European Journal of Control (2010)6:583–594

DOI:10.3166/EJC.16.583–594

European Journal of Control

Stochastic Hybrid Systems: A Powerful Framework for Complex, Large Scale Applications

John Lygeros[1,**], Maria Prandini[2,*]

[1] Automatic Control Laboratory, ETH Zurich, Switzerland;
[2] Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy

Stochastic hybrid systems allow one to model the interaction between continuous dynamics, discrete dynamics and probabilistic uncertainty. Because of their versatility, stochastic hybrid systems have emerged as a powerful framework for capturing the intricacies of complex systems. Motivated by this, considerable research effort has been devoted to the development of modeling, analysis and control methods for stochastic hybrid systems. In this introductory paper to the special issue, we outline recent progress in this challenging research area and argue that synergies between methods developed by different communities in computer science, control engineering, and stochastic analysis should be exploited to allow progress in real life, large scale applications.

Keywords: Stochastic hybrid systems; Complex systems modeling and control

1. Introduction

Over the past two decades hybrid systems have become established as a convenient framework for modeling, analysis, and control of complex systems. Hybrid systems are dynamical systems that involve the interaction of continuous (or time driven) and discrete (or event driven) dynamics. An important driving force fueling research in the area of hybrid systems has been their application to embedded control and computation, where by design digital, event driven devices have to interact with an analogue, time driven environment.

Much of the work on hybrid systems has focused on deterministic models that completely characterize the future of the system by defining its evolution in an imperative way (the model specifies what the evolution must be), without allowing any uncertainty. In practice, it is often desirable to introduce uncertainty, to allow, for example, under-modeling of certain parts of the system, external un-modeled disturbances, etc. Motivated by this, researchers in discrete event systems and in hybrid systems have introduced what are known as non-deterministic models. In non-deterministic hybrid systems, the evolution is defined in a declarative way (the system specifies what solutions are allowed) as opposed to the imperative way. Uncertainty can affect the dynamics of a non-deterministic hybrid system in different ways: choice of continuous evolution, choice of discrete transition destination, or choice between continuous evolution and a discrete transition. "Choice" in this setting may reflect disturbances that add uncertainty about the system evolution, but also control inputs that can be used to select among the different choices and hence steer the system evolution.

Deterministic and non-deterministic hybrid systems are versatile, can capture a wide range of behaviors encountered in large scale, complex systems and have proved invaluable in a number of applications, [49]. They do, however, have their limitations. In particular, non-deterministic systems provide no way of distinguishing between their many possible solutions. This implies that

*Correspondence to: M. Prandini, E-mail: prandini@elet.polimi.it
** E-mail: lygeros@control.ee.ethz.ch

Received 20 June 2010; Accepted 29 July 2010
Recommended by D.W. Clarke

one can only pose qualitative, yes-no type questions. For example, a safety question for non-deterministic hybrid systems admits only one of two answers: "The system is safe" (if none of the solutions of the system ever reaches an unsafe state), or "the system is not safe" (if some solution reaches some unsafe state).

In some applications, a worst case analysis of non-deterministic hybrid systems may be too coarse. Indeed, requirements for safety-critical systems often take the form of the probability of engaging into undesirable behavior to be below an acceptable threshold. For example, in Air Traffic Management (ATM) the question "Is it possible for an accident to happen in the ATM system?" is not particularly meaningful, since accidents do occasionally happen and, hence, the answer must be "yes". Quantitative questions like "What is the probability that an accident happens in the ATM system?" and "How can the probability of an accident be reduced?" better capture the notion of safety in ATM, and can provide more insight into the safety issue.

The need for finer, probabilistic analysis of uncertain systems has led to the study of an even wider class of hybrid systems. These systems allow, for example, things such as random failures causing unexpected transitions from one discrete state to another, or random task execution times which affect how long the system spends in different modes. Randomness may also enter the picture through noise in one or more of the continuous states of the system, in which case we must resort to stochastic differential/difference equations. To allow the realistic modeling of such phenomena, researchers have extended their study of hybrid systems beyond continuous and discrete dynamics to include probabilistic aspects. This has led to the more general class of Stochastic Hybrid Systems (SHS). Various classes of stochastic hybrid models have been introduced and numerous case studies in the literature have illustrated their potential in diverse application domains such as control of telecommunication networks, air traffic, manufacturing, biology and finance (see, for example, [14, 23] for an overview). Part of the research in this area has been driven by computer scientists, in the area of formal methods. Different approaches have originated in the engineering literature, in particular in the area of automatic control. Each of the approaches has its own advantages and disadvantages and has been applied successfully to several application areas. However, synergies still need to be deeply explored and fully exploited in order to address the challenges posed by real life, large scale applications.

In this introductory paper we provide an overview of the main issues and recent developments in the modeling, analysis and control of SHS. In the process we highlight the research challenges that SHS pose, challenges that in our view can be addressed only through interdisciplinary research among computer scientists, stochastic analysts and control engineers. Examples of papers that attempt to forge links between these different areas are included later on in this special issue.

We start with an overview of modeling issues that arise in the study of SHS in Section 2. Because the coverage is rather wide, we anticipate that the presentation will be by necessity terse and will overlook several important technical issues. A brief survey of the methods for the analysis and control of SHS is presented in Section 3, with the goal of outlining their strengths and weaknesses with respect to complex, large scale applications such as, e.g., power networks, biochemical networks, air traffic management. The focus will be on methods which have solid theoretical foundations but that can also be implemented in computational tools. In Section 4, a discussion on possible directions of research concludes the paper.

This special issue includes further four papers that address different issues on modeling, analysis and control of SHS. The first paper "Stability of stochastic delay hybrid systems with jumps" by Chenggui Yuan and Xuerong Mao studies stability of a quite general class of SHS, which requires sophisticated tools from stochastic analysis. The second paper "Reachability analysis of stochastic hybrid systems: A biodiesel production system" by Derek Riley, Kasandra Riley, and Xenofon Koutsoukos addresses the modeling and reachability analysis issues for a biodiesel production system, proposing a SHS model and adopting verification methods based on dynamic programming and the Monte Carlo approach. The third paper "Approximate model checking of stochastic hybrid systems" by A. Abate, J.P. Katoen, J. Lygeros, and M. Prandini represents a first step toward the development of a fully automatic procedure for the approximate verification of SHS. Finally, the fourth paper "Perturbation Analysis and Optimization of Stochastic Hybrid Systems" by Christos G. Cassandras, Yorai Wardi, Christos G. Panayiotou, and Chen Yao focuses on the optimal design issue and proposes a general framework for carrying out perturbation analysis in SHS of arbitrary structure.

2. Stochastic Hybrid Systems – Modeling

The great interest of the research community in SHS has produced a number of different types of stochastic hybrid models. The main difference between these classes of stochastic hybrid models lies in the way the stochasticity enters the process [59]. In order to outline the different ways in which uncertainty can affect the dynamics of a hybrid system, we first give a high level overview of continuous-time non-deterministic hybrid systems in Section 2.1. We then discuss in Section 2.2 how these sources of uncertainty can be replaced by probabilities and outline the different classes of stochastic hybrid models that arise in the process. In Section 2.3, we also establish a connection to discrete-time stochastic hybrid models, which still provide a powerful modeling framework without some of the technical complications associated with their continuous-time counterparts.

2.1. Non-Deterministic Hybrid Models

We start by recalling the definition of a general class of continuous-time non-deterministic hybrid systems known in the literature as hybrid automata (see for example [51, 3]).

