Journal of Artificial Intelligence Research 60 (2017) 263-285 Submitted 03/17; published 10/17

The Linear Programming Approach to Reach-Avoid Problems for Markov Decision Processes

Nikolaos Kariotoglou KARIOTO@CONTROL.EE.ETHZ.CH
Automatic Control Laboratory
Information Technology and Electrical Engineering
ETH Zürich, CH-8092, Switzerland

Maryam Kamgarpour MKAMGAR@CONTROL.EE.ETHZ.CH
Automatic Control Laboratory
Information Technology and Electrical Engineering
ETH Zürich, CH-8092, Switzerland

Tyler H. Summers TYLER.SUMMERS@UTDALLAS.EDU
Control, Optimization, and Networks Lab
Mechanical Engineering
The University of Texas at Dallas, Texas, TX 75080, USA

John Lygeros LYGEROS@CONTROL.EE.ETHZ.CH
Automatic Control Laboratory
Information Technology and Electrical Engineering
ETH Zürich, CH-8092, Switzerland

Abstract

One of the most fundamental problems in Markov decision processes is analysis and control synthesis for safety and reachability specifications. We consider the stochastic reach-avoid problem, in which the objective is to synthesize a control policy to maximize the probability of reaching a target set at a given time, while staying in a safe set at all prior times. We characterize the solution to this problem through an infinite dimensional linear program. We then develop a tractable approximation to the infinite dimensional linear program through finite dimensional approximations of the decision space and constraints. For a large class of Markov decision processes modeled by Gaussian mixtures kernels we show that through a proper selection of the finite dimensional space, one can further reduce the computational complexity of the resulting linear program. We validate the proposed method and analyze its potential with numerical case studies.

1. Introduction

A wide range of controlled dynamical systems can be modeled using the framework of Markov decision processes (MDPs) (Feinberg, Shwartz, & Altman, 2002; Puterman, 1994). Depending on the problem at hand, several objectives can be formulated for an MDP including maximization of a reward function or satisfaction of a specification defined by a formal language. Safety and reachability are two of the most fundamental specifications for a dynamical system. In a reach-avoid problem for an MDP, the objective is to maximize the probability of reaching a target set within a given time horizon while staying in a safe set (Abate, Prandini, Lygeros, & Sastry, 2008). This objective is stage-wise *sum-multiplicative*, in contrast to the stage-wise additive cost functions typically used in MDPs. This addresses a recognized limitation of additive cost functions: many tasks are not easily encoded by an additive cost function and are more naturally posed in terms of reaching and avoiding certain sets (Kolobov, Mausam, Weld, & Geffner, 2011; Steinmetz, Hoffmann, & Buffet, 2016). The stochastic reach-avoid framework is utilized in applications including aircraft conflict detection (Watkins & Lygeros, 2003; Ding, Kamgarpour, Summers, Abate, Lygeros, & Tomlin, 2013), feedback control of camera networks (Kariotoglou, Raimondo, Summers, & Lygeros, 2015) and optimal feedback policies for building evacuation under a randomly evolving hazards (Wood, Summers, & Lygeros, 2013).

The reach-avoid problem considered in this paper is closely related to the stochastic shortest path problem (Bertsekas & Tsitsiklis, 1991). In contrast to stochastic shortest path however, there is no cost function for transitioning from one state (often treated as a graph node) to another. As pointed out by Kolobov et al. (2011), Kolobov, Mausam, and Weld (2012), Steinmetz et al. (2016), this difference makes the dynamic programming algorithm developed for stochastic shortest path fail for the reachability and consequently the reach-avoid problem. Hence, Kolobov et al. (2011, 2012) propose the so-called generalized stochastic shortest path framework that can address a wide range of stage cost structures. Similarly, Steinmetz et al. (2016) highlight the under-explored stochastic reachability problem and proposes a heuristic approach for the stochastic reachability problem. Though our problem has very similar objective, it is not formulated in the category of MDPs considered by Kolobov et al. (2011, 2012), Steinmetz et al. (2016) due to the continuous state and input spaces. As such, our approximation approach is different than the past proposed heuristics in that we utilize optimization in continuous spaces. Note that continuous spaces are a natural modeling framework in several dynamical systems, such as robots, where the dynamics are described by physical laws of motion.

The dynamic programming (DP) principle characterizes the solution to the stochastic reach-avoid problem with continuous state and action spaces (Prandini & Hu, 2006). One can approximate the DP equations on a finite grid defined over the MDP state and action spaces. Gridding techniques are theoretically attractive since they can provide explicit error bounds for the approximation of the value function under general Lipschitz continuity assumptions (Abate, Amin, Prandini, Lygeros, & Sastry, 2007; Kushner & Dupuis, 2001). In practice, the complexity of gridding based techniques suffers from the infamous Curse of Dimensionality. That is, the sum of state and control space dimensions that can be addressed is limited by the cardinality of the state-control pairs that need to be considered to fairly approximate reach-avoid probabilities. Typically, the required cardinality to keep approximations meaningful scales exponentially with dimensions of state and action spaces. An important problem is therefore to explore approximation techniques that scale better.

Several researchers have developed approximate dynamic programming (ADP) techniques for various classes of stochastic control problems (Powell, 2007; Bertsekas, 1995). Most of the existing work has focused on problems where the state and control spaces are finite but too large to directly solve DP recursions. Our work is motivated by the technique discussed by de Farias and Van Roy (2003) where the authors develop an ADP method for optimal control of an MDP with finite state and action spaces and an infinite horizon discounted additive stage cost. In this approach, the value function of the stochastic control problem is approximated as a weighted sum of basis functions, where the weights are the solution to a linear program (LP) (de Farias & Van Roy, 2004). The number of constraints in the LP is equal to the cardinality of state and action spaces. Hence, computation becomes challenging for large MDPs. To handle this, a constraint sampling approach with probabilistic bounds has been proposed by de Farias and Van Roy (2004).

For optimal control of MDPs with continuous state and action spaces and an additive stage cost, an infinite dimensional linear program has been developed to characterize the value function (Hernández-Lerma & Lasserre, 1996). Here, the decision variable is the value function defined over the uncountable state space, hence, it is infinite dimensional. Furthermore, the number of constraints is uncountably infinite since there is one constraint corresponding to each state-input pair. Hauskrecht and Kveton (2003), Kveton, Hauskrecht, and Guestrin (2006) consider a similar setup extending to mixed continuous and discrete state variables. They propose approximating the value function as a weighted sum of basis functions and devise an efficient approach to solving the resulting large-scale LP by considering dynamical systems that are modeled by or can be fairly approximated using the so-called "factored" MDPs. In contrast, in the reach-avoid problem considered here the value function is sum-multiplicative in states over the time horizon and has no discount factor. Furthermore, we make no *a-priori* assumptions on system dynamics.

The LP approach to the stochastic reachability problem for MDPs over continuous state and action spaces and an infinite horizon was first proposed by Kamgarpour, Summers, and Lygeros (2013). An infinite dimensional linear program was formulated whose solution, in theory, would characterize the maximum reachability probability over the continuous state space. However, no computational approach to solving this problem was provided. In general, LP approaches to ADP are desirable since several commercially available software packages can handle LP problems with large numbers of decision variables and constraints. Motivated by this observation and leveraging advances in the past works by Powell (2007), Bertsekas (1995), Kamgarpour et al. (2013) we develop a computational framework to approximate the optimal value function and policy of a stochastic reach-avoid problem over continuous state and action spaces.

Our contributions are as follows. First, we derive an infinite dimensional LP formulated over the space of Borel measurable functions and prove its equivalence to the standard DP-based solution approach for the stochastic reach-avoid problem, under assumptions of the continuity of the MDP transition kernel and compactness of the action space. Second, we prove that through restricting the infinite dimensional decision space to a finite dimensional subspace spanned by a collection of basis functions (semi-infinite or robust LP), we obtain an upper bound on the stochastic reach-avoid value function. Third, we use randomized optimization to obtain a tractable finite dimensional LP with probabilistic feasibility guarantees. The final contribution of our paper is the focus on numerical validation of the LP approach to stochastic reach-avoid problems. As such, we propose a class of basis functions for reach-avoid problems for MDPs with Gaussian mixture kernels. Basis functions in this class have been successfully used in similar function approximation schemes due to their analytic properties (Kveton & Hauskrecht, 2006). We then develop several benchmark problems to test the scalability and accuracy of our proposed method.

A preliminary version of our approach appeared as a brief conference paper Kariotoglou, Summers, Summers, Kamgarpour, and Lygeros (2013). Compared to the results discussed by Kariotoglou et al. (2013), we extend and refine all theoretical statements since lemmas, propositions and theorems proven here were missing. Furthermore, we provide novel numerical studies to illustrate the accuracy of the approach and its applicability to relatively large-scale problems. Given that there are no competing approaches for the problem at hand to handle large state-input dimensions, we compare the results to well-studied heuristics, tuned to approximate the solution to simple stochastic reach-avoid problems.

The rest of the paper is organized as follows. In Section 2 we introduce the stochastic reach-avoid problem for MDPs and formulate an infinite dimensional LP that characterizes its solution. In Section 3 we derive an approach to approximate the solution to the infinite LP through restricting the decision space to a finite dimensional subspace using basis functions and reducing the infinite constraints to finite constraints through sampling. Section 4 proposes Gaussian radial basis functions to analytically compute operations arising in the LP for MDPs with Gaussian mixture kernels. In Section 5 we validate the accuracy and scalability of the solution approach with three case studies.

