Proceedings of the
47th IEEE Conference on Decision and Control
Cancun, Mexico, Dec. 9-11, 2008

An approximate dynamic programming approach to probabilistic reachability for stochastic hybrid systems

Alessandro Abate, Maria Prandini, John Lygeros, and Shankar Sastry

Abstract— **This paper addresses the computational overhead involved in probabilistic reachability computations for a general class of controlled stochastic hybrid systems. An approximate dynamic programming approach is proposed to mitigate the curse of dimensionality issue arising in the solution to the stochastic optimal control reformulation of the probabilistic reachability problem. An algorithm tailored to this problem is introduced and compared with the standard numerical solution to dynamic programming on a benchmark example.**

I. Introduction

Stochastic Hybrid Systems (SHS) are a general class of models relevant to a wide range of application contexts involving interacting discrete and continuous dynamics, as well as probabilistic uncertainty, [5], [7].

In this paper we study the *reachability* problem for SHS. Reachability is an important topic in systems theory. Qualitatively, it deals with the issue of evaluating whether the state of a system will reach a certain set during a time interval, starting from some initial conditions, and possibly subject to a control input. If such a set represents an unsafe region of the state space, one is dealing with a safety problem and has the choice to select the control input so as to avoid entering that set. In a stochastic context, it appears natural to interpret the reachability concept in probabilistic terms, and to investigate problems such as quantifying the probability of reaching a set or minimizing the probability of entering an unsafe set. If viable, this characterization appears richer and in many cases preferable to a worst case viewpoint, where one considers each admissible trajectory, neglecting how much likely it is to occur.

Recently, a number of contributions on probabilistic reachability of SHS have appeared in the literature, see for example [2], [6], [11], [13], [14]. However, reachability computations for SHS of practical scale remains a challenging open problem. In this paper we study this issue for the general class of discrete-time controlled SHS in [2]. We focus on a safety problem where the objective is to determine the control policy that maximizes the probability of remaining within a given safe set during a finite time horizon. In [2], such a safety problem has been reformulated as a stochastic optimal control problem with multiplicative cost for a controlled Markov chain, to which dynamic programming can be applied. Given that the solution to the value iteration equations obtained through the dynamic programming approach cannot be written out explicitly, the safety problem has to be solved in practice through some approximation method.

The work in [1] has shown that, under appropriate continuity assumptions on the transition probabilities that characterize the SHS dynamics, the numerical solution obtained by a standard gridding scheme is asymptotically convergent as the gridding scale parameter goes to zero. Non-asymptotic error bounds can also be given. On the other hand, the overwhelming computational burden associated with this approach makes it inapplicable in realistic situations. For problems of practical scale, storing and manipulating functions over the discretized hybrid state space becomes prohibitive, and some approximation scheme is needed. Here, we investigate and discuss an approximate value iteration algorithm that relies on a neural approximation of the value function to mitigate the curse of dimensionality. A comparative study between this technique and the grid-based numerical approximation is presented on a benchmark example.

II. Discrete-Time Stochastic Hybrid Model

The state of the controlled discrete-time stochastic hybrid system (DTSHS) model introduced in [2] is characterized by two components: a discrete and a continuous one. The continuous state evolves according to a probabilistic law that depends on the value taken by the discrete state. The discrete state can transition between different values in a finite set according to some probabilistic law that depends on the continuous state. Both the continuous and the discrete probabilistic evolutions can be affected by some control input (transition input). Furthermore, whenever a transition in the discrete state occurs, the continuous state is subject to a probabilistic reset that may depend on an additional control input (reset input).

***Definition** 1 (DTSHS):* A discrete-time stochastic hybrid system is a tuple $\mathcal{H} = (\mathcal{Q}, n, \mathcal{U}, \Sigma, T, T_q, R)$, where

- $\mathcal{Q} := \{q_1, q_2, \ldots, q_m\}, m \in \mathbb{N}$, represents the discrete state space
- $n : \mathcal{Q} \to \mathbb{N}$ assigns to each discrete state value $q \in \mathcal{Q}$ the dimension of the continuous state space $\mathbb{R}^{n(q)}$. The hybrid state space is then $\mathcal{S} := \cup_{q \in \mathcal{Q}} \{q\} \times \mathbb{R}^{n(q)}$
- $\mathcal{U}$ is a Borel space denoting the transition control space
- Σ is a Borel space representing the reset control space
- $T : \mathcal{B}(\mathbb{R}^{n(\cdot)}) \times \mathcal{S} \times \mathcal{U} \to [0, 1]$ is a Borel-measurable stochastic kernel on $\mathbb{R}^{n(\cdot)}$ given $\mathcal{S} \times \mathcal{U}$, which assigns to each $s = (q, x) \in \mathcal{S}$ and $u \in \mathcal{U}$ a probability measure $T(dx|s, u)$ on the Borel space $(\mathbb{R}^{n(q)}, \mathcal{B}(\mathbb{R}^{n(q)}))$

This work was partially supported by MIUR under the project "New methods for Identification and Adaptive Control for Industrial Systems," by the EC under the project iFly TREN/07/FP6AE/S07.71574/037180, and by the NSF grant CCR-0225610.

A. Abate is with Stanford University aabate@stanford.edu; M. Prandini is with the Politecnico di Milano prandini@elet.polimi.it; J. Lygeros is with ETH Zurich lygeros@control.ee.ethz.ch; S. Sastry is with the University of California, Berkeley sastry@eecs.berkeley.edu.