Definition 1 (Hybrid automaton): *A hybrid automaton is a collection $H = (\mathcal{Q}, \mathcal{X}, \mathcal{F}, Init, Dom, \mathcal{E}, \mathcal{G}, \mathcal{R})$, where*

- *$\mathcal{Q}$ is a finite set of discrete states*
- *$\mathcal{X} = \Re^n$ is a set of continuous states*
- *$\mathcal{F}(\cdot,\cdot) : \mathcal{Q} \times \mathcal{X} \to 2^{\mathcal{X}}$ is a differential inclusion*
- *$Init \subseteq \mathcal{Q} \times \mathcal{X}$ is a set of initial states*
- *$Dom(\cdot) : \mathcal{Q} \to 2^{\mathcal{X}}$ is a domain*
- *$\mathcal{E} \subseteq \mathcal{Q} \times \mathcal{Q}$ is a set of edges*
- *$\mathcal{G}(\cdot) : \mathcal{E} \to 2^{\mathcal{X}}$ is a guard condition*
- *$\mathcal{R}(\cdot,\cdot) : \mathcal{E} \times \mathcal{X} \to 2^{\mathcal{X}}$ is a reset map*

Note that here $2^{\mathcal{X}}$ denotes the set of all subsets of $\mathcal{X}$ so that the maps appearing in Definition 1 are all set-valued. We refer to $(q, x) \in \mathcal{Q} \times \mathcal{X}$ as the *state* of H, and to the discrete state $q \in \mathcal{Q}$ as the *mode* of H.

Hybrid automata define possible evolutions for their state. Informally, starting from an initial value $(q_0, x_0) \in Init$, the continuous state x evolves according to a solution of the differential inclusion $\dot{x} \in \mathcal{F}(q_0, x)$ (continuous evolution), while the discrete state q remains constant $q(t) = q_0$. Continuous evolution can go on as long as x remains in $Dom(q_0)$. If at some point the continuous state x reaches the guard $\mathcal{G}(q_0, q_1) \subseteq \mathcal{X}$ of some edge $(q_0, q_1) \in \mathcal{E}$, a discrete transition from q_0 to q_1 is enabled and the discrete state may change value to q_1. If this transition occurs, the continuous state gets reset to some value in $\mathcal{R}(q_0, q_1, x) \subseteq \mathcal{X}$ and continuous evolution resumes. The whole process is then repeated.

To formally define the solutions of a hybrid automaton, we first recall the definition of *hybrid time set* [3, 51]. Effectively equivalent notions of "super-dense" hybrid time can be found in [28, 52, 32].

Definition 2 (Hybrid time set): *A hybrid time set is a sequence of intervals $\tau = \{I_i\}_{i=0}^N$, finite ($N < \infty$) or infinite ($N = \infty$), such that*

- *$I_i = [\tau_i, \tau_i']$ for all $i < N$*
- *if $N < \infty$ then either $I_N = [\tau_N, \tau_N']$ or $I_N = [\tau_N, \tau_N')$*
- *$\tau_i \leq \tau_i' = \tau_{i+1}$ for all i*

Notice that the right endpoint, τ_i', of the interval I_i coincides with the left endpoint, τ_{i+1} of the interval I_{i+1} ($\tau_i' = \tau_{i+1}$). The interpretation is that these are the times at which discrete transitions of the hybrid system take place. Discrete transitions are instantaneous: τ_i' corresponds to the time instant just before a discrete transition, whereas τ_{i+1} corresponds to the time instant just after the discrete transition. This allows one to model situations where multiple discrete transitions take place one after the other at the same time instant, in which case $\tau_{i-1}' = \tau_i = \tau_i' = \tau_{i+1}$. Since all the elements in Definition 1 do not explicitly depend on time, we can take $\tau_0 = 0$ without loss of generality.

Definition 3 (Hybrid automaton execution): *An execution, (τ, q, x), of the hybrid automaton $H = (\mathcal{Q}, \mathcal{X}, \mathcal{F}, Init, Dom, \mathcal{E}, \mathcal{G}, \mathcal{R})$ comprises a hybrid time set $\tau = \{I_i\}_{i=0}^N$ and two sequences of functions, $q = \{q_i(\cdot)\}_{i=0}^N$ and $x = \{x_i(\cdot)\}_{i=0}^N$ with $q_i(\cdot) : I_i \to \mathcal{Q}$ and $x_i(\cdot) : I_i \to \mathcal{X}$, such that:*

- Initial condition*: $(q_0(\tau_0), x_0(\tau_0)) \in Init$*
- Discrete evolution*: For all i, $(q_i(\tau_i'), q_{i+1}(\tau_{i+1})) \in \mathcal{E}$, $x_i(\tau_i') \in \mathcal{G}(q_i(\tau_i'), q_{i+1}(\tau_{i+1}))$, and $x_{i+1}(\tau_{i+1}) \in \mathcal{R}(q_i(\tau_i'), q_{i+1}(\tau_{i+1}), x_i(\tau_i'))$*
- Continuous evolution*: For all i,*

  1) *$q_i(t) = q_i(\tau_i)$ for all $t \in I_i$*
  2) *$x_i(\cdot) : I_i \to \mathcal{X}$ is a solution to the differential inclusion*
  $$\frac{dx_i}{dt}(t) \in \mathcal{F}(q_i(t), x_i(t))$$
  over I_i starting at $x_i(\tau_i)$
  3) *for all $t \in [\tau_i, \tau_i')$, $x_i(t) \in Dom(q_i(t))$*

As expected several conditions need to be imposed on the elements of Definition 1 to ensure that executions in the sense of Definition 3 exist (see, for example, [3, 51, 32]).

Definition 3 imposes a number of restrictions on the state evolutions that the hybrid automaton can "accept". The first restriction dictates that the executions should start at an initial state in *Init*. The second restriction determines when discrete transitions can take place and what the state after discrete transitions can be. The state before the discrete transition $(q_i(\tau_i'), x_i(\tau_i'))$ and the state after the discrete transition $(q_{i+1}(\tau_{i+1}), x_{i+1}(\tau_{i+1}))$ should be such that $(q_i(\tau_i'), q_{i+1}(\tau_{i+1}))$ is an edge, $x_i(\tau_i')$ belongs to the guard of this edge and $x_{i+1}(\tau_{i+1})$ belongs the reset map of this edge. The guard $\mathcal{G}(e)$ can be regarded as an *enabling* condition for the discrete transition $e \in \mathcal{E}$ since *e may* be taken from a state x as long as $x \in \mathcal{G}(e)$. The third restriction determines what happens along the continuous evolution and when the continuous evolution must give way to a discrete transition. Along the continuous evolution the discrete state remains constant, whereas the continuous state flows according to the differential inclusion $\dot{x} \in \mathcal{F}(q,x)$ and must remain in the domain, $Dom(q)$, of the discrete state. Domain $Dom(q)$ can be regarded as *forcing* discrete transitions since a transition *must* be taken if the continuous state is about to leave the domain.

Considerable freedom is allowed when defining the solution in this "declarative" way. More specifically,

(i) There is a choice for the initial condition.
(ii) The direction of the continuous motion at any point in time is in general not unique; all directions in $\mathcal{F}(q,x)$ are allowed.
(iii) The mode after a discrete transition may not be uniquely defined. For example, from state (q,x) a transition to either mode q_1 or mode q_2 can be taken if $x \in \mathcal{G}(q,q_1) \cap \mathcal{G}(q,q_2)$.
(iv) The continuous state after a discrete transition may not be uniquely defined since the reset map is a set-valued map.
(v) There may be a choice between continuous evolution and a discrete transition. For example, if we have $x \in Dom(q) \cap \mathcal{G}(q,q')$, then from state (q,x) it may be possible either to evolve continuously in mode q, or to take a discrete transition to mode q'.