2. Stochastic Reach-Avoid Problem

We consider a discrete-time controlled stochastic process $x_{t+1} \sim Q(dx|x_t, u_t)$, $(x_t, u_t) \in \mathcal{X} \times \mathcal{U}$. Here, $Q : \mathcal{B}(\mathcal{X}) \times \mathcal{X} \times \mathcal{U} \to [0, 1]$ is a transition kernel and $\mathcal{B}(\mathcal{X})$ denotes the Borel σ-algebra of $\mathcal{X}$. Given a state control pair $(x_t, u_t) \in \mathcal{X} \times \mathcal{U}$, $Q(A|x_t, u_t)$ measures the probability of x_{t+1} belonging to the set $A \in \mathcal{B}(\mathcal{X})$. The transition kernel Q is a Borel-measurable stochastic kernel, that is, $Q(A|\cdot)$ is a Borel-measurable function on $\mathcal{X} \times \mathcal{U}$ for each $A \in \mathcal{B}(\mathcal{X})$ and $Q(\cdot|x, u)$ is a probability measure on $\mathcal{X}$ for each (x, u). For the rest of the paper all measurability conditions refer to Borel measurability. We allow the state space $\mathcal{X}$ to be any subset of $\mathbb{R}^n$ and assume that the control space $\mathcal{U} \subseteq \mathbb{R}^m$ is compact.

We consider a safe set $K' \in \mathcal{B}(\mathcal{X})$ and a target set $K \subseteq K'$. We define an admissible T-step control policy to be a sequence of measurable functions $\mu = \{\mu_0, \ldots, \mu_{T-1}\}$ where $\mu_i : \mathcal{X} \to \mathcal{U}$ for each $i \in \{0, \ldots, T-1\}$. The reach-avoid problem over a finite time horizon T is to find an admissible T-step control policy that maximizes the probability of x_t reaching the set K at some time $j \leq T$ while staying in K' for all $0 \leq t \leq j$. For any initial state x_0, we denote the reach-avoid probability associated with a given μ as

$$r^{\mu}_{x_0}(K, K') = \mathbb{P}^{\mu}_{x_0}\{\exists j \in \{0, \ldots, T\} : x_j \in K \wedge \forall i \in \{0, \ldots, j-1\},\ x_i \in K' \setminus K\}.$$

2.1 Dynamic Programming Approach

The reach-avoid probability $r^{\mu}_{x_0}(K, K')$ can be equivalently formulated as an expected value objective function. In contrast to an optimal control problem with additive stage cost, $r^{\mu}_{x_0}(K, K')$ is a history dependent sum-multiplicative cost function (Summers & Lygeros, 2010):

$$r^{\mu}_{x_0}(K, K') = \mathbb{E}^{\mu}_{x_0}\left[\sum_{j=0}^{T}\left(\prod_{i=0}^{j-1} \mathbb{1}_{K'\setminus K}(x_i)\right)\mathbb{1}_K(x_j)\right], \tag{1}$$

where we use the notation of $\prod_{i=k}^{j}(\cdot) = 1$ if $k > j$. Above, $\mathbb{1}_A(x)$ denotes the indicator function of a set $A \in \mathcal{B}(\mathcal{X})$. Our objective is to find $\sup_\mu r^{\mu}_{x_0}(K, K')$ and the optimal policy achieving the supremum. The sets K and K' can be time-varying or stochastic (Summers, Kamgarpour, Tomlin, & Lygeros, 2013) but for simplicity we assume here that they are constant. We denote the difference between the safe and target sets by $\bar{\mathcal{X}} := K' \setminus K$ to simplify the presentation of our results.

Similar to the dynamic programming approach to an optimal control problem with additive stage cost, the solution to the reach-avoid problem is characterized by a recursion (Summers & Lygeros, 2010) as follows. Define the value functions $V_k^* : \mathcal{X} \to [0,1]$ for $k = T-1, \dots, 0$ as

$$\begin{aligned} &V_T^*(x) = \mathbb{1}_K(x), \\ &V_k^*(x) = \sup_{u \in \mathcal{U}} \left\{ \mathbb{1}_K(x) + \mathbb{1}_{\bar{\mathcal{X}}}(x) \int_{\mathcal{X}} V_{k+1}^*(y) Q(dy|x,u) \right\}. \end{aligned} \tag{2}$$

It can be shown that $V_0^*(x_0) = \sup_\mu r_{x_0}^\mu(K, K')$ (Summers & Lygeros, 2010). Past work has focused on approximating V_k^* recursively on a discretized grid of $\bar{\mathcal{X}}$ and $\mathcal{U}$ (Prandini & Hu, 2006; Abate et al., 2007; Summers & Lygeros, 2010). Note that the DP recursion defined by (2) does not fall into the category of additive discounted cost problems. This difference yields certain approximation approaches for MDPs with discounted additive cost function not applicable to the problem at hand.

Next, we will establish the measurability and continuity properties of the reach-avoid value functions to enable the use of a linear program to approximate these functions.

Assumption 1. *For every $x \in \mathcal{X}, A \in \mathcal{B}(\mathcal{X})$ the mapping $u \mapsto Q(A|x,u)$ is continuous.*

Proposition 1. *Under Assumption* (1)*, at every step k, the supremum in* (2) *is attained by a measurable function $\mu_k^* : \mathcal{X} \to \mathcal{U}$ and the resulting $V_k^* : \mathcal{X} \to [0,1]$ is measurable.*

Proof. By induction. First, note that the indicator function $V_T^*(x) = \mathbb{1}_K(x)$ is measurable. Assuming that V_{k+1}^* is measurable we will show that V_k^* is also measurable. Define $F(x,u) = \int_{\mathcal{X}} V_{k+1}^*(y) Q(dy|x,u)$. Due to continuity of the map $u \mapsto Q(A|x,u)$ by Assumption 1, the map $u \mapsto F(x,u)$ is continuous for every x (Nowak, 1985, Fact 3.9). Since $\mathcal{U}$ is compact, there exists a measurable function $\mu_k^*(x)$ that achieves the supremum (Brown & Purves, 1973, Corollary 1). Furthermore, as shown by Bertsekas and Shreve (1978, Proposition 7.29), the mapping $(x,u) \mapsto F(x,u)$ is measurable. It follows that $F(x, \mu_k^*(x))$, and hence V_k^*, is measurable as it is composition of measurable functions. □

Since the optimizing policy is attained, we will use max instead of sup. Proposition 1 allows one to compute an optimal feedback policy at each stage k through

$$\begin{aligned} \mu_k^*(x) &\in \arg\max_{u \in \mathcal{U}} \left\{ \mathbb{1}_K(x) + \mathbb{1}_{\bar{\mathcal{X}}}(x) \int_{\mathcal{X}} V_{k+1}^*(y) Q(dy|x,u) \right\} \\ &= \arg\max_{u \in \mathcal{U}} \left\{ \int_{\mathcal{X}} V_{k+1}^*(y) Q(dy|x,u) \right\}. \end{aligned} \tag{3}$$

For functions $f, g : \mathcal{X} \to \mathbb{R}$, we use $f \le g$ to denote $f(x) \le g(x)$, $\forall x \in \mathcal{X}$. It is easy to verify by induction that $0 \le V_k^* \le 1$, for $k = T, T-1, \dots, 0$. Furthermore, due to the indicator functions in (2), $V_k^*(x)$ are defined on disjoint regions of $\mathcal{X}$ as:

$$V_k^*(x) = \begin{cases} 1, & x \in K \\ \max_{u \in \mathcal{U}} \int_{\mathcal{X}} V_{k+1}^*(y) Q(dy|x,u), & x \in \bar{\mathcal{X}} \\ 0, & x \in \mathcal{X} \setminus K' \end{cases} \tag{4}$$

Hence, it suffices to compute V_k^* and the optimizing policy on $\bar{\mathcal{X}}$. We show that with an additional assumption on kernel Q, V_k^* is continuous on $\bar{\mathcal{X}}$. The continuity is a desired property for approximating V_k^* on $\bar{\mathcal{X}}$ using basis functions.

Assumption 2. *For every $A \in \mathcal{B}(\mathcal{X})$ the mapping $(x,u) \mapsto Q(A|x,u)$ is continuous.*

Proposition 2. *Under Assumption* (2)*, $V_k^*(x)$ is piecewise continuous on $\mathcal{X}$.*

Proof. From continuity of $(x,u) \mapsto Q(A|x,u)$ we conclude that the mapping $(x,u) \mapsto F(x,u)$ is continuous (Nowak, 1985, Fact 3.9). From the Maximum Theorem (Sundaram, 1996), it follows that $F(x,u^*(x))$ and thus each $V_k^*(x)$, is continuous on $\bar{\mathcal{X}}$. The result follows by (4). □

2.2 Linear Programming Approach

Let $\mathcal{F} := \{f : \mathcal{X} \to \mathbb{R},\ f \text{ is measurable}\}$. For $V \in \mathcal{F}$ define two operators $T_u, T : \mathcal{F} \to \mathcal{F}$

$$\mathcal{T}_u[V](x) = \int_{\mathcal{X}} V(y)Q(dy|x,u), \tag{5}$$

$$\mathcal{T}[V](x) = \max_{u\in\mathcal{U}} \mathcal{T}_u[V](x). \tag{6}$$

Let ν be a non-negative measure supported on $\bar{\mathcal{X}}$, referred to as state-relevance measure.

Theorem 1. *Suppose Assumption 1 holds. For $k \in \{0,\dots,T-1\}$, let V_{k+1}^* be the value function at step $k+1$ defined in* (2)*. Consider the infinite dimensional linear program:*

$$\inf_{V(\cdot)\in\mathcal{F}} \quad \int_{\bar{\mathcal{X}}} V(x)\nu(\mathrm{dx}) \tag{Inf-LP}$$

$$\text{subject to} \quad V(x) \geq \mathcal{T}_u[V_{k+1}^*](x), \quad \forall (x,u) \in \bar{\mathcal{X}} \times \mathcal{U}. \tag{7}$$

(a) Any feasible solution of (Inf-LP) *is an upper bound on the optimal reach-avoid value function V_k^* on $\bar{\mathcal{X}}$; (b) V_k^* is a solution to this optimization problem and any other solution to* (Inf-LP) *is equal to V_k^*, ν-almost everywhere on $\bar{\mathcal{X}}$.*

Although the domain of V is restricted to $\bar{\mathcal{X}} = K' \setminus K$, in order to evaluate a constraint in (7) for each (x,u), one has to evaluate V_{k+1}^* also on the set K to be able to compute the integral $\mathcal{T}_u[V_{k+1}^*]$. This value equals one by definition of the DP in (2).