- $T_q : \mathcal{Q} \times \mathcal{S} \times \mathcal{U} \to [0,1]$ is a discrete stochastic kernel on $\mathcal{Q}$ given $\mathcal{S} \times \mathcal{U}$, which assigns to each $s \in \mathcal{S}$ and $u \in \mathcal{U}$, a probability distribution $T_q(q|s,u)$ over $\mathcal{Q}$
- $R : \mathcal{B}(\mathbb{R}^{n(\cdot)}) \times \mathcal{S} \times \Sigma \times \mathcal{Q} \to [0,1]$ is a Borel-measurable stochastic kernel on $\mathbb{R}^{n(\cdot)}$ given $\mathcal{S} \times \Sigma \times \mathcal{Q}$, that assigns to each $s \in \mathcal{S}$, $\sigma \in \Sigma$, and $q' \in \mathcal{Q}$, a probability measure $R(dx|s,\sigma,q')$ on $(\mathbb{R}^{n(q')}, \mathcal{B}(\mathbb{R}^{n(q')}))$. □

In order to define an execution for a DTSHS we have to specify how the system is initialized and how the control inputs to the system are selected.

The system initialization at the initial time $k = 0$ is specified through a probability measure $\pi_0 : \mathcal{B}(\mathcal{S}) \to [0,1]$ on the Borel space $(\mathcal{S}, \mathcal{B}(\mathcal{S}))$, where $\mathcal{B}(\mathcal{S})$ is the σ-field generated by the subsets of $\mathcal{S}$ of the form $\cup_q \{q\} \times C_q$, with C_q denoting a Borel set in $\mathbb{R}^{n(q)}$. As for the inputs, we consider the case where the control inputs are selected based on the current value of the hybrid state according to a Markov policy [2]. A Markov policy for $\mathcal{H}$ is a sequence $\mu = (\mu_0, \mu_1, \ldots, \mu_{N-1})$ of universally measurable maps [3] $\mu_k : \mathcal{S} \to \mathcal{U} \times \Sigma$, $k = 0, 1, \ldots, N-1$. We denote the set of Markov policies as $\mathcal{M}_m$.

Let $\tau_x : \mathcal{B}(\mathbb{R}^{n(\cdot)}) \times \mathcal{S} \times \mathcal{U} \times \Sigma \times \mathcal{Q} \to [0,1]$ be a stochastic kernel on $\mathbb{R}^{n(\cdot)}$ given $\mathcal{S} \times \mathcal{U} \times \Sigma \times \mathcal{Q}$, which assigns to each $s = (q,x) \in \mathcal{S}$, $u \in \mathcal{U}$, $\sigma \in \Sigma$, and $q' \in \mathcal{Q}$ a probability measure on the Borel space $(\mathbb{R}^{n(q')}, \mathcal{B}(\mathbb{R}^{n(q')}))$ as follows:

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

Based on τ_x we can introduce the Borel-measurable stochastic kernel $T_s : \mathcal{B}(\mathcal{S}) \times \mathcal{S} \times \mathcal{U} \times \Sigma \to [0,1]$ on $\mathcal{S}$ given $\mathcal{S} \times \mathcal{U} \times \Sigma$, which assigns to each $s = (q,x) \in \mathcal{S}$ and $(u,\sigma) \in \mathcal{U} \times \Sigma$ a probability measure on the Borel space $(\mathcal{S}, \mathcal{B}(\mathcal{S}))$ as follows:

$$T_s(ds' \,|s,(u,\sigma)) = \tau_x(dx' \,|s,u,\sigma,q') T_q(q'|s,u), \qquad (1)$$

$s, s' = (q', x') \in \mathcal{S}$, $(u,\sigma) \in \mathcal{U} \times \Sigma$.

***Definition** 2 (Execution):* An execution for a DTSHS $\mathcal{H} = (\mathcal{Q}, n, \mathcal{U}, \Sigma, T, T_q, R)$ associated with a Markov policy $\mu = (\mu_0, \mu_1, \ldots, \mu_{N-1}) \in \mathcal{M}_m$ and an initial distribution π_0 is an $\mathcal{S}$-valued stochastic process $\{\mathbf{s}(k), k \in [0,N]\}$, whose sample paths are obtained according to the following algorithm, where all the random extractions considered are mutually independent:

extract from $\mathcal{S}$ a value s_0 for $\mathbf{s}(0)$ according to π_0;

for $k = 0$ to $N-1$

set $(u_k, \sigma_k) = \mu_k(s_k)$;

extract from $\mathcal{S}$ a value s_{k+1} for $\mathbf{s}(k+1)$ according to $T_s(\cdot\,|s_k,(u_k,\sigma_k))$;

end □

A DTSHS $\mathcal{H}$ can then be described as a controlled Markov process with state space $\mathcal{S}$, control space $\mathcal{U} \times \Sigma$, and controlled transition probability function T_s defined in (1). Thus, the execution $\{\mathbf{s}(k), k \in [0,N]\}$ associated with a specific $\mu \in \mathcal{M}_m$ and a probability π_0 is a time inhomogeneous Markov process defined on the canonical sample space $\Omega = \mathcal{S}^N$, endowed with its product topology $\mathcal{B}(\Omega)$, with probability measure $P^\mu_{\pi_0}$ uniquely defined by the transition kernel T_s, the policy $\mu \in \mathcal{M}_m$, and the initial probability measure π_0 (see [3, Proposition 7.45]). When π_0 is concentrated on a point $s \in \mathcal{S}$, that is $\pi_0(ds) = \delta_s(ds)$, we shall write simply P^μ_s.