While all the choices listed above are possible, there is nothing to distinguish between them. Selecting among these choices leads to different system executions, but there is no implication that some of these executions are more likely than others. As a consequence, all analysis and control problems formulated for non-deterministic systems are of the "yes-or-no," "worst case" type. Such "yes-or-no" type questions include for example the questions "is a given subset of the hybrid state space an invariant set of the system?" (i.e., do all executions of the system always stay in this set) or "is a given subset of the hybrid state space a viable set?" (i.e., is it the case that for all initial conditions, there exist executions that stay in the set for ever), [3]. In the non-deterministic hybrid framework it is impossible to argue about how likely it is that the state will remain in a set; either it does, or it does not. SHS have been introduced precisely to relax this limitation.

2.2. Continuous-Time Stochastic Hybrid Systems

We now discuss how the "hard" choices outlined above can be relaxed by introducing probabilities.

Given a finite set $\mathcal{Q}$, $\mathcal{X} = \Re^n$ and a set valued map $Dom(\cdot) : \mathcal{Q} \to 2^{\mathcal{X}}$ assigning to each $q \in \mathcal{Q}$ an open subset of $\Re^n$, we define

$$S = \bigcup_{q \in \mathcal{Q}} \{q\} \times Dom(q) \subseteq \mathcal{Q} \times X.$$

Assuming that $\mathcal{Q}$ is endowed with the discrete topology (all subsets are open) one can see that the set S is an open subset of the topological space $\mathcal{Q} \times \mathcal{X}$. A metric (equivalent to the Euclidean metric on $Dom(q)$ for each q) can be defined such that S becomes a metric space [27]. We use $\mathcal{B}(S)$ to denote the Borel σ-algebra of S, i.e. the σ-algebra generated by all open subsets of S in this metric topology. Let also $\overline{S}$ denote the closure of the set S in $\mathcal{Q} \times \mathcal{X}$.

Definition 4 (Stochastic hybrid automaton): *A stochastic hybrid automaton is a collection $H = (\mathcal{Q}, \mathcal{X}, Dom, f, g, Init, \lambda, R)$ where*

- *$\mathcal{Q}$ is a finite set of discrete states*
- *$\mathcal{X} = \Re^n$ is a set of continuous states*
- *$Dom(\cdot) : \mathcal{Q} \to 2^{\mathcal{X}}$ is a set valued map assigning to each $q \in \mathcal{Q}$ an open subset of $\Re^n$*
- *$f : S \to \Re^n$ is a vector field*
- *$g : S \to \Re^{n \times m}$ is a diffusion coefficient*
- *$Init : \mathcal{B}(S) \to [0,1]$ is an initial probability measure on $(S, \mathcal{B}(S))$*
- *$\lambda : \underline{S} \to \Re_+$ is a transition rate function*
- *$R : \overline{S} \times \mathcal{B}(S) \to [0,1]$ is a transition measure*

The executions of a stochastic hybrid automaton can be informally defined as follows. Assume without loss of generality that the system evolution starts at time $t = 0$. An initial state $(q(0), x(0)) = (q_0, x_0)$ is extracted at random from S according to the probability measure $Init$:

$$P\{(q_0, x_0) \in A\} = Init(A) \text{ for all } A \in \mathcal{B}(S).$$

The continuous state then evolves from x_0 along a continuous solution of the stochastic differential equation

$$dx(t) = f(q_0, x(t))dt + g(q_0, x(t))dw(t), \tag{1}$$

where $w(t)$ denotes the standard m-dimensional Wiener process. In the meantime the discrete state remains constant: $q(t) = q_0$.

The discrete state can only change value when a discrete transition takes place. We distinguish between *forced transitions* and *spontaneous transitions* based on the way they occur:

1. Forced transitions take place on the boundary of S, when the continuous state $x(t)$ attempts to leave the set $Dom(q_0)$. Since $x(t)$ is assumed to be continuous as a function of time and the set $Dom(q_0)$ is assumed to be open such a transition will take place at the stopping time

$$T_f = \inf\{t \geq 0 \mid (q_0, x(t)) \notin S\}$$

for the diffusion process (1)

2. Spontaneous transitions take place in the interior of S as those of a Poisson arrival process, with a time varying rate $\lambda(q_0, x(t))$ that depends on the value of the continuous state $x(t)$.

Spontaneous transitions can be turned into an exit time condition by introducing an auxiliary continuous state variable $x_{n+1} \in \Re$ jointly evolving with x according to the differential equation

$$\frac{dx_{n+1}}{dt}(t) = \lambda(q_0, x(t))$$

from $x_{n+1}(0) = \ln(z_0)$ where z_0 is a random variable uniformly distributed in $[0, 1]$. Notice that $x_{n+1}(0) < 0$ and $x_{n+1}(t)$ is non-decreasing as a function of time (since by assumption $\lambda(q, x) \geq 0$ for all $(q, x) \in S$). Let

$$T_s = \inf\{t \geq 0 \mid x_{n+1}(t) \geq 0\}.$$

Under mild assumptions, this is a well defined stopping time. Moreover, for each solution $x(t)$ of the stochastic differential equation

$$P\{T_s > t\} = e^{-\int_0^t \lambda(q_0, x(s))ds}$$

as required. The next transition time can then be defined as

$$T_1 = \min\{T_f, T_s\},$$

which is again a well defined stopping time for the extended system on the extended state space $\hat{S} = S \times (-\infty, 0) \subseteq \mathcal{Q} \times \Re^{n+1}$.

Once the discrete transition (either forced or spontaneous) takes place at time T_1, the state variable x_{n+1} is reset to $\ln(z_1)$ with z_1 uniformly distributed in $[0, 1]$:

$$x_{n+1}(T_1) = \ln(z_1),$$

whereas the hybrid state (q, x) is reset according to the transition kernel R:

$$P\{(q(T_1), x(T_1)) \in A \mid q(T_1^-), x(T_1^-)\}$$
$$= \mathcal{R}(q(T_1^-), x(T_1^-), A) \quad \text{for all } A \in \mathcal{B}(S),$$

where $(q(T_1^-), x(T_1^-)) = \lim_{t \nearrow T_1}(q(t), x(t))$.

Once the state $(q(T_1), x(T_1))$ after the first discrete transition has been selected, the process is repeated to the second transition time T_2. Notice that the resulting stochastic process is defined over the usual time axis $t \in \Re_+$ and not over the "super-dense" time sets of Definition 2.

Clearly several assumptions need to be introduced on the elements of Definition 4 for the above intuitive discussion to make mathematical sense: Lipschitz continuity of the vector fields, measurability of the transition rates and transition kernels, etc. For details, the interested reader is referred to [17, 45].