Proof. Let $J^* \in \mathbb{R}$ denote the optimal value of the objective function in (Inf-LP). From the definition of V_k^* and Proposition 1, $V_k^* \in \mathcal{F}$ and is equal to the supremum over $u \in \mathcal{U}$ of the right hand side of the constraint (7). Hence, for any feasible $V \in \mathcal{F}$, we have $V(x) \geq V_k^*(x)$ for all $x \in \bar{\mathcal{X}}$ and part (a) is shown. By non-negativity of ν it follows that for any feasible V, $\int_{\bar{\mathcal{X}}} V(x)\nu(\mathrm{dx}) \geq \int_{\bar{\mathcal{X}}} V_k^*(x)\nu(\mathrm{dx})$, which implies $J^* \geq \int_{\bar{\mathcal{X}}} V_k^*(x)\nu(\mathrm{dx})$. On the other hand, $J^* \leq \int_{\bar{\mathcal{X}}} V_k^*(x)\nu(\mathrm{dx})$ since it is the least cost among the set of feasible functions. Hence, $J^* = \int_{\bar{\mathcal{X}}} V_k^*(x)\nu(\mathrm{dx})$ and V_k^* is an optimal solution. To show that any other solution to (Inf-LP) is equal to V_k^* ν-almost everywhere on $\bar{\mathcal{X}}$, assume there exists a function V^*, optimal for (Inf-LP) that is strictly greater than V_k^* on a set $A_m \in \mathcal{B}(\mathcal{X})$ of non-zero ν-measure. Since V^* and V_k^* are both optimal, we have that $\int_{\bar{\mathcal{X}}} V^*(x)\nu(\mathrm{dx}) =$

$\int_{\bar{\mathcal{X}}} V_k^*(x)\nu(\mathrm{dx}) = J^*$. We can then reduce V^* to the value of V_k^* on A_m, while ensuring feasibility of V^*. This reduces the value of $\int_{\bar{\mathcal{X}}} V^*(x)\nu(\mathrm{dx})$ below J^*, contradicting that V^* is optimal and part (b) is shown. □

Note that the decision variable in Inf-LP lives in $\mathcal{F}$, an infinite dimensional space. The objectives and constraints are linear in the decision variable but there are infinitely many constraints since $\bar{\mathcal{X}}$ and $\mathcal{U}$ are continuous spaces. This class of problems is referred to in literature as an infinite dimensional linear program (Anderson & Nash, 1987; Hernández-Lerma & Lasserre, 1998). As shown in Theorem 1, the sequence of value functions of the stochastic reach-avoid problem derived in (2) are equivalently characterized as solutions of a sequence of infinite dimensional linear programs. Thus, instead of the classical space gridding approaches to solve (2), we focus on approximating V_k^* by approximating the solutions to (Inf-LP).

3. Approximation with a Finite Linear Program

An infinite dimensional LP is in general NP-hard (Anderson & Nash, 1987; Hernández-Lerma & Lasserre, 1998). We approximate the solution to (Inf-LP) by deriving a finite LP through two steps. First, we restrict the decision space to a finite dimensional subspace $\mathcal{F}^M \subset \mathcal{F}$. Second, we replace the infinite constraints in (7) with a sufficiently large finite number of randomly sampled constraints.

3.1 Restriction to a Finite Dimensional Function Class

Let $\mathcal{F}^M$ be a finite dimensional subspace of $\mathcal{F}$ spanned by M basis elements denoted by $\{\phi_i\}_{i=1}^M$. Given $f \in \mathcal{F}$, consider the following semi-infinite linear program defined over functions $\sum_{i=1}^M w_i\phi_i(x) \in \mathcal{F}^M$ with decision variable $w \in \mathbb{R}^M$:

$$\begin{aligned} \text{S-LP}(f) := \min_{w_1,\dots,w_M} \quad & \sum_{i=1}^M w_i \int_{\bar{\mathcal{X}}} \phi_i(x)\nu(\mathrm{dx}) && \text{(Semi-LP)} \\ \text{subject to} \quad & \sum_{i=1}^M w_i\phi_i(x) \geq \mathcal{T}_u[f](x),\ \forall (x,u) \in \bar{\mathcal{X}} \times \mathcal{U}\,. && (8) \end{aligned}$$

The above linear program has finitely many decision variables and infinitely many constraints. It is referred to as a semi-infinite linear program. We assume that problem (Semi-LP) is feasible. Note that for a bounded f, this can always be guaranteed by including $\phi(x) = 1$ in the basis functions.

Consider the following semi-norm on $\mathcal{F}$ induced by the state-relevance measure ν, $\|V\|_{1,\nu} := \int_{\bar{\mathcal{X}}} |V(x)|\nu(\mathrm{dx})$. In the infinite dimensional linear program (Inf-LP) the choice of ν does not affect the optimal solution, as seen in Theorem (1). For finite dimensional approximations, as will be shown in the next Lemma, ν influences approximation accuracy in different regions of $\bar{\mathcal{X}}$.

Let $\hat{V}_f = \sum_{i=1}^M \hat{w}_i\phi_i$ be a solution to (Semi-LP) and $V_f^* \in \mathcal{F}$ be a solution to (Inf-LP), where the right-hand-side constraint V_{k+1}^* is replaced with f.

Lemma 1. *$\hat{V}_f$ achieves the minimum of $\left\|V - V_f^*\right\|_{1,\nu}$, over the set $\{V \in \mathcal{F}^M,\ V \geq V_f^*\}$.*

Proof. It follows from the proof of Theorem (1) that $V_f^* = \sup_u \mathcal{T}_u[f]$, ν-almost everywhere. Now, a function $\hat{V} \in \mathcal{F}^M$ is an upper bound on $V_f^* = \sup_u \mathcal{T}_u[f]$ if and only if it satisfies constraint (8). To show that $\hat{V}_f$ minimizes the $(1,\nu)$-norm distance to V_f^*, notice that for any $V(x) = \sum_{i=1}^M w_i \phi_i(x)$ satisfying (8) we have that

$$\|V - V_f^*\|_{1,\nu} = \int_{\bar{\mathcal{X}}} |V(x) - V_f^*(x)| \nu(\mathrm{dx}) = \int_{\bar{\mathcal{X}}} V(x)\nu(\mathrm{dx}) - \int_{\bar{\mathcal{X}}} V_f^*(x)\nu(\mathrm{dx}),$$

where the second equality is due to the fact that V is an upper bound of V_f^*. Since V_f^* is a fixed constant in the norm optimization of the lemma above, the result follows. □

The interpretation of the above lemma is that in the class of functions that upper bound V_f^*, we find the function with the least distance to V_f^*, where the distance is defined with respect to the $(1,\nu)$-norm. We will use semi-infinite problem (Semi-LP) to recursively approximate V_k^* using a weighted sum of basis functions as follows.

For every $k \in \{0, \dots, T-1\}$, let $\mathcal{F}^{M_k}$ denote the span of a fixed set of M_k basis elements $\{\phi_i^k\}_{i=1}^{M_k}$ where each $\phi_i^k \in \mathcal{F}$. Start with the known value function V_T^* and recursively construct $\hat{V}_k(x) = \sum_{i=1}^{M_k} \hat{w}_i^k \phi_i^k(x)$ where $\hat{w}_i^k$ is the solution to (Semi-LP) obtained by substituting $f = \hat{V}_{k+1}$ in (Semi-LP), that is, solving S-LP$(\hat{V}_{k+1})$.

Proposition 3. *The functions* $\hat{V}_k$
(a) $\hat{V}_k(x) \geq V_k^*(x)$ *for all* $x \in \bar{\mathcal{X}}$ *and* $k = 0, \dots, T-1$.
(b) Each $\hat{V}_k$ *is an optimizer in:*

$$\min_{V(\cdot)\in\mathcal{F}^{M_k}} \quad \left\|V - V_k^*\right\|_{1,\nu} \tag{9}$$

$$\text{subject to} \quad V(x) \geq \mathcal{T}_u[\hat{V}_{k+1}](x), \quad \forall (x,u) \in \bar{\mathcal{X}} \times \mathcal{U}. \tag{10}$$

Proof. We prove part (a) by induction. Note that at step $T-1$ the results above hold as a direct consequence of Lemma (1). Now, suppose at time step k, $\hat{V}_k(x) \geq V_k^*(x)$. From monotonicity of the operator $\mathcal{T}_u$ (Summers & Lygeros, 2010), it follows that $\mathcal{T}_u[\hat{V}_k](x) \geq \mathcal{T}_u[V_k^*](x)$. By constraint (10), it follows that $\hat{V}_{k-1}(x) \geq \mathcal{T}_u[\hat{V}_k](x) \geq \mathcal{T}_u[V_k^*](x) = V_{k-1}^*(x)$, where the last equality is due to the definition of V_k^* in (4). To prove part (b), consider that $\hat{V}_k$ is the solution for (Semi-LP) with $f = \hat{V}_{k+1}$ which implies that $\hat{V}_k(x) \geq \mathcal{T}_u[\hat{V}_{k+1}](x), \forall (x,u) \in \bar{\mathcal{X}} \times \mathcal{U}$ and it thus satisfies (10). Being a solution to (Semi-LP) also implies that $\hat{V}_k$ achieves the minimum of $\|V - V_{k+1}^*\|_{1,\nu}$ over the set $\{V \in \mathcal{F}^{M_k}\}$. The cost function $\|V - V_k^*\|_{1,\nu}$ expands to $\|V - V_k^*\|_{1,\nu} = \int_{\bar{\mathcal{X}}} |V(x) - V_k^*(x)|\nu(\mathrm{dx}) = \int_{\bar{\mathcal{X}}} V(x)\nu(\mathrm{dx}) - \int_{\bar{\mathcal{X}}} V_k^*(x)\nu(\mathrm{dx})$ where the last step follows from part (b) proven above. Hence minimizing $\|V - V_k^*\|_{1,\nu}$ is equivalent to minimizing $\|V - V_{k+1}^*\|_{1,\nu}$ since V_{k+1}^* and $V_k^*(x)$ are fixed. □

The above proposition illustrates that by restricting the decision space of the infinite dimensional linear program, we obtain an upper bound to the reach-avoid value functions V_k^*, at every step k. This is also the least (in the $(1,\nu)$-norm) upper bound in the space spanned by the basis functions subject to constraint (10).