III. Probabilistic Reachability

We start by considering the following reachability problem: given a stochastic hybrid system $\mathcal{H}$, determine the probability that the execution associated with some Markov policy $\mu \in \mathcal{M}_m$ and initialization π_0 will remain in a Borel set $A \in \mathcal{B}(\mathcal{S})$ during the whole time horizon $[0,N]$:

$$p^\mu_{\pi_0}(A) := P^\mu_{\pi_0}(\mathbf{s}(k) \in A \text{ for all } k \in [0,N]). \qquad (2)$$

If π_0 is concentrated on $s \in \mathcal{S}$, we use the notation $p^\mu_s(A)$. If set $\bar{A} = \mathcal{S} \setminus A$ represents an unsafe set for $\mathcal{H}$, by computing $p^\mu_{s_0}(A)$, we shall evaluate the safety level for system $\mathcal{H}$ when it starts from s_0 and is subject to the policy $\mu \in \mathcal{M}_m$. Observe that

$$\prod_{k=0}^{N} \mathbf{1}_A(s_k) = \begin{cases} 1, & \text{if } s_k \in A \text{ for all } k \in [0,N] \\ 0, & \text{otherwise,} \end{cases}$$

where $s_k \in \mathcal{S}$, $k \in [0,N]$ and $\mathbf{1}_A : \mathcal{S} \to \{0,1\}$ denotes the indicator function of set A. Then,

$$p^\mu_{\pi_0}(A) = E^\mu_{\pi_0}\left[\prod_{k=0}^{N} \mathbf{1}_A(\mathbf{s}(k))\right],$$

where $E^\mu_{\pi_0}$ denotes the expected value with respect to the probability measure $P^\mu_{\pi_0}$. Based on this representation of $p^\mu_{\pi_0}(A)$ as a multiplicative cost, reachability can be addressed by dynamic programming, [2]. Consider a Markov policy $\mu = (\mu_0, \mu_1, \ldots, \mu_{N-1}) \in \mathcal{M}_m$. For each $k \in [0,N]$, $s \in \mathcal{S}$, define $V^\mu_k : \mathcal{S} \to [0,1]$ as

$$V^\mu_k(s) := \mathbf{1}_A(s) \int_{A^{N-k}} \prod_{h=k+1}^{N-1} T^{\mu_h}_s(ds_{h+1}|s_h) T^{\mu_k}_s(ds_{k+1}|s),$$

where $\int_{A^0} \ldots = 1$ and $T^{\mu_l}_s(\cdot|s) = T_s(\cdot|s,\mu_l(s))$, $s \in \mathcal{S}$, $l \in [k, N-1]$. It is easily seen that $V^\mu_k(s)$, $s \in \mathcal{S}$, represents the probability of staying inside the safe set A over the (residual) time horizon $[k,N]$ under policy $\mu \in \mathcal{M}_m$, when the state at time k is $s \in \mathcal{S} : V^\mu_k(s) = E^\mu_{\pi_0}\big[\prod_{h=k}^{N} \mathbf{1}_A(\mathbf{s}(h)) | \mathbf{s}(k) = s\big]$. Note that $V^\mu_k(s)$ does not depend on π_0. For any $\pi_0, p^\mu_{\pi_0}(A)$ can then be expressed as

$$p^\mu_{\pi_0}(A) = \int_{\mathcal{S}} V^\mu_0(s) \pi_0(ds).$$

In [2] it is shown that, for a fixed Markov policy $\mu = (\mu_0, \mu_1, \ldots, \mu_{N-1})$, $\mu_k : \mathcal{S} \to \mathcal{U} \times \Sigma$, $k = 0, 1, \ldots, N-1$, the functions $V^\mu_k : \mathcal{S} \to [0,1]$, $k = 0, 1 \ldots, N$, can be computed by the following backward recursion:

$$V^\mu_k(s) = \mathbf{1}_A(s) \int_{\mathcal{S}} V^\mu_{k+1}(s_{k+1}) T^{\mu_k}_s(ds_{k+1}|s),$$

initialized with $V^\mu_N(s) = \mathbf{1}_A(s)$, $s \in \mathcal{S}$.

In the case when policy $\mu \in \mathcal{M}_m$ is not fixed a-priori and we are dealing with a safety problem, we have the

possibility to design μ so as to maximize the safety level, [2]. A Markov policy $\mu^* \in \mathcal{M}_m$ is *maximally safe* if $p_s^{\mu^*}(A) = \sup_{\mu\in\mathcal{M}_m} p_s^{\mu}(A)$, $\forall s \in \mathcal{S}$.

The problem of computing a maximally safe policy for a DTSHS $\mathcal{H}$ is a stochastic optimal control problem with multiplicative cost for a controlled Markov process on a hybrid state space. Not surprisingly, Theorem 1 shows how to compute a maximally safe policy based on a dynamic programming backward iterative algorithm, [2].

Theorem 1: Define $V_k^* : \mathcal{S} \to [0,1]$, $k = 0, 1, \ldots, N$, by the backward recursion:

$$V_k^*(s) = \sup_{(u,\sigma)\in\mathcal{U}\times\Sigma} \mathbf{1}_A(s) \int_{\mathcal{S}} V_{k+1}^*(s_{k+1}) T_s(ds_{k+1}|s,(u,\sigma)),$$