Definition 5 (Stochastic hybrid automaton execution): *A stochastic process $(q(t), x(t))$ is called an execution of the stochastic hybrid automaton $H = (\mathcal{Q}, \mathcal{X}, Dom, f, g, Init, \lambda, R)$ if there exists a sequence of stopping times $T_0 = 0 < T_1 < T_2 \leq \ldots$ and an auxiliary stochastic process $x_{n+1}(t)$ such that*

- *$(q(T_0), x(T_0))$ is distributed according to the probability measure Init*

and for all $i = 0, 1, \ldots$

- *For all $t \in [T_i, T_{i+1})$, $q(t) = q(T_i)$, $x(t)$ is a continuous solution of the stochastic differential equation*

$$dx(t) = f(q(T_i), x(t))dt + g(q(T_i), x(t))dw(t)$$

starting at $x(T_i)$, and

$$x_{n+1}(t) = x_{n+1}(T_i) + \int_{T_i}^t \lambda(q(T_i), x(s))ds.$$

- $T_{i+1} = \inf\{t \geq T_i \mid (q(t), x(t), x_{n+1}(t)) \notin \hat{S}\}$
- *$(q(T_{i+1}), x(T_{i+1}))$ is distributed according to $\mathcal{R}(q(T_i), x(T_{i+1}^-), \cdot)$ and conditionally independent of all the other random variables at time $0 \leq t \leq T_{i+1}$ given $(q(T_i), x(T_{i+1}^-))$*
- *$e^{x_{n+1}(T_i)}$ is uniform in $[0, 1]$ and independent of all the other random variables at time $0 \leq t \leq T_i$*

Under relatively mild assumptions, it can be shown that this defines a strong Markov process on the hybrid state space S, whose sample paths are continuous from the right with left limits [17]. Moreover, using methods from stochastic analysis, existence and uniqueness conditions for this Markov process can be established [45].

The class of SHS defined above is very general and encompasses a very wide range of stochastic phenomena. Naturally, at this level of generality detailed analysis of the properties of a model is difficult, since more powerful modeling capabilities generally correspond to models that are more difficult to analyze. The situation gets even more complicated if input variables are considered in addition to the stochastic terms. Input variables allow for the introduction of control, or non-deterministic (as opposed to stochastic) disturbances, such as an adversary in a stochastic game. Similar to the stochastic terms, input variables can enter the continuous motion (continuous control) and the discrete transitions (transition rate control, forcing transition control, and continuous reset control). A modeling formalism that can accommodate both stochastic and non-deterministic features is needed in such a case. Here, we do not address this issue and only summarize in Table 1 the modeling choices made in some of the key references in the literature on continuous-time SHS. An indication of how input variables can be introduced is provided in Section 2.3 with reference to the discrete-time case.

Table 1. Overview of continuous time stochastic hybrid models

Characteristics	[27]	[31]	[11, 55]	[44]	[54, 60, 63]	[37]	[40]	[17, 45, 9]
Stochastic differential equation		√	√	√	√		√	√
Probabilistic resets	√		√	√		√	√	√
Spontaneous transitions	√	√			√	√		√
Forced transitions	√		√	√			√	√
Continuous control	√	√	√	√				
Transition rate control	√	√						
Forcing transition control	√		√					
Continuous reset control	√		√					

The problems one can study of SHS are of more variety and "shades" than those of non-deterministic hybrid systems, and the results obtained are often more robust and less conservative. As an example, a reachability problem in the deterministic case is a yes/no problem, while in the stochastic case one faces a continuous spectrum of "soft" problems with quantitative answers, such as the hitting probability, the expected hitting time, the hitting distribution, etc. Even though in some cases it is easy to infer certain properties of the system by intuition, the formal analysis of the resulting stochastic process is far from trivial. For example, in general, it is not even clear whether questions like "what is the probability that the state of the system will remain within a certain set for all $t \in [0, T]$?" are mathematically well posed: To ask this question one needs to ensure that the set of executions that have this property form an event (measurable set) in the underlying probability space. This is by no means obvious, since the definition of this set of executions involves taking an intersection over the uncountable set $[0, T]$. This turns out to be the case for several important classes of stochastic hybrid automata [18], though technical conditions need to be imposed. Moreover, even assuming that the questions are well posed mathematically, finding the answers is generally very difficult.

Over the years researchers have been able, however, to establish subclasses of SHS whose special structure allows the development of sophisticated analysis and control methods. Notable examples are represented by *piecewise deterministic Markov processes* and *switching diffusion processes*:

- Piecewise deterministic Markov processes [27] are characterized by a deterministic continuous evolution, since the stochastic differential equation is replaced by an ordinary differential equation. Rich random structure is instead preserved in the discrete transitions (both spontaneous and forced transitions are possible, as well as a random destination for the discrete and continuous states after the discrete transition).
- Switching diffusion processes [31] are characterized by $\sum_{q' \in \mathcal{Q}} \mathcal{R}(q, x, \{q', x\}) = 1$ for all $(q, x) \in \overline{S}$, and $Dom(q) = \mathcal{X}$ for all $q \in \mathcal{Q}$. The resulting stochastic process exhibits no forced transitions and is such that the continuous state is continuous as a function of time. This class of processes are also known as stochastic differential equations with Markovian switching [54] (often under the further assumption that $\lambda(q, x)$ is constant as a function of x for each $q \in \mathcal{Q}$).

In particular, optimal control methodologies have been developed for the models of [27, 31] and stability results for the models of [54]. Very little is instead known about these issues when it comes to the models of [17, 45].

Some of the technical difficulties associated with continuous-time SHS are not present in their discrete-time counterparts. Discrete-time SHS can actually be thought of as a special case of discrete-time, Markov chains on general state spaces and, hence, the powerful machinery that has been developed for discrete-time Markov chains in terms of tools for stability analysis, optimal control, etc. can be deployed. This is briefly illustrated in the next subsection.

2.3. Discrete-Time Stochastic Hybrid Systems

We conclude this section on stochastic hybrid models with a brief treatment of SHS in discrete-time. We define a discrete-time stochastic hybrid automaton (DTSHA for short) as the discrete-time counterpart of the continuous-time stochastic hybrid automaton model described in Section 2.2. Again this is only one of many possible ways one can think of discrete-time SHS; comparable modeling frameworks can be found for example in [10, 1].

The state of a DTSHA comprises a discrete component and a continuous component taking values in a finite set of modes $\mathcal{Q}$ and in $\mathcal{X} = \Re^n$, respectively. As before, the hybrid state space

$$S = \bigcup_{q \in \mathcal{Q}} \{q\} \times \mathcal{X}$$

can be turned into a metric space. We denote as $\mathcal{B}(S)$ the Borel σ-algebra generated by the corresponding open sets, and by $\mathcal{B}(\mathcal{X})$ the Borel σ-algebra of $\mathcal{X}$ with the standard Euclidean topology.

Definition 6 (Discrete-time stochastic hybrid automaton): *A discrete-time stochastic hybrid automaton is a collection* $H = (\mathcal{Q}, \mathcal{X}, U, \Sigma, Init, T_x, T_q, R)$, *where*

- $\mathcal{Q}$ *is a finite set of discrete states*
- $\mathcal{X} = \Re^n$ *is a set of continuous states*
- U *is a compact Borel space representing the transition control space*
- Σ *is a compact Borel space representing the reset control space*
- $Init : \mathcal{B}(S) \to [0,1]$ *is an initial probability measure on* $(S, \mathcal{B}(S))$
- $T_x : \mathcal{B}(\mathcal{X}) \times S \times U \to [0,1]$ *is a Borel measurable stochastic kernel on* $\mathcal{X}$ *given* $S \times U$, *which assigns to each* $(q,x) \in S$ *and* $u \in U$ *a probability measure* $T_x(\cdot|(q,x),u)$ *on* $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$
- $T_q : \mathcal{Q} \times S \times U \to [0,1]$ *is a discrete stochastic kernel on* $\mathcal{Q}$ *given* $S \times U$, *which assigns to each* $(q,x) \in S$ *and* $u \in U$, *a discrete probability distribution* $T_q(\cdot|(q,x),u)$ *over* $\mathcal{Q}$
- $R : \mathcal{B}(\mathcal{X}) \times S \times \Sigma \times \mathcal{Q} \to [0,1]$ *is a Borel measurable stochastic kernel on* $\mathcal{X}$ *given* $S \times \Sigma \times \mathcal{Q}$, *that assigns to each* $(q,x) \in S$, $\sigma \in \Sigma$, *and* $q' \in \mathcal{Q}$, *a probability measure* $R(\cdot|(q,x),\sigma,q')$ *on* $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$

The execution of a DTSHA can be defined algorithmically as follows.

