state switching integrator over $N_r = 100$ simulations; the statistical period is $d_{st} = 1$. In this case, the computing time is proportional with the simulation length as represented in Figure 9 (due to different scales, the graphics are normalized).

For the two state integrator with fixed simulation length ($T_f = 1000$), when the number of simulations grows, the computing time has a considerable augmentation as it's represented in Table 4. When the number of states of the Markov chain grows, we are in a special case which is represented in Table 5, where the length of the simulation and the number of simulations are fixed ($T_f = 1000$, $N_r = 1000$). In this case, although the computation time is highly related to the number of states of the Markov chain, there is no exponential augmentation. We can easily observe the quasi-linear evolution in Figure 9, which confirms the proper scalability of the proposed algorithm to the size of the system, assuring that it can be properly used for larger Markov chains.

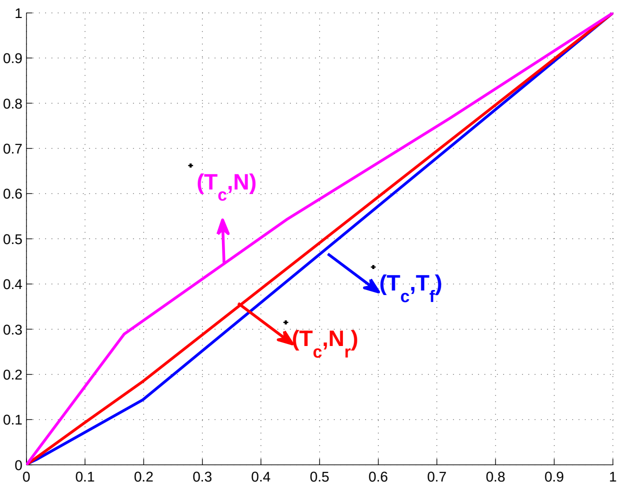

Fig. 9 T_c over T_f, N_r et N

T_f	10	50	100	1000	5000
T_c	0.45[sec]	0.54[sec]	0.63[sec]	2.57[sec]	15.22[sec]

Table 3 Computing time T_c over the simulation length T_f.

N_r	100	1000	5000	10000	50000
T_c	2.57[sec]	22.18[sec]	1.82[min]	3.61[min]	19.4[min]

Table 4 Computing time T_c over the number of simulations N_r.

N	2	5	10	15	20	30
T_c	22.18[sec]	1.61[min]	2.7[min]	3.66[min]	4.66[min]	6.68[min]

Table 5 Computing time T_c over the number of states N.

5 Conclusions

This article presents a simulation method in continuous time for the stochastic switching systems. Taking into consideration the random events that change the behavior of the system, we apply the control only when is needed. The algorithm can be easily used for stochastic switching integrators having a large number of states and can be seen as an efficient method to obtain the minimal energy consumed by these switching systems when applying the event-based control. As well it can be used as a validation method for analytical results concerning the stochastic switching systems having the event-based control.