$s \in \mathcal{S}$, initialized with $V_N^*(s) = \mathbf{1}_A(s)$, $s \in \mathcal{S}$.
Then, $V_0^*(s) = \sup_{\mu\in\mathcal{M}_m} p_s^{\mu}(A)$, $s \in \mathcal{S}$.
If $\mu_k^* : \mathcal{S} \to \mathcal{U}\times\Sigma$, $k \in [0, N-1]$, is such that, $\forall s \in A$

$$\mu_k^*(s) \in \arg \sup_{(u,\sigma)\in\mathcal{U}\times\Sigma} \mathbf{1}_A(s) \int_{\mathcal{S}} V_{k+1}^*(s_{k+1}) T_s(ds_{k+1}|s,(u,\sigma)), \tag{3}$$

then, $\mu^* = (\mu_0^*, \ldots, \mu_{N-1}^*)$ is a maximally safe Markov policy. □

Note that $V_k^*(s)$ represents the maximal safety level over the time interval $[k, N]$ starting from $s \in \mathcal{S}$: $V_k^*(s) = \sup_{\mu\in\mathcal{M}_m} V_k^{\mu}(s)$. In the dynamic programming literature V_k^{μ} and V_k^* are called *value function* and *optimal value function*, respectively, and the backward recursion that yields V_0^* is known as *value iteration*.

IV. Approximate Value Iteration for Reachability Computations

The generality of the stochastic hybrid model introduced in Definition 1 and the structure of the value iteration in Theorem 1 suggest that the solution to the reachability problem will rarely admit an explicit form. Hence, an implementable version of the procedure in Theorem 1 needs to be proposed. In particular, the computational aspects associated with the problem are of key importance for its "practical" solution.

The classical method for the numerical solution to dynamic programming rests on the discretization of the state and control spaces: the (approximate) optimal value functions are represented by piecewise constant functions on a partition of the state space, and optimization is performed over the discretized input set. In [1], this approximation scheme is applied to the value iteration in Theorem 1. Since the optimal value functions $V_k^* : \mathcal{S} \to [0,1], k = 0, 1, \ldots, N$, are identically zero outside the safe set $A = \cup_{q\in\mathcal{Q}}\{q\} \times A_q$, then computations are in fact confined to A, hence gridding can be restricted to the sets $A_q \in \mathcal{B}(\mathbb{R}^{n(q)}), q \in \mathcal{Q}$. In [1], it is shown that if A_q, $q \in \mathcal{Q}$, are compact sets, then, under weak regularity assumptions on the stochastic kernel T_s (Lipschitz continuity), the so-obtained numerical solution converges to the actual solution with known rate, as the grid size goes to zero.

Unfortunately, as is often the case for grid-based methods, the scalability issue appears to be critical for the applicability to practical problems of this numerical approximation scheme. In each iteration one has to manipulate and store functions that are represented by a number of values that grows exponentially with the dimension of the continuous state space. This *curse of dimensionality* makes the solution of reachability problems for high-dimensional DTSHS prohibitive and calls for some approximation scheme to reduce the computational burden.

The approximate value iteration method tries to defeat the curse of dimensionality by approximating the optimal value functions by finitely parameterized functions of known structure. Value iteration is then applied to these compactly-represented approximations, rather than to look-up tables, as in the grid-based approximation of the original DP. Depending on the size of the parameters set, this may result in an effective speedup of the overall computations. Typically, a linear combination of pre-specified basis functions is adopted. In our hybrid setting the approximate optimal value function at time $k \in \{0, 1, \ldots, N-1\}$ takes the form:

$$\hat{V}_k^*(s; w_k) = \sum_{i=1}^{h} w_{i,k}^q \phi_i(x),\ s = (q, x) \in \mathcal{S},$$

where the parameter vector consists of $w_k = (w_k^{q_1}, w_k^{q_2}, \ldots, w_k^{q_m}), w_k^q = (w_{1,k}^q, w_{2,k}^q, \ldots, w_{h,k}^q)$.
Determining the approximate optimal value function at time k amounts to determining its parameter vector w_k^*. Here, unlike the more standard non-hybrid setting, for any k we have m approximate functions, one per mode $q \in \mathcal{Q}$, and each one with its own parameter vector w_k^q.

The approximate optimal value functions $\hat{V}_k^*(\cdot; w_k^*)$, $k = 0, 1, \ldots, N-1$, are computed according to a backward iterative procedure initialized as in Theorem 1, where each iteration consists of two steps, as described hereafter, at time $k \in \{0, 1, \ldots, N-1\}$. Suppose that $\hat{V}_{k+1}^*(\cdot; w_{k+1}^*)$ is known. Then, $\hat{V}_k^*(s; w_k^*)$ is obtained as follows:

Step 1. apply the value iteration operator to $\hat{V}_{k+1}^*(\cdot; w_{k+1}^*)$:

$$\bar{V}_k(s) = \sup_{(u,\sigma)\in\mathcal{U}\times\Sigma} \mathbf{1}_A(s) \int_A \hat{V}_{k+1}^*(s_{k+1}; w_{k+1}^*) T_s(ds_{k+1}|s,(u,\sigma)), \tag{4}$$

Step 2. minimize the weighted L^2-norm of the error:

$$w_k^* = \arg \min_{w_k\in\mathbb{R}^{mh}} \int_A \pi(s)\big(\bar{V}_k(s) - \hat{V}_k^*(s; w_k)\big)^2 ds, \tag{5}$$

where $\pi(s) \geq 0$, $s \in A$, is a weighting function that allows to obtain a more accurate fitting for those states that are known to be critical or frequently visited by the system executions. In our reachability application π takes larger values close to the boundary of the safe set A. The resulting two-step iteration can be viewed as the application of a modified version of the value iteration operator that includes a projection with respect to a weighted L^2-norm, [8].