3.2 Restriction to a Finite Number of Constraints

The semi-infinite linear program, (Semi-LP) is in general NP-hard (Bertsimas, Brown, & Caramanis, 2011; Ben-Tal & Nemirovski, 2002; Hettich & Kortanek, 1993). One way to approximate the solution is to solve a relaxation of the problem by imposing the constraints on only a finite set of points from $\bar{\mathcal{X}} \times \mathcal{U}$. One can then use generalization results from sampled convex programs (Calafiore & Campi, 2006; Campi, Garatti, & Prandini, 2009) to quantify, probabilistically, the feasibility of the obtained solution for the original infinite constraints.

Construct a set of samples $S := \{(x^i, u^i)\}_{i=1}^N$ by drawing N independent points according to a probability measure on $\bar{\mathcal{X}} \times \mathcal{U}$ denoted by $\mathbb{P}_{\bar{\mathcal{X}}\times\mathcal{U}}$. For a fixed function $f \in \mathcal{F}$, consider the following finite LP defined over functions $\sum_{i=1}^M w_i\phi_i(x) \in \mathcal{F}^M$:

$$\begin{aligned} \text{F-LP}(f) := \min_{w_1,\dots,w_M} \quad & \sum_{i=1}^M w_i \int_{\bar{\mathcal{X}}} \phi_i(x)\nu(\mathrm{dx}) \\ \text{subject to} \quad & \sum_{i=1}^M w_i\phi_i(x) \geq \mathcal{T}_u[f](x),\ \forall (x,u) \in S \end{aligned} \tag{Fin-LP}$$

Assume for any $S \subset \bar{\mathcal{X}} \times U$, the feasible region of (Fin-LP) is non-empty and the optimizer is unique. Let $\tilde{w}^S$ be the sample dependent optimizer in (Fin-LP).

Lemma 2. *(Campi et al., 2009, Theorem 1) Choose a violation and confidence levels* $\varepsilon, \beta \in (0,1)$. *If*

$$N \geq \frac{2}{\varepsilon}\left(M + \ln\left(\frac{1}{\beta}\right)\right)$$

Then, with confidence $1-\beta$

$$\mathbb{P}_{\bar{\mathcal{X}}\times\mathcal{U}}\big(\sum_{i=1}^M \tilde{w}_i^S\phi_i(x)(x)\big) < \mathcal{T}_u[f](x)) \leq \varepsilon. \tag{11}$$

The above result indicates that the optimizer $\tilde{w}^S$ violates the constraints of the linear program on a subset of $\bar{\mathcal{X}} \times \mathcal{U}$ with maximum measure ε. Note that $\mathbb{P}_{\bar{\mathcal{X}}\times\mathcal{U}}$ is a choice. The interpretation of (11) is that we bound the size of the set in which the constraints are violated with respect to the chosen measure $\mathbb{P}_{\bar{\mathcal{X}}\times\mathcal{U}}$.

In summary, we can recursively construct $\tilde{V}_k = \sum_{i=1}^{M_k} \tilde{w}_i^k\phi_i^k$ by solving (Fin-LP) using $f = V_T^*$ at time T, and $f = \tilde{V}_{k+1}$ for $k = T-1,\dots,0$. Hence, we solve F-LP($\tilde{V}_{k+1}$) recursively. Given potentially time-varying confidence β_k and violation probability ε_k, we then find the number of samples $N_k(\varepsilon_k, \beta_k, M_k)$ to include in (Fin-LP). It follows that with confidence greater than $1-\beta_k$, the probability of violating the upper bound constraint is less than ε_k. Consequently, the approximation functions $\tilde{V}_k$ are only probabilistic upper bounds on the value functions V_k^*, in contrast to the guaranteed upper bounds provided in Proposition (3). In particular, it can be shown that with confidence $1-\sum_{i=0}^{T-1}\beta_i$, the probability that $\tilde{V}_0$ is an upper bound on V_0^* is greater than $1-\sum_{i=0}^{T-1}\varepsilon_i$ (Kariotoglou, Margellos, & Lygeros, 2016). This recursive approximation technique can be extended to large horizons with a linear increase in the number of decision variables. However, the approximation guarantees get progressively worse.

Remark: An alternative approach to approximate the solution to a semi-infinite linear program could be the constraint generation technique (Patrascu, Poupart, Schuurmans, Boutilier, & Guestrin, 2002; Guestrin, Koller, Parr, & Venkataraman, 2003). Similar to our approach, in constraint generation, the problem is first solved by considering a finite subset S of the constraints. Then, given the optimizer w^S, the most violating constraint in the set $\bar{\mathcal{X}} \times \mathcal{U}$ is computed and added to the set of constraints and the process is repeated until a stopping criteria is reached. Notice that the problem of finding the most violating constraints in our case is a non-convex optimization problem. To the best of our knowledge, convergence of the constraint generation approach is not guaranteed in this case.

Remark: To evaluate the accuracy of $\tilde{V}_k$, ideally, one may attempt to bound $\|\tilde{V}_k - V_k^*\|_{1,\nu}$ as a function of $\|\hat{V}_k - V_k^*\|_{1,\nu}$, where $\hat{V}_k$ is computed according to Proposition (3) and bound $\|\hat{V}_k - V_k^*\|_{1,\nu}$ to a given accuracy using the properties of the basis functions and V_k^*. The first bound is concerned with quantifying the accuracy in the objective function of a sampled convex program. To do so, the number of samples can be updated by considering the Lipschitz properties of the objective function (Mohajerin Esfahani, Sutter, & Lygeros, 2015). The problem of how to determine the basis functions to bound $\|\hat{V}_k - V_k^*\|_{1,\nu}$ to a given accuracy is also heavily dependent on the continuity properties of the objective function and in general is a challenging open problem. For a technical discussion on quantifying the error between an infinite dimensional LP and approximations based on finite dimensional restrictions we refer readers to the works by Hernández-Lerma and Lasserre (1998), Mohajerin Esfahani et al. (2015).

In the remainder of the paper we will evaluate the computational tractability and accuracy of (Fin-LP) in estimating reach-avoid value functions for a general subset of MDPs.

4. Radial Basis Functions for MDPs with Gaussian Mixture Kernels

For a general class of MDPs modeled by Gaussian mixture kernels (Khansari-Zadeh & Billard, 2011) we propose using Gaussian radial basis functions (GRBFs) for approximating the reach-avoid value functions. Through this choice, the constraint in (Fin-LP) involving the integration $\mathcal{T}_u[f]$ can be computed in closed form. Moreover, it is known that radial basis functions are a sufficiently rich function class to approximate continuous functions (Hartman, Keeler, & Kowalski, 1990; Sandberg, 2001; Park & Sandberg, 1991; Cybenko, 1989). In fact, Kveton and Hauskrecht (2006) also propose an algorithm for learning basis functions in the context of approximate linear programming using mean-parametrized GRBFs.

4.1 Basis Function Choice

To apply GRBFs in the reach-avoid framework, we consider the following problem data:

1. The kernel Q is a Gaussian mixture kernel $\sum_{j=1}^{J} \alpha_j \mathcal{N}(\mu_j, \Sigma_j)$ with diagonal covariance matrices Σ_j, means μ_j and weights α_j such that $\sum_{j=1}^{J} \alpha_j = 1$ for a finite $J \in \mathbb{N}_+$.

2. The target and safe sets K and K' are unions of disjoint hyper-rectangle sets, i.e. $K = \bigcup_{p=1}^{P} K_p = \bigcup_{p=1}^{P} \big(\bigtimes_{l=1}^{n} [a_l^p, b_l^p]\big)$ and $K' = \bigcup_{m=1}^{M} K'_m = \bigcup_{m=1}^{M} \big(\bigtimes_{l=1}^{n} [c_l^m, d_l^m]\big)$ for finite $P, M \in \mathbb{N}_+$ with $n = \dim(\mathcal{X})$ and $a^p, b^p, c^m, d^m \in \mathbb{R}^n$, $\forall p, m$.

The above restrictions apply to a large class of MDPs. For example, the kernel of general nonlinear systems subject to additive Gaussian mixture noise is a Gaussian mixture kernel. Moreover, in several problems, the state and input constraints are decoupled in different dimensions resulting in disjoint hyper-rectangles as constraint sets. It should be noted that whenever the safe and target sets are polytopic and cannot be written as unions of disjoint hyper-rectangles, one can approximate them as such to arbitrary accuracy using the methods suggested by Bemporad, Filippi, and Torrisi (2004). For a fixed approximation accuracy such algorithms are of polynomial complexity with respect to the dimension of the space. Approximations of more general sets with polytopes within a predefined accuracy is a much harder problem and algorithms may scale exponentially in the dimension of the problem (Bronstein, 2008).