Definition 7 (Discrete-time stochastic automaton execution): *A stochastic process* $\{q(k), x(k)\}_{k=0}^{N}$ *is called an execution of a* $H = (\mathcal{Q}, \mathcal{X}, U, \Sigma, Init, T_x, T_q, R)$ *over the horizon* $\{0, 1, \ldots, N\}$ *associated with an input sequence* $\{u_k, \sigma_k\}_{k=0}^{N-1}$ *if its sample paths are obtained according to the following algorithm:*

Extract a value $(q_0, x_0) \in S$ *for* $(q(0), x(0))$ *according to Init, and set* $k = 0$;
while $k < N$ *do*
  extract a value $q_{k+1} \in \mathcal{Q}$ *for* $q(k+1)$ *according to* $T_q(\cdot\,|(q_k, x_k)), u_k)$;
  if $q_{k+1} = q_k$, *then*
    extract a value $x_{k+1} \in \mathcal{X}$ *for* $x(k+1)$ *according to* $T_x(\cdot\,|(q_k, x_k), u_k)$;
  else
    extract a value $x_{k+1} \in \mathcal{X}$ *for* $x(k+1)$ *according to* $R(\cdot\,|(q_k, x_k), \sigma_k, q_{k+1})$;
  increment k;
end.

It is worth noticing that the DTSHA in Definition 7 allows one to capture in discrete-time the same qualitative features as continuous-time stochastic hybrid automata. In particular, it is easy to see that, by appropriately defining the discrete transition kernel T_q, it is possible to model both *forced transitions* that must occur when the continuous state exits some prescribed domain and *spontaneous transitions* that may occur during the continuous state evolution. For the forced transitions, the domain set $Dom(q)$ associated with mode $q \in \mathcal{Q}$ can be expressed in terms of T_q by setting $T_q(q|(q,x),u)$ equal to zero for all $x \notin Dom(q)$, irrespective of the value of the control input $u \in U$. Thus, while the system evolves in mode q, a jump from q to some $q' \neq q$ is forced as soon as $x \notin Dom(q)$. For the spontaneous transitions, if a discrete transition from q to $q' \neq q$ is enabled at $(q,x) \in S$ by the control input $u \in U$, then this can be encoded by the condition $T_q(q'|(q,x),u) > 0$ (cf. $\lambda(q,x) > 0$ for continuous-time stochastic hybrid automata).

We next show that a DTSHA can be expressed more compactly as a controlled Markov chain with state $s = (q,x)$ taking value in the hybrid state space S. To this purpose, we introduce a stochastic kernel $T : \mathcal{B}(\mathcal{X}) \times S \times U \times \Sigma \times \mathcal{Q} \to [0,1]$ on $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$ given $S \times U \times \Sigma \times \mathcal{Q}$ defined by

$$T(\cdot\,|(q,x),u,\sigma,q') = \begin{cases} T_x(\cdot|(q,x),u), & \text{if } q' = q \\ R(\cdot|(q,x),\sigma,q'), & \text{if } q' \neq q, \end{cases}$$

which can be used in the DTSHA algorithm to randomly select a value $x(k+1)$, given the values taken by $q(k)$, $x(k))$, $u(k)$, and $q(k+1)$. Based on T, the stochastic kernel $P : \mathcal{B}(S) \times S \times U \times \Sigma \to [0,1]$ on $(S, \mathcal{B}(S))$ given $S \times U \times \Sigma$ can be defined as

$$P((dx,q')\,|(q,x),(u,\sigma)) = T(dx\,|(q,x),u,\sigma,q')T_q(q'|(q,x),u), \tag{2}$$

Then, the DTSHA algorithm in Definition 7 can be rewritten as

Extract a value $s_0 \in S$ for $s(0)$ according to *Init*, and
set $k = 0$;
while $k < N$ do
  extract a value $s_{k+1} \in S$ for $s(k+1)$ according to $P(\cdot\,|s_k, u_k, \sigma_k)$;
  increment k;
end.

This shows that a DTSHA can be described as a controlled Markov process with state space S, control space $A = U \times \Sigma$, and controlled transition probability function $P : \mathcal{B}(S) \times S \times A \to [0,1]$ defined in (2). As a consequence, the extensive literature that has been developed over the years for discrete-time controlled Markov processes (see, for example, [56] and [12]) can be extended to address analysis and control problems for DTSHA. The reachability problem for DTSHA has been studied in [1] based on this approach.

3. Stochastic Hybrid Systems – Analysis and Control

Because of the fact that SHS involve the interaction of continuous and discrete phenomena, they have attracted the attention of three different communities. Part of the research in this area has been driven by computer scientists, in the area of formal methods, part by engineers, in the area of automatic control, and part by stochastic analysist. In this section we restrict our attention to methods that have solid theoretical foundations but also come with good computational support. In particular, we consider

1. Model checking methods.
2. Optimal control methods, including dynamic programming and model predictive control (MPC).
3. Randomized methods.

Model checking and randomized methods are by nature computational. As for optimal control methods, PDE solvers and Markov chain approximations can be used to numerically solve dynamic programming problems, and convex optimization can be used for MPC. The difficulty is that the computational tools supporting the different methods have different strengths and weaknesses (in terms of the size of the system they can handle, the classes of properties they can address, etc.) which limit their application to particular classes of problems.

Model checking methods: Model checking is a model-based verification technique that was first developed for establishing software correctness [6], [24]. In model checking, a given property is checked by exploring all possible executions of the system in a brute-force manner. To this purpose, an abstract model of the system that is just detailed enough to evaluate the property of interest is adopted. Models are typically nondeterministic finite-state automata, with a finite state space and a set of transitions describing how the system evolves from one state into another. Properties of interest are specified using temporal logics such as Computation Tree Logic (CTL) or Linear Temporal Logic (LTL). More recently, probabilistic model checking has been introduced as an automated technique for verifying properties specified in terms of probability that a certain condition is satisfied. The adopted stochastic models are discrete and continuous time Markov chains and the properties are specified in terms of some probabilistic logic like Probabilistic Computation Tree Logic (PCTL) [35]. Probabilistic model checking is based on conventional model checking, since it relies on reachability analysis of the underlying transition system, but also involves the propagation of probabilities through appropriate numerical methods. During the last decade, effective and efficient probabilistic model-checking algorithms have been developed [5] and implemented in software tools such as PRISM [38] and MRMC [43]. A detailed overview is provided in [6]. Model checking for stochastic systems with a hybrid state space is a rather unexplored field, with the notable exception of [65] which covers discrete time SHS. A first step in extending the results for model checking Markov chain to SHS has been taken recently with some work on time-inhomogeneous Markov chains, [34], [42], in which the probabilistic transitions are time-dependent. Time-inhomogeneous Markov chain is a very versatile class of models and is a natural stepping-stone towards more SHS such as piecewise deterministic Markov processes, [27], which present a quite similar probabilistic nature of mode transitions. A dynamic programming approach that allows one to encode PCTL properties through "value functions" was developed in [62].