Note that, in the implementation of the approximate value iteration algorithm, one needs to compute the integral in equation (4). Despite the fact that $V_{k+1}^*(\cdot; w_{k+1}^*)$ is known in analytic form, this integral, in general, has to be solved numerically. In contrast with the numerical approximation to DP based on gridding, though, the values of $V_{k+1}^*(\cdot; w_{k+1}^*)$ that are needed for solving the integral can be determined

on-the-fly, based on its analytic expression. As for the integral in (5), an accurate approximation can be obtained by considering π as a probability density with support on A and replacing (5) with:

$$w_k^* = \arg \min_{w_k \in \mathbb{R}^{mh}} \sum_{s \in \bar{\mathcal{S}}} \left(\bar{V}_k(s) - \hat{V}_k^*(s; w_k) \right)^2, \qquad (6)$$

where $\bar{\mathcal{S}}$ are a set of samples extracted from A according to π. The problem of solving (6) can be viewed as that of training the neural network $\hat{V}_k^*(\cdot; w_k)$ with weights w_k based on the training data set $\{(s, \bar{V}_k(s),\ s \in \bar{\mathcal{S}}\}$. This allows to use well-studied algorithms developed in the neural networks field for the implementation of the approximate value iteration [4].

Once the approximate optimal value functions $\hat{V}_k^*(\cdot; w_k^*)$, $k = 0, 1, \ldots, N-1$, are known, it is then possible to compute the approximate maximally safe policy $\hat{\mu}^* = (\hat{\mu}_0^*, \ldots, \hat{\mu}_{N-1}^*)$ as in Theorem 1. The availability of an analytic–though approximate–expression of the optimal value functions that is also easy to store makes it more convenient to compute on-line only the control input to be applied at the states actually visited by the system during its execution.

V. Case Study

In this section we refer to a benchmark case study for hybrid systems described in [10]. The objective is the simultaneous temperature regulation in r rooms, where $r \geq 1$, by means of a single heater that can switch between different rooms. The task consists of designing a (switching) control strategy that establishes which room should be heated at what time based on the measurements of the r rooms temperatures, so as to maintain the temperature of each room within a prescribed range over a finite time horizon.

We compare the results obtained by the approximate value iteration algorithm (denoted as AVI) with those obtained through the numerical solution to the dynamic programming equations based on state space gridding (denoted as DP), which have been shown in [1] to be potentially as close to the actual solutions as desired.

A. Modeling

The system is modeled by a DTSHS, whose discrete state component $\mathbf{q}$ represents which of the r rooms is being heated, and whose continuous state component $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_r)$ represents the uniform temperature in the r rooms. The discrete state space can then be defined as $\mathcal{Q} = \{\texttt{ON}_1, \texttt{ON}_2, \ldots, \texttt{ON}_r, \texttt{OFF}\}$, where in mode $\texttt{ON}_i$ it is room i to be heated and in mode $\texttt{OFF}$ no room is heated. The map $n : \mathcal{Q} \to \mathbb{N}$ is the constant map $n(q) = r, \forall q \in \mathcal{Q}$.

The reset control space is trivial, $\Sigma = \{0\}$. The transition control space is $\mathcal{U} = \{\texttt{SW}_1, \texttt{SW}_2, \ldots, \texttt{SW}_r, \texttt{SW}_{\texttt{OFF}}\}$, where $\texttt{SW}_i$ and $\texttt{SW}_{\texttt{OFF}}$ correspond to the command of heating room i and heating no room, respectively.

The evolution of the temperature $\mathbf{x}_i$ in room i is governed by the following linear stochastic difference equation:

$$\begin{aligned} \mathbf{x}_i(k+1) = \mathbf{x}_i(k) &+ \sum_{j \neq i} a_{ij} \big(\mathbf{x}_j(k) - \mathbf{x}_i(k) \big) \qquad (7) \\ &+ b_i \big(x_a - \mathbf{x}_i(k) \big) + c_i h_i(k) + \mathbf{n}_i(k), \end{aligned}$$

which is obtained by discretizing, via the constant-step Euler-Maruyama scheme with discretization step Δ, a set of continuous time equations, as described in [2]. The term x_a is the ambient temperature, which is assumed to be fixed. The constants b_i, a_{ij}, and c_i are non-negative and represent the average heat loss rates of room i to the ambient (b_i) and to room $j \neq i$ (a_{ij}), and the heat rate supplied by the heater in room i (c_i), all normalized with respect to the average thermal capacity of room i and rescaled by Δ (according to the integration scheme). The term $h_i(k)$ is a Boolean function equal to 1 if $\mathbf{q}(k) = \texttt{ON}_i$ (i.e. if it is room i to be heated at time k), and equal to 0 otherwise. Furthermore, the disturbance $\{\mathbf{n}_i(k), k = 0, \ldots, N-1\}$ affecting the temperature evolution is a sequence of i.i.d Gaussian random variables with zero mean and variance ν^2 proportional to Δ.

Let $\mathcal{N}(\cdot; m, V)$ denote the probability measure over $(\mathbb{R}^r, \mathcal{B}(\mathbb{R}^r))$ associated with a Gaussian density function with mean m and covariance matrix V. Then, the continuous, control-independent transition kernel T (implicitly defined by (7)) can be expressed as follows:

$$T(\cdot\,|(q,x),u) = T(\cdot\,|(q,x)) = \mathcal{N}(\cdot; x + \Xi x + \Gamma(q), \nu^2 I),$$

where Ξ is a square matrix of size r, $\Gamma(q)$ is an r-dimensional column vector that depends on $q \in \mathcal{Q}$, and I is the identity matrix of size r. For any $i = 1, \ldots, r$, the element in row i and column j of matrix Ξ is given by $[\Xi]_{ij} = a_{ij}$, if $j \neq i$, and $[\Xi]_{ij} = -b_i - \sum_{k \neq i, k \in \mathcal{Q}} a_{ik}$, if $j = i$; as for the vector $\Gamma(q)$, its i^{th} component is $[\Gamma(q)]_i = b_i x_a + c_i$, if $q = \texttt{ON}_i$, and $[\Gamma(q)]_i = b_i x_a$, if $q \in \mathcal{Q} \setminus \{\texttt{ON}_i\}$.