For each time step k, let $\mathcal{F}^{M_k}$ denote the span of a set of GRBFs $\{\phi_i^k\}_{i=1}^{M_k} : \mathbb{R}^n \to \mathbb{R}$:

$$\phi_i^k(x) = \prod_{l=1}^{n} \frac{1}{\sqrt{2\pi s_{i,l}^k}} \exp\left(-\frac{1}{2}\frac{(x_l - c_{i,l}^k)^2}{s_{i,l}^k}\right), \tag{12}$$

where $\{c_{i,l}^k\}_{l=1}^n \in \mathbb{R}, \{s_{i,l}^k\}_{l=1}^n \in \mathbb{R}_+$ are the centers and the variances, respectively, of the GRBF. The class of GRBFs is closed with respect to multiplication (Hartman et al., 1990, Section 2). In particular, let $f^1 = \sum_{i=1}^{M_k} w_i^1 \phi_i^k$, $f^2 = \sum_{j=1}^{M_k} w_j^2 \phi_j^k$. Then, $f^1 f^2 = \sum_{i=1}^{M_k} \sum_{j=1}^{M_k} w_i^1 w_j^2 \tilde{\phi}_{ij}^k$, where the centers and variances of the bases $\tilde{\phi}_{ij}^k$ are explicit functions of those of ϕ_i^k, ϕ_j^k.

Integrating the proposed GRBFs over a union of hyper-rectangles decomposes into one dimensional integrals of Gaussian functions. In particular, let $\tilde{V}_k(x) = \sum_{i=1}^{M_k} \tilde{w}_i^k \phi_i^k(x)$ denote the approximate value function at time k and $A = \bigcup_{d=1}^{D}([a_1^d, b_1^d] \times \cdots \times [a_n^d, b_n^d])$, denote a finite union of hyper-rectangles. The integral of $\tilde{V}_k$ over A after some algebra reduces to

$$\int_A \tilde{V}_k(x)\nu(\mathrm{dx}) = \sum_{d=1}^{D}\sum_{i=1}^{M_k} \tilde{w}_i^k \prod_{l=1}^{n}\left(\frac{1}{2}\,\mathrm{erf}\left(\frac{b_l^d - c_{i,l}^k}{\sqrt{2 s_{i,l}^k}}\right) - \frac{1}{2}\,\mathrm{erf}\left(\frac{a_l^d - c_{i,l}^k}{\sqrt{2 s_{i,l}^k}}\right)\right), \tag{13}$$

where ν is assumed to be uniform product measure on each dimension d and erf denotes the error function defined as $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp\big(-t^2\big)\mathrm{d}t$.

Due to the decomposition of the reach-avoid value functions on the sets $K = \cup_{p=1}^{P} K_p$ and $\bar{\mathcal{X}} = K' \setminus K = (K' = \cup_{m=1}^{M} K'_m) \setminus K$ as stated in (4), $\mathcal{T}_u[\tilde{V}_k]$ in (5) is equivalent to

$$\int_{\mathcal{X}} \tilde{V}_k(y) Q(dy|x,u) = \sum_{m=1}^{M}\sum_{i=1}^{M_k} \tilde{w}_i^k \int_{K'_m} \phi_i^k(y) Q(dy|x,u) + \sum_{p=1}^{P} \int_{K_p} Q(dy|x,u). \tag{14}$$

Since a Gaussian mixture kernel Q can be written as a GRBF, every term inside the integral above is a product of GRBFs. Hence, it is a GRBF with known centers and variances. The integrals over K'_m and K_p in the right-hand-side of the (Fin-LP) can thus be computed using (13) at a sampled point (x^s, u^s).

4.2 Recursive Value Function and Policy Approximation

We summarize the method to approximate the reach-avoid value function in Algorithm 1. The design choices include the number of basis functions, their centers and variances, the sample violation and confidence bounds in Lemma 2 and the state-relevance weights. The number of basis functions is problem dependent and in our case studies, we use trial and error to fix this number. We choose the centers of the GRBFs by sampling them from a uniform probability measure supported on $\bar{\mathcal{X}}$. We sample the variances from a uniform measure supported on a bounded set that depends on problem data. Note that the method is still applicable if centers and variances are not sampled but set in another way, for example using neural network training or trial and error. Typically, ε and β are chosen to be close to 0 to enhance the feasibility guarantees of Lemma 2 at the expense of more constraints in (Fin-LP). Furthermore, we choose the state-relevance measure ν as a uniform product measure on the space $\bar{\mathcal{X}}$ to use the analytic integration in (13). This corresponds to equal weighting on potential errors on different state space regions.

Given the approximate value functions, we compute the so-called greedy control policy:

$$\tilde{\mu}_k(x) = \arg\max_{u \in \mathcal{U}} \int_{\mathcal{X}} \tilde{V}_{k+1}(y) Q(dy|x,u). \tag{15}$$

The optimization problem in (15) is non-convex. However, the cost function is smooth with respect to u for a fixed $x \in \bar{\mathcal{X}}$, the gradient and Hessian information can be analytically obtained using the erf function and the decision space $\mathcal{U}$ is typically low dimensional (in most mechanical systems for example, $\dim \mathcal{U} \leq \dim \mathcal{X}$). Thus, a locally optimal solution can be obtained efficiently using off-the-shelf optimization solvers.

5. Numerical Case Studies

We develop and solve a series of benchmark problems and evaluate our approximate solutions in two ways. First, we compute the closed-loop empirical reach-avoid policy by applying the approximated control input obtained from (15). Second, we use scalable alternative approaches to approximate the benchmark reach-avoid problems. To this end, we consider three reach-avoid problems that differ in structure and complexity. The first two examples are academic and illustrate the scalability and accuracy of the approach. The last example is a practical problem, where the approach was also implemented on a miniature race-car testbed. Throughout, we refer to our approach as the ADP approach. All computations were carried out on an Intel Core i7 Q820 CPU clocked at 1.73 GHz with 16GB of RAM memory, using IBM's CPLEX optimization toolkit in its default settings.

5.1 Example 1

We consider linear systems with additive Gaussian noise, $x_{k+1} = x_k + u_k + \omega_k$, where $x_k \in \mathcal{X} = \mathbb{R}^n$, $u_k \in \mathcal{U} = [-0.1, 0.1]^n$ and ω_k is distributed as a Gaussian random variable $\omega_k \sim \mathcal{N}(\mathbf{0}_{n\times 1}, \Sigma)$ with diagonal covariance matrix. We consider a target set $K = [-0.1, 0.1]^n$ centered at the origin and a safe set $K' = [-1, 1]^n$ (see Figure 1 for a 2D illustration). The objective is to reach the target set while staying in the safe set over a horizon of $T = 5$ steps. We approximated the value function using Algorithm 1 for a range of system

Algorithm 1 linear programming based reach-avoid value function approximation

```
Input Data:
    • State and control spaces $\bar{\mathcal{X}} \times \mathcal{U}$, reach-avoid time horizon $T$.
    • Target and safe sets $K$ and $K'$, written as unions of disjoint hyper-rectangles.
    • Centers and variances of the MDP Gaussian mixture kernel $Q$.
Design parameters:
    • Number of basis functions $\{M_k\}_{k=0}^{T-1}$.
    • Violation and confidence levels $\{\varepsilon_i\}_{i=0}^{T-1}$, $\{1-\beta_i\}_{i=0}^{T-1}$, probability measure $\mathbb{P}_{\bar{\mathcal{X}}\times\mathcal{U}}$.
    • Probability measure of centers and variances for the basis functions $\{\phi_i^k\}_{i=1}^{M_k}$.
    • State-relevance measure $\nu$ decomposed as a product measure on the state space.
Initialize $\tilde{V}_T(x) \leftarrow \mathbb{1}_K(x)$.
for $k = T-1, T-2, \dots, 0$ do
    Construct $\mathcal{F}^{M_k}$ by sampling $M_k$ centers $\{c_i\}_{i=1}^{M_k}$ and variances $\{s_i\}_{i=1}^{M_k}$ according to the chosen probability measures.
    Sample $N(\varepsilon_k, \beta_k, M_k)$ pairs $(x^s, u^s)$ from $\bar{\mathcal{X}} \times \mathcal{U}$ using the measure $\mathbb{P}_{\bar{\mathcal{X}}\times\mathcal{U}}$.
    for all $(x^s, u^s)$ do
        Evaluate $\mathcal{T}_{u^s}[\tilde{V}_{k+1}](x^s)$ using (14).
    end for
    Solve the finite LP in (Fin-LP) to obtain $\tilde{w}^k = (\tilde{w}_1^k, \dots, \tilde{w}_{M_k}^k)$.
    Set the approximated value function on $\bar{\mathcal{X}}$ to $\tilde{V}_k(x) = \sum_{i=1}^{M_k} \tilde{w}_i^k \phi_i^k(x)$.
end for
```

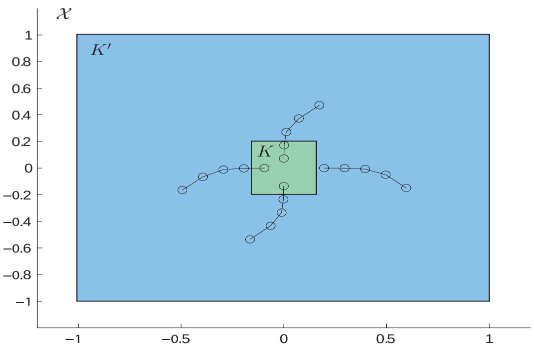


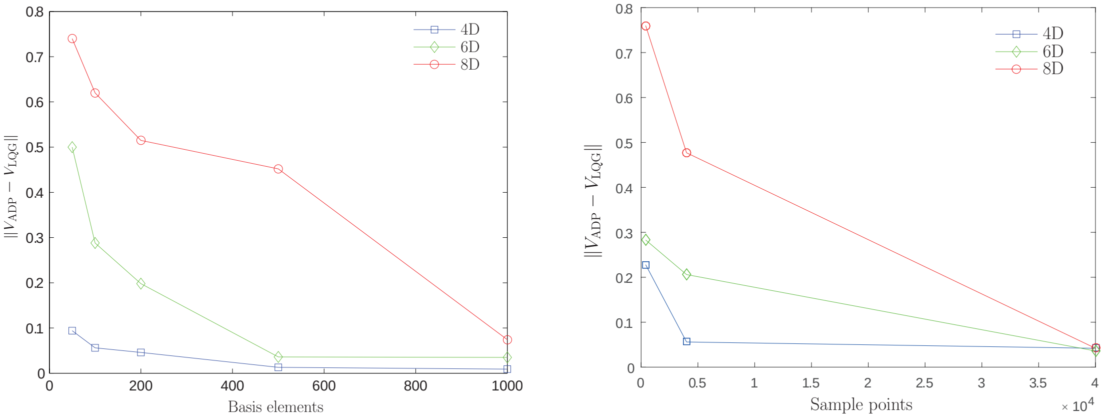
Figure 1: 2D depiction of safe and target sets and sample trajectories.