Optimal control methods: Optimal control and game theoretical problems for stochastic (non-hybrid) systems with continuous state space can be solved by formally approximating the system with a controlled Markov chain, and then solving the optimal control problem for the approximating system, [47, 46]. Inspired by this approach, convergent Markov chain approximation schemes for optimal control and game theoretical problems have been recently proposed in [44] for SHS. [15, 31, 30] provide a systematic investigation of classes of optimal control problems over switching diffusions and, for the same modeling framework, [29] copes with stochastic differential games. For similar models, [16] has investigated optimal control issues over an infinite horizon by deriving variational inequalities, as well as by resorting to dynamic programming. Another seminal work is that of [27], which studies optimal control problems for piecewise deterministic Markov processes. An extension of the results in [27] to general stochastic hybrid systems has been proposed in [19]. In a number of instances the notion of reachability, a fundamental concept in systems and control theory, can be put in relationship with that of optimal control [36, 50]. This tenet has been leveraged to tackle optimal control for, respectively, discrete-time [1] and continuous-time [57] SHS. The first of the two works exploits the theory of dynamic programming and extends earlier results in [12], whereas the second contribution relies on the solution of Hamilton-Jacobi equations via level-set methods. The work in [10] proposes an optimization-based, chance-constrained optimal control problem applied to simple classes of stochastic hybrid models. This effort connects with the literature on (deterministic) MPC. This subject is formally extended to stochastic MPC in [25, 13, 58, 61, 39]. For a similar MPC problem randomized techniques are proposed in [53, 8] and a related randomized approach is applied in [48] to air traffic control. Given the state-of-the-art in optimal control techniques, a general optimal control theory for SHS has not been attained as of yet, in particular with regards to synthesis techniques that scaled well and can be used with high-dimensional models.

Randomized methods: During the last 15 years, the systems and control community has shown a growing interest in randomized methods for system analysis and design. The main reason for this interest is that these methods seem to be capable of overcoming traditional difficulties inherent in many deterministic approaches, especially in relation to computational issues, [21]. A quite straightforward application of the randomized approach is system verification, where, in order to certify that a system operates as desired, one estimates the (hopefully) negligible probability of the undesired behavior by running multiple executions of the system (Monte Carlo simulations), [64]. Performance analysis for a system affected by uncertainty is a further interesting application, the problem being that of evaluating if a certain performance level is guaranteed over most of the instances of the uncertain parameter, [4, 7, 68, 20]. Randomized methods have also been conceived for filtering and state estimation. Here the problem is determining the expected value of the state of a system based on some (partial) observations and on the knowledge of the initial distribution of the state. Analytic solutions to this problem exist only in very specific cases (Kalman filtering). The randomized numerical solution introduced in [33] is instead general, and consists in computing a weighted average of the state values obtained by considering different "particles" extracted at random from the initial state distribution, associating to each particle the same weight, making each particle evolve through the system dynamics, and rescaling its weight every time an observation becomes available based on its likelihood given that observation (particle filtering). Asymptotic convergence of the Monte Carlo estimate to the true value can be proven as the number of particles grows to infinity [26]. A few tutorials on particle filtering have been published quite recently (see e.g. [2] and [22]). An issue encountered in system verification is that the number of simulations needed to obtain a meaningful estimate of a rare event probability can be quite high and indeed prohibitive for a large-scale system. As for filtering, the initially generated particles can progressively "die" in that their weights can get lower and lower as more observations become available. Regeneration and importance sampling techniques are then used to solve this issue. Ad-hoc solutions to both these issues have been proposed for some classes of SHS in e.g. [69], [70], [71]. A more recent application area for randomized methods that remains only partially explored as for SHS is system design, and, more precisely, the problem of choosing the design parameters of a system affected by uncertainty so as to optimize its performance. Two main approaches to this problem have been proposed in the literature: i) maximization of the average performance (robustness on average); ii) maximization of the performance over the set of all uncertainty instances except for an user-chosen fraction (chance-constrained solution). Since exact analytic solutions are hard – if not impossible – to find, one has to head for approximate numerical solutions. The randomized solution to i) proposed in the seminal paper [72] is based on the notion of uniform convergence of the empirical mean and on results taken from the statistical learning theory. The solution in [73] is instead based on more basic concepts of probability theory, but comes with no accuracy guarantees. Both these solutions have been adopted in the literature, see e.g. [74], [48]. The randomized approach to the chance-constrained problem ii) proposed in [75] is quite computationally effective and amenable for various applications, [76], since it also allows one to include various constraints, but it is applicable only to problems that are convex in the design parameters. Further randomized approaches to system analysis and design, including sample generation, are described in the recent book [77].

4. Conclusions and Future Directions of Research

From the discussion in Section 2, it should be apparent that SHS offer an ideal framework for capturing the intricacies and complex, heterogeneous dynamics one encounters in large scale applications. The picture that emerges from the brief survey in Section 3, however, suggests that none of the available methods for SHS analysis and control are powerful enough for dealing with realistic scale problems that arise in application areas such as power networks, biochemical networks, telecommunication, air traffic management, etc.

To allow for such a large scale, complex and networked application, we envision two fundamental research directions:

1. Development of a modeling framework for SHS that supports composition and abstraction operations.
2. Cross-fertilization of computational methods to extend the domain of application of the current computational tools to larger and more complex systems.

Composition and abstraction form the basis of scalable, hierarchical modeling, analysis and control methodologies that enable one to deal with large scale, networked systems. Roughly speaking, rather than dealing with a large scale system as a single monolithic block, composition and abstraction operations allow one to decompose the overall system and the properties of interest into a hierarchy of interconnected subsystems, each with its own desirable properties. The properties of each subsystem are then analyzed/synthesized separately, a task which is generally much simpler than the corresponding task for the overall system. The composition and abstraction framework provides the support necessary so that following this divide-and-conquer analysis/design phase one can subsequently draw conclusions about the properties of the overall system. Methods based on composition, abstraction and hierarchical reasoning have been developed over the years for purely discrete systems. This effort was motivated to a large extent by applications to communication and computer networks, ad-hoc networking, etc, where a decomposition of the overall system into subsystems is natural. Such methods, however, are incapable of dealing with applications such as power networks. In this case the decomposition of the overall system into subsystems is still natural, but the subsystems themselves and the coupling between them involve continuous dynamics (for example, the fluctuation of voltages and frequencies in the network) which play a crucial role (for example, determine whether load shedding or undesirable power oscillations will take place). Problems like these (and problems in transportation and other domains with similar features) have motivated attempts to develop composition and abstraction methods for purely continuous systems and for hybrid systems. These methods are currently in various stages of maturity. None of them, however, is capable of dealing with the third ingredient of SHS, namely probabilistic uncertainty. Even though some attempts in this direction have been reported in the literature [67, 66, 41], to the best of our knowledge none of the currently available approaches even comes close to dealing with the types of problems one encounters in large scale networked systems. Novel, more general and more powerful methods are needed to alleviate this shortcoming.

As for the second direction of research, none of the currently existing methods is powerful enough to address the problems that arise in large scale systems on its own. All of the existing methods have their strengths and weaknesses. Model checking methods, for example, can deal with very complex properties encoded in temporal logic and are supported by efficient computational tools. They cannot, however, deal directly with continuous state spaces and do not provide a framework in which feedback controllers can be designed to, for example, maximize the probability that a certain desirable property is satisfied; both of these features are necessary for large scale control applications. Optimal control methods readily provide a framework for designing controllers in a continuous or even hybrid state setting. They do not, however, lend themselves to the investigation of complex properties; only plain reachability questions can be addressed at the moment, and even these only recently. Moreover, the only computational methods available for dealing with optimal control problems with continuous and hybrid states rely on numerical gridding of the state space, which suffers from the curse of dimensionality and is therefore not applicable to large scale systems. Randomized methods should in principle be able to alleviate this difficulty somewhat, but their use and convergence properties for systems with hybrid state space are not fully understood in the currently available theory.