We assume that whenever a discrete transition occurs, say from mode q to mode q', the temperature resets according to the dynamics of mode q. This is modeled by defining the reset kernel $R(\cdot\,|(q,x),\sigma,q') = T(\cdot\,|(q,x))$, $q, q' \in \mathcal{Q}$, $\sigma \in \Sigma$, $x \in \mathbb{R}^r$.

The transition control input affects the discrete state evolution through the discrete transition kernel T_q. In this case study, discrete transitions are not influenced by the continuous state component, so that the discrete state evolves according to a (finite state and finite input) controlled Markov chain with controlled transition probabilities $T_q : \mathcal{Q} \times \mathcal{Q} \times \mathcal{U} \to [0,1]$, where $T_q(q'|q,u)$ represents the probability that mode q' is the successor of mode q when the transition control input u is applied. For ease of notation we set $T_q(q'|q,u) = \alpha_{qq'}(u)$, $q, q' \in \mathcal{Q}$.

B. Control

The objective of the case study is to maintain the temperature of the r rooms within a certain range over a finite time horizon by heating one room at a time and switching the heating action between the different rooms. To this purpose, we devise a Markov policy that decides at each time instant which room should be heated based on the current value of the temperature in the r rooms. This control design problem

can be reformulated as a safety problem for the model introduced above. The 'safe' set A is represented by the desired temperature range for each room.

The value iteration in Theorem 1 can be used to compute a maximally safe policy $\mu^* = (\mu_0^*, \mu_1^*, \ldots, \mu_{N-1}^*)$, $\mu_k^* : \mathcal{S} \to \mathcal{U}$, $k = 0, 1, \ldots, N-1$, and the maximal safety level function $V_0^*(s)$, $s \in A$, representing the maximal probability $p_s^{\mu^*}(A)$ of remaining within the safe set A over the time horizon $[0, N]$, starting from the initial condition $s \in A$.

C. Numerical results

We present the results for the case of $r = 1$ and $r = 2$ rooms. The temperature is measured in degrees Celsius and one discrete time unit corresponds to $\Delta = 2$ minutes. The discrete time horizon is $[0, N]$ with $N = 300$, which corresponds to 10 hours.

In the single room case, the discrete state space is $\mathcal{Q} = \{\texttt{ON}, \texttt{OFF}\}$ and the continuous state space is $\mathbb{R}$. For $r = 2$, $\mathcal{Q} = \{\texttt{ON}_1, \texttt{ON}_2, \texttt{OFF}\}$ and the continuous space is $\mathbb{R}^2$. The desired temperature interval is $[17.5, 22]$ in both cases. Thus, the safe set A is given by $A = \mathcal{Q} \times A_x$ with $A_x := [17.5, 22]$ if $r = 1$, while $A_x := [17.5, 22] \times [17.5, 22]$ if $r = 2$.

The parameters in equation (7) for the case $r = 2$ are set equal to: $x_a = 6$, $b_1 = b_2 = 0.1/30$, $a_{12} = a_{21} = 0.25/30$, $c_1 = 12/30$, $c_2 = 14/30$, and $\nu^2 = 1/30$. In the single room case ($r = 1$), only x_a, b_1, c_1, and ν^2 should be considered.

The transition control input takes values in $\mathcal{U} = \{\texttt{SW}_1, \texttt{SW}_2, \texttt{SW}_{\texttt{OFF}}\}$ for $r = 2$, and $\mathcal{U} = \{\texttt{SW}_{\texttt{ON}}, \texttt{SW}_{\texttt{OFF}}\}$ for $r = 1$. We suppose that when a command to transition from one mode to another is issued, then the prescribed switch actually occurs with a probability 0.8, whereas the complement probability is evenly shared between the case where the situation remains unchanged (which models a *delay*) or (in the $r = 2$ configuration) the case where a transition to the third, non-recommended node, occurs (which models a *faulty behavior*). Instead, when a command of remaining in the current mode of operation is issued, this happens with probability 1. These specifications can be formalized by appropriately defining the transition probabilities $\{\alpha_{qq'}(u), q, q' \in \mathcal{Q}\}$, for any $u \in \mathcal{U}$. In the $r = 1$ case, for $u = \texttt{SW}_{\texttt{ON}}$, $\alpha_{\texttt{ON ON}}(\texttt{SW}_{\texttt{ON}}) = 1$, $\alpha_{\texttt{ON OFF}}(\texttt{SW}_{\texttt{ON}}) = 0$, $\alpha_{\texttt{OFF ON}}(\texttt{SW}_{\texttt{ON}}) = 0.8$, and $\alpha_{\texttt{OFF OFF}}(\texttt{SW}_{\texttt{ON}}) = 0.2$. Instead, in the $r = 2$ case, for $u = \texttt{SW}_1$, $\alpha_{\texttt{ON}_1\texttt{ON}_1}(\texttt{SW}_1) = 1$, $\alpha_{\texttt{ON}_2\texttt{ON}_1}(\texttt{SW}_1) = 0.8$, $\alpha_{\texttt{ON}_2\texttt{ON}_2}(\texttt{SW}_1) = 0.1$, $\alpha_{\texttt{OFF ON}_1}(\texttt{SW}_1) = 0.8$, and $\alpha_{\texttt{OFF OFF}}(\texttt{SW}_1) = 0.1$, the other probabilities $\alpha_{qq'}(\texttt{SW}_1)$ being determined by the normalization condition $\sum_{q' \in \mathcal{Q}} \alpha_{qq'}(\texttt{SW}_1) = 1$, $q \in \mathcal{Q}$.