$\dim(\mathcal{X} \times \mathcal{U})$	4D	6D	8D
M_k	100	500	1000
N_k	4184	20184	40184
ε_k	0.05	0.05	0.05
$1-\beta_k$	0.99	0.99	0.99
$\\|\tilde{V}_0 - V_{\text{ADP}}\\|$	0.0692	0.104	0.224
Construction (sec)	4	85	450
LP solution (sec)	2	50	520
Memory (MB)	3.2	80	320

Table 1: Parameters and properties of the value function approximation scheme.

dimensions $\dim(\mathcal{X} \times \mathcal{U}) = 4, 6, 8$, to analyze scalability and accuracy of the LP-based reach-avoid solution in a benchmark problem that scales up in a straightforward way.

The transition kernel of the considered linear system is Gaussian $x_{k+1} \sim \mathcal{N}(x_k + u_k, \Sigma)$. The sets K and K' are hyper-rectangles. Thus, the GRBF framework applies. We chose 100, 500 and 1000 GRBF elements for the reach-avoid problems of $\dim(\mathcal{X} \times \mathcal{U}) = 4, 6, 8$, respectively (Table 1). We used uniform measures supported on $\bar{\mathcal{X}}$ and $[0.02, 0.095]^n$ to



(a) Mean absolute difference between the empirical reach-avoid probabilities achieved by the ADP (V_{ADP}) and LQG (V_{LQG}) policies as a function basis number. For each case of $\dim(\mathcal{X} \times \mathcal{U}) = 4, 6, 8$, the number of samples is kept constant over the different number of basis elements $50, 100, 200, 500, 1000$ and is equal to the numbers reported in Table 1.

(b) Mean absolute difference between the empirical reach-avoid probabilities achieved by the ADP (V_{ADP}) and LQG (V_{LQG}) policies as a function of number of samples. For each case of $\dim(\mathcal{X} \times \mathcal{U}) = 4, 6, 8$, the number of basis elements is kept constant over the different number of sample elements $400, 4000, 40000$ and is equal to the numbers reported in Table 1.

Figure 2: Example 1 - performance of the algorithm as a function of parameters.

sample the GRBFs' centers and variances, respectively. The violation and confidence levels for every $k \in \{0, \dots, 4\}$ were set to $\varepsilon_k = 0.05$, $1 - \beta_k = 0.99$ and the measure $\mathbb{P}_{\bar{\mathcal{X}} \times \mathcal{U}}$ required to generate samples from $\bar{\mathcal{X}} \times \mathcal{U}$ was chosen to be uniform. Since there is no reason to favor some states more than others, we also chose ν as a uniform measure, supported on $\bar{\mathcal{X}}$. Following Algorithm 1 we obtain a sequence of approximate value functions $\{\tilde{V}_k\}_{k=0}^{4}$.

To evaluate the performance of the approximation, we sampled 100 initial conditions x_0, uniformly from $\bar{\mathcal{X}}$. For each initial condition we generated 100 noise trajectories $\{\omega_k\}_{k=0}^{T-1}$. We computed the policy along the resulting state trajectory using (15). We then counted the number of trajectories that successfully completed the reach-avoid objective, i.e. reach K without leaving K' in T steps. In Table 1 we denote by $\|\tilde{V}_0 - V_{\mathrm{ADP}}\|$ the mean absolute difference between the empirical success denoted by V_{ADP}, and the predicted performance $\tilde{V}_0$, evaluated over the considered initial conditions. The memory and computation times reported correspond to constructing and solving each LP.

Since the system is linear, the noise is Gaussian and the target and safe sets are symmetric and centered around the origin, we can use the so-called Linear Quadratic Gaussian (LQG) controller (Bertsekas, 1995). This controller has the objective to drive the states close to the origin while ensuring the energy of the input is minimized. The closed-form optimal policy for the LQG problem can be easily computed (Bertsekas, 1995). As such, by properly tuning the corresponding weights of the states and inputs in the LQG objective based on the target and constraint sets, we can heuristically achieve the reach-avoid objective. This is further explained below.

The LQG problem for a linear stochastic system $x_{k+1} = Ax_k + Bu_k + \omega_k$, as the one considered above, is defined by an expected value quadratic cost function:

$$\min_{\{u_k\}_{k=0}^{T-1}} \mathbb{E}_{x_0}^{\mu}(\sum_{k=0}^{T-1} x_k^\top Q x_k + u_k^\top R u_k) + x_T^\top Q x_T.$$

Above, $Q \in \mathcal{S}_+^n$ and $R \in \mathcal{S}_{++}^m$, where $\mathcal{S}_+^n$ and $\mathcal{S}_{++}^m$ denote the set of $n \times n$ positive semidefinite and $m \times m$ positive definite matrices, respectively. We choose Q and R to correspond to the largest ellipsoids inscribed in K and $\mathcal{U}$, respectively. Through this choice the level sets of the LQG cost function proportionally correspond to the size of the target and control constraint sets. Intuitively, the penalization of states through the quadratic cost Q drives the state to the origin. The penalization of the input does not guarantee feasibility of the input constraints. Therefore, we project the LQG control on the feasible set $\mathcal{U}$. Using the same initial conditions and noise trajectories as those used with the ADP controller above, we simulated the performance of the LQG controller. We counted the number of trajectories that reach K without leaving K' over the horizon of $T = 5$ steps.

Figure 2a shows the mean over the initial conditions of the absolute difference between V_{LQG} and V_{ADP} as a function of number of basis functions. We observe a trend of increasing accuracy with increasing number of basis functions. Figure 2b shows the same metric but as a function of the total number of sample pairs from $\mathcal{X} \times \mathcal{U}$ for a fixed number of basis functions. Changing the number of samples N, affects the violation level ε_k (assuming constant β_k) and the approximation quality seems to improve with increasing N. In Table 2, we observe a trade-off between accuracy and computational time for the 6D problem varying the number of samples; the result is analogous in the 4D and 8D problems.

N	$\|V_{\text{ADP}} - V_{\text{LQG}}\|$	Construction (sec)	LP solution (sec)	Memory (MB)
400	0.283	2.20	3.57	1.60
4000	0.206	17.0	97.0	16.0
40000	0.036	170	162	160

Table 2: Accuracy and computation time as a function of number of sampled points in $\dim(\mathcal{X} \times \mathcal{U}) = 6$, with $M_k = 500$ and $1 - \beta_k = 0.99$.

5.2 Example 2

We consider the same linear dynamical system $x_{k+1} = x_k + u_k + \omega_k$, with target set K as defined in Section 5.1. In addition, in this example, the avoid set includes obstacles placed randomly within the state space as depicted in Figure 3. The safe set is $(K' \setminus \bigcup_{j=1}^5 K_\alpha^j)$, where K' was defined in the previous example, and each K_α^j denotes a hyper-rectangular obstacle. We denote the union of obstacle sets by $K_\alpha = \bigcup_{j=1}^5 K_\alpha^j$. The reach-avoid time horizon is $T = 7$. We use Algorithm 1 to approximate the optimal reach-avoid value function and compute the greedy policy.

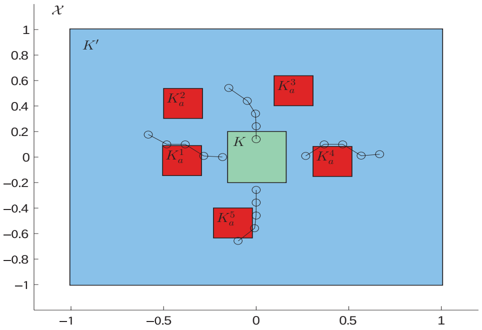


Figure 3: Example 2 - 2D depiction of obstacles and sample trajectories.

$\dim(\mathcal{X} \times \mathcal{U})$	4D	6D	8D
M_k	100	500	1000
N_k	4184	20184	40184
ε_k	0.05	0.05	0.05
$1 - \beta_k$	0.99	0.99	0.99
$\Vert\tilde{V}_0 - V_{\text{ADP}}\Vert$	0.095	0.118	0.191
Construction (sec)	4.20	130	671
LP solution (sec)	3.2	80	700
Memory (MB)	3.20	80.0	320

Table 3: Parameters and properties of the value function approximation scheme.

We chose the same basis function numbers, basis parameters, sampling and state-relevance measures as well as violation and confidence levels as in Section 5.1, shown in Table 3. We simulated the performance of the ADP controller starting from 100 different initial conditions, selected such that at least one obstacle blocks the direct path to the origin. For every initial condition we sampled 100 different noise trajectory realizations and applied the corresponding control policies computed through (15). We then computed the empirical ADP reach-avoid success probability (denoted by V_{ADP}) by counting the total number of trajectories that reach K while avoiding reaching the obstacles or leaving K'.

Note that due to the presence of multiple obstacles, the LQG approach cannot be used as a heuristic for comparison. Nevertheless, the problem of reaching a target set without passing through any obstacles is an instance of a path planning problem and has been studied thoroughly for deterministic systems (e.g., Borenstein & Koren, 1991; Richards & How, 2002; Van Den Berg, Ferguson, & Kuffner, 2006). For a benchmark comparison we use the approach developed by Richards and How (2002) and formulate the reach-avoid problem for the noise-free system as a constrained mixed logic dynamical system (MLD) (Bemporad & Morari, 1999). This problem can in turn be recast as a mixed integer quadratic program (MiQP) and solved to optimality using standard branch and bound techniques. To account for noise in the dynamics ω_k, we used a heuristic approach as follows. We truncated the density function of the random variables ω_k at 95% of their total mass and enlarged each obstacle set K_α by the maximum value of the truncated ω_k in each dimension. This resembles the robust (worst-case) approach to control design.