Novel stochastic hybrid analysis and control methods should be developed by establishing links between the different methods and taking advantage of their mutual strengths. As an example, dynamic programming interpretation of temporal logic formulas that also applies to systems with continuous states could be conceived. Numerical computation based on gridding and finite approximation can then be used in conjunction with model checking tools to study very complex properties for SHS. To circumvent the curse of dimensionality (and hence extend the method to large scale systems), randomized methods will then be used to obtain a more efficient (albeit approximate) solution to the problem. This combination of methods, which can be thought of as an approximate dynamic programming model checking framework for SHS, will be applicable to systems that are much larger and more complex than randomized methods, dynamic programming methods and model checking methods can deal with by themselves.

Acknowledgments

Research supported by the European Commission under project *iFly* FP6-TREN-037180, and by MIUR (Ministero dell'Istruzione, dell'Università e della Ricerca) under the project *Identification and adaptive control of industrial systems*.

References

1. Abate A, Prandini M, Lygeros J, Sastry S. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica* 44(11): 2724–2734, 2008.
2. Arulampalam S, Maskell S, Gordon N, Clapp T. A tutorial on particle filters for on-line nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans Signal Process* 50(2): 174–188, 2002.
3. Aubin J-P, Lygeros J, Quincampoix M, Sastry SS, Seube N. Impulse differential inclusions: A viability approach to hybrid systems. *IEEE Trans Autom Control* 47(1): 2–20, 2002.

4. Bai EW, Tempo M, Fu M. Worst-case properties of the uniform distribution and randomized algorithms for robustness analysis. *Math Control Signal Syst* 11: 183–196, 1998.
5. Baier C, Haverkort BR, Hermanns H, Katoen J-P. Model-checking algorithms for continuous-time Markov Chains. *IEEE Trans Softw Eng* 29(6): 524–541, 2003.
6. Baier C, Katoen J-P. *Principles of Model Checking*. MIT Press, 2008.
7. Barmish BR, Lagoa CM. The uniform distribution: a rigorous justification for its use in robustness analysis. *Math Control Signal Syst* 10: 203–222, 1997.
8. Batina I. *Model predictive control for stochastic hybrid systems by randomized algorithms*. PhD thesis, Technische Universiteit Eindhoven, 2004.
9. Bect J. A unifying formulation of the Fokker-Planck-Kolmogorov equation for general stochastic hybrid systems. *Nonlinear Anal Hybrid Syst* 4(2): 357–370, 2010.
10. Bemporad A, Di Cairano S. Optimal control of discrete hybrid stochastic automata. In Thiele L, Morari M, editors, *Hybrid Systems: Computation and Control*, number 3414 in LNCS, pages 151–167. Springer-Verlag, Berlin, 2005.
11. Bensoussan A, Menaldi JL. Stochastic hybrid control. *J Math Anal Appl* 249: 261–288, 2000.
12. Bertsekas DP, Shreve SE. *Stochastic optimal control: The discrete time case*. Athena Scientific, Belmont, Massachusetts, 1996.
13. Bertsimas D, Brown DB. Constrained stochastic LQG: a tractable approach. *IEEE Trans Autom Control* 52(10): 1826–1841, 2007.
14. Blom HAP, Lygeros J, editors. *Stochastic Hybrid Systems: Theory and Safety Critical Applications*. Number 337 in Lecture Notes in Control and Information Sciences. Springer-Verlag, Berlin, 2006.
15. Borkar VS. *Optimal Control of Diffusion Processes*, volume 203 of *Pitman Research Notes in Math*. Longman Scientific and Technical, Harlow, UK, 1989.
16. Borkar VS, Ghosh MK, Sahay P. Optimal control of a stochastic hybrid system with discounted cost. *J Optim Theory Appl* 101(3): 557–580, 1999.
17. Bujorianu M, Lygeros J. Toward a general theory of stochastic hybrid systems. In Blom HAP, Lygeros J, editors, *Stochastic Hybrid Systems: Theory and Safety Critical Applications*, number 337 in LNCIS, pages 3–30. Springer-Verlag, Berlin, 2006.
18. Bujorianu ML, Lygeros J. Reachability questions in piecewise deterministic Markov processes. In Maler O, Pnueli A, editors, *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science 2623, pages 126–140. Springer Verlag, 2003.
19. Bujorianu ML, Lygeros J. General stochastic hybrid systems: Modeling and optimal control. In *Proc. 43rd IEEE Conf. on Decision and Control*. Atlantis, Paradise Island, Bahamas, 2004.
20. Calafiore G, Dabbene F, Tempo R. Randomized algorithms for probabilistic robustness with real and complex structured uncertainty. *IEEE Trans Autom Control* 45: 2218–2235, 2000.
21. Calafiore G, Dabbene F, Tempo R. A survey of randomized algorithms for control synthesis and performance verification. *J Complexity* 23(3): 301–316, 2007.
22. Cappe O, Godsill SJ, Moulines E. An overview of existing methods and recent advances in sequential Monte Carlo. *IEEE Proc* 95(5): 899–924, 2007.
23. Cassandras CG, Lygeros J, editors. *Stochastic Hybrid Systems*. Number 24 in Control Engineering. CRC Press, Boca Raton, 2006.
24. Clarke EM, Grumberg O, Peled D. *Model Checking*. MIT Press, 1999.
25. Couchmann PD, Cannon M, Kouvaritakis B. Stochastic MPC with inequality stability constraints. *Automatica* 42(12): 2169–2174, 2006.
26. Crisan D, Doucet A. A survey of convergence results on particle filtering for practitioners. *IEEE Trans Signal Process* 50(3): 736–746, 2002.
27. Davis MHA. *Markov Processes and Optimization*. Chapman & Hall, London, 1993.
28. Deshpande A. *Control of Hybrid Systems*. PhD thesis, Department of Electrical Engineering, University of California, Berkeley, 1994.
29. Ghosh MK, Marcus SI. Stochastic differential games with multiple modes. Technical report. The Institute for Systems Research, University of Maryland, 1995.
30. Ghosh MK, Arapostathis A, Marcus SI. Optimal control of switching diffusions with ap- plication to flexible manufacturing systems. *SIAM J Control Optim* 30(6): 1–23, 1992.
31. Ghosh MK, Arapostathis A, Marcus SI. Ergodic control of switching diffusions. *SIAM J Control Optim* 35(6): 1952–1988, 1997.
32. Goebel R, Teel A. Solutions to hybrid inclusions via set and graphical convergence with stability theory applications. *Automatica* 42(4): 573–587, 2006.
33. Gordon NJ, Salmond DJ, Smith AF. Novel approach to non-linear non-Gaussian Bayesian state estimation. *IEE Proc F* 140(2): 107–113, 1993.
34. Han T, Katoen J-P, Mereacre A. Compositional modeling and minimization of time-inhomogeneous Markov chains. In *Hybrid Systems: Computation and Control*, volume 4981 of *LNCS*, pages 244–258. Springer, 2008.
35. Hansson H, Jonsson B. A logic for reasoning about time and reliability. *Form Asp Comput* 6(5): 512–535, 1994.
36. Hedlund S, Rantzer A. Convex dynamic programming for hybrid systems. *IEEE Trans Autom Control* 47(9): 1536–1540, 2002.
37. Hespanha J. A model for stochastic hybrid systems with application to communication networks. *Nonlinear Anal* 62(8): 1353–1383, 2005.
38. Hinton A, Kwiatkowska M, Norman G, Parker D. PRISM: A tool for automatic verification of probabilistic systems. In *Tools and Algorithms for the Construction and Analysis of Systems*, volume 3920 of *LNCS*, pages 441–444. Springer, 2006.
39. Hokayem P, Chatterjee D, Lygeros J. On stochastic model predictive control with bounded control inputs. In *Proc. 48th IEEE Conf. on Decision and Control*, Shanghai, China, 2009.
40. Hu J, Lygeros J, Sastry SS. Towards a theory of stochastic hybrid systems. In Lynch N, Krogh BH, editors, *Hybrid Systems: Computation and Control*, number 1790 in LNCS, pages 160–173. Springer-Verlag, Berlin, 2000.
41. Julius AA, Pappas GJ. Approximations of stochastic hybrid systems. *IEEE Trans Autom Control* 54(6): 1193–1203, 2009.
42. Katoen J-P, Mereacre A. Model checking HML on piecewise-constant inhomogeneous Markov chains. In *Formal Modeling and Analysis of Timed Systems*, volume 5215 of *LNCS*, pages 203–217. Springer, 2008.