References

1. Pola,G. and Bujorianu,M.L. and Lygeros,J. and Di Benedetto,M.D., IFAC Conf. on Analysis and Design of Hybrid Systems, 720-733, (2003)
2. Tomlin, C. and Pappas, G.J. and Sastry, S. *Conflict resolution for air traffic management: a study in multi-agent hybrid systems* IEEE Trans. on Aut. Control, 43, 4, 508-521, 1998
3. Varaiya, P. *Smart cars on smart roads: problems of control* IEEE Trans. on Automatic Control, 9, 52, 195-207, (1993)
4. Hespanha,J.P. Hybrid Systems: Computation and Control, 322-338, Springer Verlag, Berlin (2005)
5. Egerstedt,M. Hybrid Systems: Computation and Control, 103-116, Springer, Berlin (2000)
6. Balluchi,A. and Benvenuti,L. and Benedetto,M. D. D. and Miconi,G.M. and Pozzi,U. and Villa,T. and Wong-Toi,H. and Sangiovanni-Vincentelli,A.L. Hybrid Systems: Computation and Control, 32-44, Springer, Berlin (2000)
7. Khare,N. A. and Amonlirdviman,K. and Tree,D.R.P. and Chen,W.S. and Axelrod,J.D. and Tomlin,C.J. *Mathematical modeling of planar cell polarity to understand domineering nonautonomy* Science, 307, 5708, 423-426, (2005)
8. Liberzon Daniel, Switching in Systems and Control, Birkhauser, Boston, (2003)
9. Wei, K.C. and Tsao, Q.Q. and Otto, N.C., Determining Buffer Size Requirements Using Stochastic Approximations Methods, Ford, Research Repor,t SR-89-73, (1989)
10. Gibson,M.A. and Bruck,J. *Efficient exact stochastic simulation chemical systems with many species and many channels*, J. Phys. Chem., 104, 9, 1876 -1889, (2000)
11. Cassandras,Christos G. and Lygeros John, Stochastic Hybrid systems(Automation and Control Engineering), Taylor and Francis, New York, (2007)
12. Hu, J. and Wu,W. and Sastry,S. Modeling subtilin production in bacillus subtilis using stochastic hybrid systems, Hybrid Systems Computation and Control, Springer, 417 - 31, (2004)
13. Salis,H. and Kaznessis,Y. *Accurate hybrid stochastic simulation of a system of coupled chemical or biochemical reactions*, J. Chem. Phys., 122, 5, 54-103, (2005)
14. Suri,Rajan and Fu,Bor-Ruey Proceedings of the Winter Simulation Conf., 968-977, (1991)
15. Royer Alexandre,Évaluation de performances de réseaux de communication á l'aide de chaînes de Markov Hybride, PhD Thesis, Institut National Polytechnique de Grenoble, (2006)
16. Pettersson,J. and Ledung,L. and Zhang,X. Decision Support for pulp mill operations based on large-scale optimization, Control Systems Conf., (2006)
17. Dodds,S.J *Adaptive, high precision, satellite attitude control for microprocessor implementation*, Automatica, 17, 4, 563 - 573, (1981)
18. Wilson,H.R. *Spikes, Decisions, and Actions: The Dynamical Foundations of Neuroscience, Oxford University Press*, USA, (1999)
19. Åström, K.J. *Analysis and Design of Nonlinear Control Systems*, 127-147, Springer, (2002)
20. Kofman,E. and Braslavsky,J. *Level crossing sampling in feedback stabilization under data-rate constraints, 45th IEEE Conf. on Decision and Control*, 4423-4428, (2006)
21. Tabuada,P. *Event-triggered real-time scheduling of stabilizing control tasks, IEEE Trans. on Automatic Control*, 52, 9, 1680-1685, (2007)
22. Viswnadham,N. and Pattipati,Krishna R. and Gopalakrishna,V. *Performability Studies of Automated Manufacturing Systems with Multiple Part Types*, IEEE Trans. on Robotics and Automation, 11, 5, 692-709, (1995)
23. Balakrishna,R. Iyer and Donatiello,L. and Heidelberger,P. *Analysis of Performability for Stochastic Models of Fault-Tolerant Systems*, IEEE Trans. on Computers, C-35, 10, 902-907, (1986)
24. Cassandras,Christos G. Stochastic Hybrid Systems, Taylor and Francis, 6, 139-164, New York, (2008)
25. Mihaita, A.Simona and Mocanu, Stephane, *An Energy Model for Event-Based Control of a Switched Integrator*, IFAC World Congress, Milano, (2011)

26. de Saporta, Benoîte and Dufour,F. and Zhang,H. and Elegbede, C., *Optimal stopping for predictive maintenance*, to appear in Journal of Risk and Reliability, (2011), http://hal.archives-ouvertes.fr/hal-00554759/en/
27. de Saporta, B. and Dufour,F., *Numerical method for impulse control of Piecewise Deterministic Markov Processes*, INRIA Technical Report, http://hal.archives-ouvertes.fr/hal-00541413/en/
28. Brandejsky, A. and de Saporta, B. and Dufour, F., *Numerical methods for the exit time of a piecewise-deterministic Markov process*, INRIA Technical Report, http://hal.archives-ouvertes.fr/hal-00546339/en/
29. Bally, V. and Pags, G., *A quantization algorithm for solving multi-dimensional discrete-time optimal stopping problems*, Bernoulli 9, 6, 1003-1049, (2003)
30. Pags, G. and Pham, H. and Printemps, J., *Optimal quantization methods and applications to numerical problems in finance*, in *Handbook of Computational and Numerical Methods in Finance*, 253-297, *Birkhäuser*, Boston, MA, (2004)