Starting from the same initial conditions as in the ADP approach, we simulated the performance of the MiQP-based control policy on the 100 trajectory realizations used in the ADP controller. We implemented the policy in receding horizon by measuring the state at each horizon step. The empirical success probability of trajectories that reach K while staying safe is denoted by V_{MiQP}. The mean difference $\Vert V_{\text{ADP}} - V_{\text{MiQP}}\Vert$ is presented in Table 4 and is computed by averaging the corresponding empirical reach-avoid success probabilities over the initial conditions. As seen in this table, as the number of basis functions increases, $\Vert V_{\text{ADP}} - V_{\text{MiQP}}\Vert$ decreases. This can indicate that the reach-avoid value function approximation is increasing in accuracy. Note that for an increase in the planning horizon T, the number of binary variables (and hence the computational complexity) in MiQP grows exponentially, whereas in the LP-based reach-avoid approach, the computation effort grows linearly with the horizon.

M_k	$\lVert V_{\text{ADP}} - V_{\text{MiQP}} \rVert$	Construction (sec)	LP solution (sec)	Memory (MB)
50	0.214	1.67	0.18	0.784
100	0.168	5.59	2.66	3.20
200	0.084	22.0	4.30	12.8
500	0.070	130	80.0	80.0
1000	0.045	507	1210	320

Table 4: Example 2 - Accuracy and computational requirements for $\dim(\mathcal{X} \times \mathcal{U}) = 6$.

5.3 Example 3

Consider the problem of driving a race car through a tight corner in the presence of static obstacles, illustrated in Figure 4. As part of the ORCA project of the Automatic Control Lab, a six state variable nonlinear model with two control inputs has been identified to describe the movement of 1:43 scale race cars. The model derivation can be found in the work by Liniger, Domahidi, and Morari (2014) and is based on a unicycle approximation with parameters identified on the experimental platform of the ORCA project using model cars manufactured by Kyosho. We denote the state space by $\mathcal{X} \subset \mathbb{R}^6$, the control space by $\mathcal{U} \subset \mathbb{R}^2$ and the identified dynamics by a function $f : \mathcal{X} \times \mathcal{U} \mapsto \mathcal{X}$. The first two elements of each state $x \in \mathcal{X}$ correspond to spatial dimensions, the third to orientation, the fourth and fifth to body fixed longitudinal and lateral velocities and the sixth to angular velocity. The two control inputs $u \in \mathcal{U}$ are the throttle duty cycle and the steering angle.

We will show how one can address the problem as a finite horizon reach-avoid problem and approximate its solution using the methodology presented. There are naturally several other approaches to address this problem (e.g., Richards & How, 2002; Couëtoux, Hoock, Sokolovska, Teytaud, & Bonnard, 2011). Our choice is only meant to illustrate the applicability of the framework for a general nonlinear dynamical system in high dimensions.

As typically observed in practice, the state predicted by the identified dynamics and the state measurements recorded on the experimental platform are different due to process and measurement noise. Analyzing the deviation between predictions and measurements,

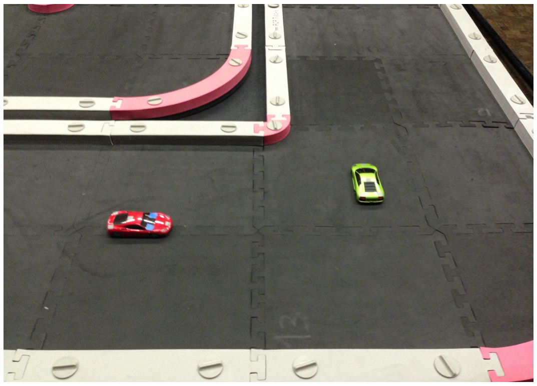


Figure 4: Example 3 - The set up of the Race-car cornering problem.

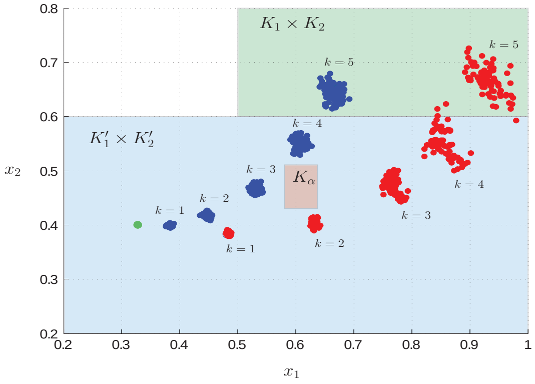


Figure 5: Example 3 - sample point clusters based on reach-avoid computation.

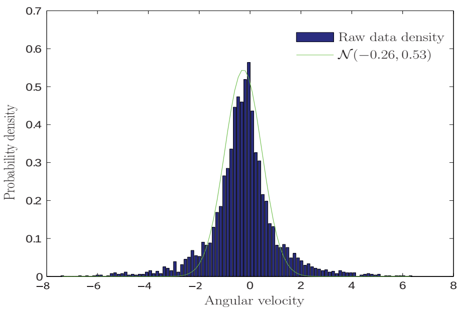


Figure 6: Empirical noise distribution.

Safe region	min	max	variances
K'_1 (m)	0.2	1	$[8 \times 10^{-4}$,$1.2 \times 10^{-3}]$
K'_2 (m)	0.2	0.6	$[8 \times 10^{-4}$,$1.2 \times 10^{-3}]$
K'_3 (rad)	$-\pi$	π	$[5 \times 10^{-3}$,$1.5 \times 10^{-2}]$
K'_4 (m/s)	0.3	3.5	$[5 \times 10^{-3}$,$1.5 \times 10^{-2}]$
K'_5 (m/s)	-1.5	1.5	$[5 \times 10^{-3}$,$1.5 \times 10^{-2}]$
K'_6 (rad/s)	-8	8	[2.00,4.00]

Table 5: State constraints and basis functions' variances used in ADP approximation.

we captured the uncertainties in the original model using additive Gaussian noise, $g(x, u) = f(x, u) + \omega, \quad \omega \sim \mathcal{N}(\mu, \Sigma)$, $\mu \in \mathbb{R}^6, \Sigma \in \mathcal{S}^6_{++}$, where $\mathcal{S}^6_{++}$ denotes the set of positive-definite matrices of dimension 6. The noise mean μ, and diagonal covariance matrix Σ have been selected such that the probability density function of the Markov decision process describing the uncertain dynamics resembles the empirical data obtained via measurements. As an example, Figure 6 illustrates the fit for the angular velocity where $\mu_6 = -0.26$ and $\Sigma(6, 6) = 0.53$. It follows that the kernel of the stochastic process is a GRBF with a single basis function described by the Gaussian distribution $\mathcal{N}(f(x, u) + \mu, \Sigma)$.

We cast the problem of driving the race car through a tight corner without reaching obstacles as a stochastic reach-avoid problem. Despite the highly nonlinear dynamics, the stochastic reach-avoid set-up can readily be applied to this problem. We consider a horizon of $T = 6$ and a sampling time of 0.08 seconds. The safe region of the spatial dimensions is defined as $(K'_1 \times K'_2)\backslash K_\alpha$ where $K_\alpha \subset \mathbb{R}^2$ denotes the obstacle, see Figures 4 and 5. The safe set in 6D is thus defined as $K' = ((K'_1 \times K'_2) \setminus K_\alpha) \times K'_3 \times K'_4 \times K'_5 \times K'_6$ where K'_3, K'_4, K'_5, K'_6 describe the physical limitations of the model car (see Table 5). Similarly, the target set for the spatial dimensions is denoted by $K_1 \times K_2$ and corresponds to the end of the turn as shown in Figure 5. The target set in 6D is then defined as $K = K_1 \times K_2 \times K'_3 \times K'_4 \times K'_5 \times K'_6$, which contains all states $x \in K'$ for which $(x_1, x_2) \in K_1 \times K_2$. The constraint sets are naturally decoupled over the state dimensions. Note that for practical purposes we have violated the assumption in Section 2 that the target set is a subset of the safe set in the spatial dimension (see Figure 5). The methodology and results remain the same if one extends the spatial safe set $K'_1 \times K'_2$ to include $K_1 \times K_2$.

We used 2000 GRBFs for each approximation step with centers and variances sampled according to uniform measures supported on $\bar{\mathcal{X}}$ and on the hyper-rectangle defined by the product of intervals in the rows of Table 5, respectively. We used a uniform state-relevance measure and a uniform sampling measure to construct each one of the finite linear programs in Algorithm 1. All violation and confidence levels were chosen to be $\varepsilon_k = 0.2$ and $1 - \beta_k = 0.99$ respectively for $k = \{0, \ldots, 5\}$. We then implemented the steps of Algorithm 1 and compute a sequence of approximate value functions.

To evaluate the quality of the approximations we initialized the car at two different initial conditions $x^1 = (0.33, 0.4, -0.2, 0.5, 0, 0)$ and $x^2 = (0.33, 0.4, -0.2, 2, 0, 0)$. They correspond to entering the corner at low ($x^1_4 = 0.5$ m/s) and high ($x^2_4 = 2$ m/s) longitudinal velocities. The approximate value functions evaluate to $\tilde{V}_0(x^1) = 0.98$, $\tilde{V}_0(x^2) = 1$ and indicate success with high probabilities for both cases. Interestingly, the associated trajectories computed via the greedy policy defined through (15) vary significantly. In the low velocity case, the car avoids the obstacle by driving above it while in the high velocity case, it does so by driving below it (see Figure 5). Such a behavior is expected since the car can slip if it turns aggressively at high velocities. We also computed empirical reach-avoid probabilities in simulation by sampling 100 noise trajectories from each initial state and implementing the ADP control policy using the associated value function approximation. The sample trajectories are plotted in Figure 5 with $V_{\mathrm{ADP}}(x^1) = 1$ and $V_{\mathrm{ADP}}(x^2) = 0.99$

The controller was tested on the ORCA setup by running 10 experiments from each initial condition[1]. We pre-computed the control inputs at the predicted mean trajectory of the states over the horizon for each experiment. Implementing the feedback policy online would require solving problem (15) within the sampling time of 0.08 seconds. In theory, this computation is possible since the control space is only two dimensional but it requires developing an embedded nonlinear programming solver compatible with the ORCA setup. Here, we have implemented the open loop controller. We note however that if the open loop controller performs accurately, the closed loop computation can only improve the performance by utilizing updated state measurements.