43. Katoen J-P, Zapreev IS, Hahn EM, Hermanns H, Jansen DN. The ins and outs of the probabilistic model checker MRMC In *Conference on Quantitative Evaluation of Systems*, pages 167–176. IEEE Computer Society, 2009.
44. Koutsoukos XD. Optimal control of stochastic hybrid systems based on locally consistent Markov decision processes. *Int J Hybrid Syst* 4: 301–318, 2004.
45. Krystul J, Blom HAP, Bagchi A. Stochastic differential equations on hybrid state spaces. In Cassandras CG, Lygeros J, editors, *Stochastic Hybrid Systems*, number 24 in Control Engineering, pages 15–45. CRC Press, Boca Raton, 2006.
46. Kushner HJ. Numerical approximations for stochastic differential games. *SIAM J Control Optim* 41(2): 457–486, 2002.
47. Kushner HJ, Dupuis PG. *Numerical Methods for Stochastic Control Problems in Continuous Time.* Springer-Verlag, New York, 2001.
48. Lecchini A, Glover W, Lygeros J, Maciejowski J. Monte Carlo optimization for conflict resolution in air traffic control. *IEEE Trans Intell Transp Syst* 7: 470–482, 2006.
49. Lunze J, Lamnabhi-Lagarrigue F, editors. *Handbook of Hybrid Systems Control, Theory Tools Applications.* Cambridge University Press, Cambridge, 2009.
50. Lygeros J. On reachability and minimum cost optimal control. *Automatica* 40(6): 917–927, 2004.
51. Lygeros J, Johansson KH, Simić SN, Zhang J, Sastry SS. Dynamical properties of hybrid automata. *IEEE Trans Autom Control* 48(1): 2–17, 2003.
52. Lynch N, Segala R, Vaandrager F. Hybrid I/O automata. *Inf Comput* 185(1): 105–157, 2003.
53. Maciejowski JM, Lecchini-Visintini A, Lygeros J. Nmpc for complex stochastic systems using Markov Chain Monte Carlo. In *Assessment and Future Directions of Nonlinear Model Predictive Control*, volume 358 of *LNCIS*, pages 269–281. Springer, 2007.
54. Mao X, Yuan C. *Stochastic differential equations with Markovian switching*. Imperial College Press, 2006.
55. Menaldi JL. Stochastic hybrid optimal control models. *Aportaciones Matematicas (Sociedad Matematica Mexicana)* 16: 205–250, 2001.
56. Meyn SP, Tweedie RL. *Markov Chains and Stochastic Stability*. Online edition: http://probability.ca/MT/, 2005. First edition: Springer, London, 1993.
57. Mitchell I, Templeton J. A toolbox of Hamilton-Jacobi solvers for analysis of nondeterministic continuous and hybrid systems. In *Hybrid Systems: Computation and Control*, volume 3414 of *LNCS*, pages 480–494. Springer, 2005.
58. Oldewurtel F, Jones CN, Morari M. A tractable approximation of chance constrained stochastic MPC based on affine disturbance feedback. In *Proc. 47th IEEE Conf. on Decision and Control*, Cancun, Mexico, 2008.
59. Pola G, Bujorianu ML, Lygeros J, Di Benedetto MD. Stochastic hybrid models: An overview with applications to air traffic management. In *Proceedings of the 2003 IFAC Conference on Analysis and Design of Hybrid Systems (ADHS'03)*, pages 45–50, June 2003.
60. Prandini M, Hu J. Stochastic reachability: Theory and numerical approximation. In Cassandras CG, Lygeros J, editors, *Stochastic Hybrid Systems*, Automation and Control Engineering Series 24, pages 107–138. Taylor & Francis Group/CRC Press, 2006.
61. Primbs J. A soft constraint approach to stochastic receding horizon control. In *Proc. 46th IEEE Conf. on Decision and Control*, 2007.
62. Ramponi F, Chatterjee D, Summers S, Lygeros J. On the connections between PCTL and dynamic programming. In *Hybrid Systems: Computation and Control*, page 253262. ACM, 2010.
63. Riley D, Koutsoukos X, Riley K. Modeling and simulation of biochemical processes using stochastic hybrid systems: The sugar cataract development process. In *Hybrid Systems: Computation and Control*, volume 4981 of *LNCS*, page 429442. Springer, 2008.
64. Rubino G, Tuffin B, editors. *Rare Event Simulation using Monte Carlo Methods*. John Wiley & Sons, 2009.
65. Sproston J. Decidable model checking of probabilistic hybrid automata. In *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 1926 of *LNCS*, pages 501–514. Springer, 2000.
66. Strubbe S, van der Schaft A. Compositional modelling of stochastic hybrid systems. In Cassandras CG, Lygeros J, editors, *Stochastic Hybrid Systems*, Automation and Control Engineering Series 24, pages 47–78. Taylor & Francis Group/CRC Press, 2006.
67. Strubbe SN. *Compositional Modelling of Stochastic Hybrid Systems*. PhD thesis, Twente University, 2005.
68. Tempo R, Bai E-W, Dabbene F. Probabilistic robustness analysis: explicit bounds for the minimum number of samples. *Syst Control Lett* 30: 237–242, 1997.
69. Blom HA, Krystul J, Bakker GJ. Free flight collision risk estimation by sequential MC simulation. In Cassandras CG, Lygeros J, editors, *Stochastic Hybrid Systems, Automation and Control Engineering Series* 24, pages 247–279. Taylor & Francis Group/CRC Press, 2006.
70. Andrieu C, Davy M, Doucet A. Efficient particle filtering for jump Markov systems. Application to time-varying autoregressions. *IEEE Trans Signal Process* 2003; 49: 1762–1770.
71. Blom HA, Bloem EA. Exact Bayesian and particle filtering of stochastic hybrid systems. *IEEE Trans Aerosp Electron Syst* 2007; 43: 55–70.
72. Vidyasagar M. Randomized algorithms for robust controller synthesis using statistical learning theory. *Automatica* 2001; 37: 1515–1528.
73. Muller P. Simulation based optimal design. In Berger J, Bernardo O, David A, Smith A, editors, *Bayesian Statistics 6*, pages 459–474. London, UK: Oxford University Press, 1999.
74. Campi MC, Prandini M. Randomized algorithms for the synthesis of cautious adaptive controllers. *Syst Control Lett* 2003; 49: 21–36.
75. Calafiore G, Campi MC. The scenario approach to robust control design. *IEEE Trans Autom Control* 2006; 51: 742–753.
76. Campi MC, Garatti S, Prandini M. The scenario approach for systems and control design. *Annu Rev. Control* 2009; 33: 149–157.
77. Tempo R, Calafiore G, Dabbene F. Randomized algorithms for analysis and control of uncertain systems. London, UK.: Springer-Verlag, 2005.