For the DP approximation, we have adopted a uniform gridding of the continuous domains. More precisely, in the $r = 1$ case the safe interval $A_x = [17.5, 22]$ of temperatures is partitioned in 100 subintervals. In the $r = 2$ case, we have considered three levels of uniform discretization, made up of respectively 18, 36, and 72 bins. Hence, by "level of discretization k" we mean that each side of the continuous set $A_x = [17.5, 22] \times [17.5, 22]$ is partitioned into k subintervals–thus inducing a partition of A_x into k^2 cells.

With regards to the AVI approximation described in section IV, in both studies we have employed a set $\bar{\mathcal{S}}$ of 300 representative points per mode for the training of the approximating neural network. As we expect the safety level to be relatively flat in the central region of the temperature range, while dropping at its boundaries, these points have been sampled according to a probability distribution that favors extractions close to the boundary of the safe set. The integral in equation (4) has been solved numerically using uniform gridding as in the DP approximation. Generalized regression neural networks, that is linearly-parameterized radial basis networks, have been chosen for approximating the optimal value functions. The incremental gradient descent method with an adaptive step has been adopted to solve the least squares minimization problem for training the neural network on the estimated training data, [4].

Figure 1 compares the maximally safe policy estimates obtained from the AVI (left plot) and DP (right plot) approximation schemes. Function $\hat{\mu}_k^*$ at time step $k = 50$ is plotted, coding $\texttt{SW}_{\texttt{ON}}$ with 1 and $\texttt{SW}_{\texttt{OFF}}$ with 0. Figure 2 shows some executions obtained by selecting the initial condition through uniform sampling over the safe set, and by applying the maximally safe policy derived with, respectively, the AVI (left plot) and the DP (right plot) approximation schemes.

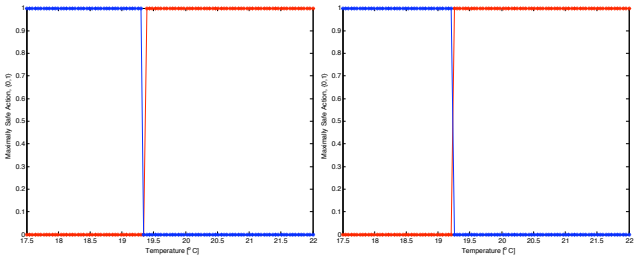

Fig. 1. Single room case: maximally safe policy at time $k = 50$ determined through the AVI (left) and the DP (right) approximation scheme. The blue line represents $\hat{\mu}_k^*(\texttt{OFF}, \cdot)$, and the red line represents $1 - \hat{\mu}_k^*(\texttt{ON}, \cdot)$.

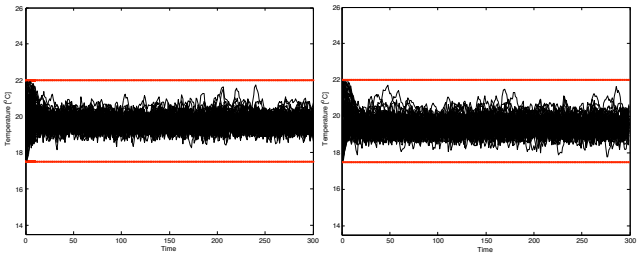

Fig. 2. Single room case: executions under the maximally safe policy $\hat{\mu}^*$ determined through the AVI (left) and the DP (right) approximation scheme.

As for the case $r = 2$, Figure 3 plots the maximal safety level $\hat{V}_0^*(q, x)$ obtained via the AVI and DP approximations as a function of x for the initial mode $q = \texttt{OFF}$, when the discretization level is 18 and 36. Figure 4 represents the corresponding estimates of the maximally safe policy at time $k = 0$ for the discretization level 18. The computational effort involved in the AVI approximation is compared in Table I with that of the DP approximation based on uniform gridding, for different discretization levels. The average time required for reachability computations is reported together with the standard deviation. Computations were performed

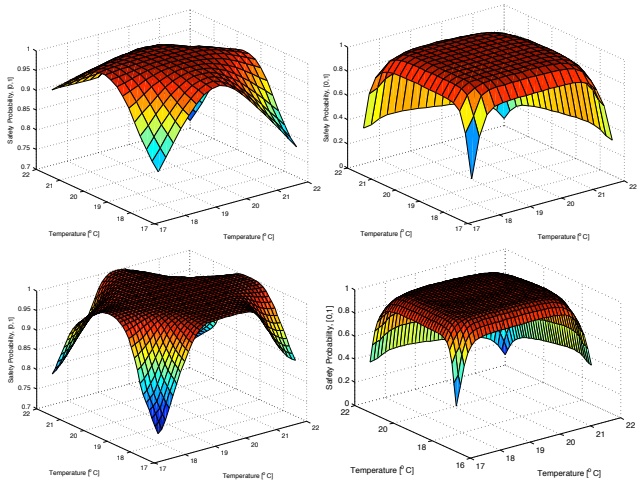

Fig. 3. Two rooms case: maximal safety level function determined through the AVI (left) and the DP (right) approximation scheme, initial mode OFF, discretization level 18 (top row) and 36 (bottom row).

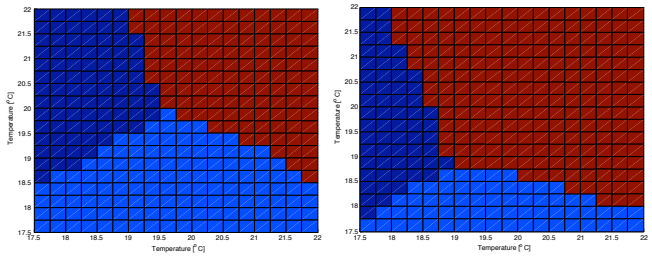

Fig. 4. Two rooms case: maximally safe policy at time $k = 0$ determined by the AVI (left) and DP (right) approximation scheme, initial mode OFF, discretization level 18. Red = $\mathtt{SW}_{\mathtt{OFF}}$, dark blue = $\mathtt{SW}_1$, light blue = $\mathtt{SW}_2$.

TABLE I

COMPUTATIONAL PERFORMANCE OF AVI AND DP APPROXIMATIONS

	AVI [sec]	DP [sec]
one room partition in 100 intervals	$\mu = 11.47$ $\sigma = 0.04$	$\mu = 6.13$ $\sigma = 0.37$
two rooms discretization level 18	$\mu = 21.91$ $\sigma = 0.46$	$\mu = 212.05$ $\sigma = 3.03$
two rooms discretization level 36	$\mu = 26.67$ $\sigma = 0.54$	$\mu = 892.2$ $\sigma = 8.01$
two rooms discretization level 72	$\mu = 45.07$ $\sigma = 2.67$	$\mu = 6265.3$ $\sigma = 16.1$

using MATLAB on an Intel Xeon CPU with 2GHz, 4 GB. Table I shows that AVI eventually outperforms the DP numerical approximation. In particular, it is by considering how the simulation time scales with the discretization level that the difference between the two approaches emerges. While the performances are comparable in the case when the discretization level is low, as soon as a higher accuracy is required, the DP approximation method considerably slows down.

VI. CONCLUSIONS

Prompted by the well-known curse of dimensionality issue arising in the numerical solution to dynamic programming and inspired by [9], [12], this paper proposes to adopt a neural approximation to perform probabilistic reachability computations. An approximate value iteration algorithm tailored to the probabilistic safety problem for a controlled stochastic hybrid model is proposed. The outcome of a simulation study on a benchmark example suggests that the approximate value iteration can outperform the standard numerical approximation to dynamic programming based on gridding when the dimension of the continuous state space is high. In turn, the approximate value iteration approach requires a certain amount of setup, in terms of the choice of the kernels for the approximating functions, of the selection of the number of samples, and of the tuning of the training algorithm. Moreover, unlike in the numerical dynamic programming approximation [1], it is quite difficult to analyze the quality of the approximate value iteration solution, which depends on the adopted class of approximating functions and on the error propagation through iterations. In this respect, this work should be seen as a first step towards the adoption of neural approximation for probabilistic reachability computations of stochastic hybrid systems.

REFERENCES

[1] A. Abate, S. Amin, M. Prandini, J. Lygeros, and S. Sastry. Computational approaches to reachability analysis of stochastic hybrid systems. In A. Bemporad, A. Bicchi, and G. Buttazzo, editors, *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science 4416, pages 4–17. Springer Verlag, 2007.

[2] A. Abate, M. Prandini, J. Lygeros, and S. Sastry. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, Nov 2008. In press.

[3] D. P. Bertsekas and S. E. Shreve. *Stochastic optimal control: the discrete-time case*. Athena Scientific, 1996.

[4] D. P. Bertsekas and J. N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[5] H.A.P. Blom and J. Lygeros, editors. *Stochastic Hybrid Systems: Theory and Safety Critical Applications*. LNCIS, vol. 337, Springer Verlag, 2006.

[6] M. L. Bujorianu. Extended stochastic hybrid systems and their reachability problem. In R. Alur and G. Pappas, editors, *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science 2993, pages 234–249. Springer Verlag, 2004.

[7] C.G. Cassandras and J. Lygeros, editors. *Stochastic hybrid systems*. Automation and Control Engineering Series 24. Taylor & Francis Group/CRC Press, 2006.

[8] D.P. de Farias and B. Van Roy. On the existence of fixed points for approximate value iteration and temporal-difference learning. *Journal of Optimization Theory and Applications*, 105, 2000.

[9] B. Djeridane and J. Lygeros. Neural approximation of PDE solutions: An application to reachability computations. In *IEEE Conference on Decision and Control*, pages 3034–3039, San Diego, USA, December 2006.

[10] A. Fehnker and F. Ivančić. Benchmarks for hybrid systems verifications. In R. Alur and G.J. Pappas, editors, *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science 2993, pages 326–341. Springer Verlag, 2004.

[11] X. Koutsoukos and D. Riley. Computational methods for reachability analysis of stochastic hybrid systems. In J. Hespanha and A. Tiwari, editors, *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science 3927, pages 377–391. Springer Verlag, 2006.

[12] K. N. Niarchos and J. Lygeros. A Neural Approximation to Continuous Time Reachability Computations. In *IEEE Conference on Decision and Control*, San Diego, USA, December 2006.

[13] S. Prajna, A. Jadbabaie, and G.J. Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8):1415–1428, Aug 2007.

[14] M. Prandini and J. Hu. Stochastic reachability: Theory and numerical approximation. In C.G. Cassandras and J. Lygeros, editors, *Stochastic hybrid systems*, Automation and Control Engineering Series 24, pages 107–138. Taylor & Francis Group/CRC Press, 2006.