6. Conclusions

We developed a numerical approach to compute the value function of the stochastic reach-avoid problem for Markov decision processes with continuous state and action spaces. Since the method relies on solving linear programs we were able to tackle reach-avoid problems with larger dimensions than those addressed with established state-input space gridding methods. The potential of the approach was analyzed through two benchmark case studies and a trajectory planning problem for a six dimensional nonlinear system with two inputs. To the best of our knowledge, this is the first time that stochastic reach-avoid problems up to eight continuous state and input dimensions have been addressed.

We are currently focusing on the problem of systematically choosing the basis function parameters by exploiting knowledge about the system dynamics. Furthermore, we are developing decomposition methods for the large linear programs that arise in our approximation scheme to allow addressing control of MDPs in higher dimensions. An interesting research problem is to explore alternative tractable reformulations of the infinite constraints in the semi-infinite linear programs, using for example, constraint generation techniques or symbolic approaches (Kveton et al., 2006; Vianna, De Barros, & Sanner, 2015). Finally, given the close connections between reinforcement learning and approximate dynamic programming, it will be interesting to explore the possibility of using our stochastic reach-avoid formulation and results in developing inverse reinforcement learning algorithms. Such algorithms could incorporate safety and reachability objectives in addition to optimizing an additive reward or cost function.

[1] Please visit the YouTube channel of ETHZurichIfA for the video of the experiment

Acknowledgments

The work of N. Kariotoglou was supported by the Swiss National Science Foundation under grant number 200021_137876. The work of M. Kamgarpour is supported by the ERC Starting Grant CONENE. The authors would like to thank Alexander Liniger from the Automatic Control Laboratory in ETH Zürich for his help in testing the algorithm on the ORCA platform.

References

Abate, A., Amin, S., Prandini, M., Lygeros, J., & Sastry, S. (2007). Computational approaches to reachability analysis of stochastic hybrid systems. In *Hybrid Systems: Computation and Control*, pp. 4–17. Springer.

Abate, A., Prandini, M., Lygeros, J., & Sastry, S. (2008). Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, *44*(11), 2724–2734.

Anderson, E. J., & Nash, P. (1987). *Linear programming in infinite-dimensional spaces: theory and applications*. Wiley New York.

Bemporad, A., Filippi, C., & Torrisi, F. D. (2004). Inner and outer approximations of polytopes using boxes. *Computational Geometry*, *27*(2), 151–178.

Bemporad, A., & Morari, M. (1999). Control of systems integrating logic, dynamics, and constraints. *Automatica*, *35*(3), 407–427.

Ben-Tal, A., & Nemirovski, A. (2002). Robust optimization–methodology and applications. *Mathematical Programming*, *92*(3), 453–480.

Bertsekas, D. P., & Shreve, S. E. (1978). *Stochastic optimal control: The discrete time case*, Vol. 139. Academic Press NY.

Bertsekas, D. P. (1995). *Dynamic programming and optimal control*, Vol. 1. Athena scientific Belmont, MA.

Bertsekas, D. P., & Tsitsiklis, J. N. (1991). An analysis of stochastic shortest path problems. *Mathematics of Operations Research*, *16*, 580–595.

Bertsimas, D., Brown, D. B., & Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM review*, *53*(3), 464–501.

Borenstein, J., & Koren, Y. (1991). The vector field histogram-fast obstacle avoidance for mobile robots. *IEEE Transactions on Robotics and Automation*, *7*(3), 278–288.

Bronstein, E. M. (2008). Approximation of convex sets by polytopes. *Journal of Mathematical Sciences*, *153*(6), 727–762.

Brown, L. D., & Purves, R. (1973). Measurable selections of extrema. *The Annals of Statistics*, *1*(5), 902–912.

Calafiore, G. C., & Campi, M. C. (2006). The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, *51*(5), 742–753.

Campi, M. C., Garatti, S., & Prandini, M. (2009). The scenario approach for systems and control design. *Annual Reviews in Control*, *33*(2), 149–157.

Couëtoux, A., Hoock, J.-B., Sokolovska, N., Teytaud, O., & Bonnard, N. (2011). Continuous upper confidence trees. In *International Conference on Learning and Intelligent Optimization*, pp. 433–445. Springer.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, *2*(4), 303–314.

de Farias, D. P., & Van Roy, B. (2003). The linear programming approach to approximate dynamic programming. *Operations Research*, *51*(6), 850–865.

de Farias, D. P., & Van Roy, B. (2004). On constraint sampling in the linear programming approach to approximate dynamic programming. *Mathematics of Operations Research*, *29*(3), 462–478.

Ding, J., Kamgarpour, M., Summers, S., Abate, A., Lygeros, J., & Tomlin, C. (2013). A stochastic games framework for verification and control of discrete time stochastic hybrid systems. *Automatica*, *49*(9), 2665–2674.

Feinberg, E. A., Shwartz, A., & Altman, E. (2002). *Handbook of Markov decision processes: methods and applications*. Kluwer Academic Publishers Boston, MA.

Guestrin, C., Koller, D., Parr, R., & Venkataraman, S. (2003). Efficient solution algorithms for factored MDPs. *Journal of Artificial Intelligence Research*, *19*, 399–468.

Hartman, E. J., Keeler, J. D., & Kowalski, J. M. (1990). Layered neural networks with Gaussian hidden units as universal approximations. *Neural Computation*, *2*(2), 210–215.

Hauskrecht, M., & Kveton, B. (2003). Linear program approximations for factored continuous-state Markov decision processes.. In *NIPS*, pp. 895–902.

Hernández-Lerma, O., & Lasserre, J. B. (1996). *Discrete-time Markov control processes: basic optimality criteria*. Springer New York.

Hernández-Lerma, O., & Lasserre, J. B. (1998). Approximation schemes for infinite linear programs. *SIAM Journal on Optimization*, *8*(4), 973–988.

Hettich, R., & Kortanek, K. O. (1993). Semi-infinite programming: theory, methods, and applications. *SIAM review*, *35*(3), 380–429.

Kamgarpour, M., Summers, S., & Lygeros, J. (2013). Control Design for Property Specifications on Stochastic Hybrid Systems. In *Hybrid Systems: Computation and Control*, pp. 303–312. ACM.

Kariotoglou, N., Raimondo, D. M., Summers, S. J., & Lygeros, J. (2015). Multi-agent autonomous surveillance: a framework based on stochastic reachability and hierarchical task allocation. *Journal of Dynamic Systems, Measurement, and Control*, *137*(3), 031008.

Kariotoglou, N., Summers, S. J., Summers, T. H., Kamgarpour, M., & Lygeros, J. (2013). Approximate dynamic programming for stochastic reachability. In *IEEE European Control Conference*, pp. 584–589.

Kariotoglou, N., Margellos, K., & Lygeros, J. (2016). On the computational complexity and generalization properties of multi-stage and stage-wise coupled scenario programs. *Systems & Control Letters*, *94*, 63–69.

Khansari-Zadeh, S. M., & Billard, A. (2011). Learning stable nonlinear dynamical systems with gaussian mixture models. *IEEE Transactions on Robotics*, *27*(5), 943–957.

Kolobov, A., Mausam, & Weld, D. (2012). A theory of goal-oriented MDPs with dead ends. In *Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence (UAI12)*, pp. 438–447.

Kolobov, A., Mausam, M., Weld, D. S., & Geffner, H. (2011). Heuristic search for generalized stochastic shortest path MDPs. In *21st International Conference on Automated Planning and Scheduling*, pp. 130–137.

Kushner, H. J., & Dupuis, P. (2001). *Numerical methods for stochastic control problems in continuous time*, Vol. 24. Springer.

Kveton, B., Hauskrecht, M., & Guestrin, C. (2006). Solving factored MDPs with hybrid state and action variables. *Journal of Artificial Intelligence Research*, *27*(1), 153–201.

Kveton, B., & Hauskrecht, M. (2006). Learning basis functions in hybrid domains. In *Proceedings of the National Conference on Artificial Intelligence*, Vol. 21, pp. 1161–1166.

Liniger, A., Domahidi, A., & Morari, M. (2014). Optimization-based autonomous racing of 1: 43 scale RC cars. *Optimal Control Applications and Methods*, *36*, 628–647.

Mohajerin Esfahani, P., Sutter, T., & Lygeros, J. (2015). Performance bounds for the scenario approach and an extension to a class of non-convex programs. *IEEE Transactions on Automatic Control*, *60*(1), 46–58.

Nowak, A. (1985). Universally measurable strategies in zero-sum stochastic games. *The Annals of Probability*, *13*, 269–287.

Park, J., & Sandberg, I. W. (1991). Universal approximation using radial-basis-function networks. *Neural Computation*, *3*(2), 246–257.

Patrascu, R., Poupart, P., Schuurmans, D., Boutilier, C., & Guestrin, C. (2002). Greedy linear value-approximation for factored Markov decision processes. In *Proceedings of the 18th National Conference on Artificial Intelligence*, pp. 285–291.

Powell, W. B. (2007). *Approximate Dynamic Programming: Solving the curses of dimensionality*, Vol. 703. John Wiley & Sons.

Prandini, M., & Hu, J. (2006). Stochastic reachability: Theory and numerical approximation. *Stochastic hybrid systems, Automation and Control Engineering Series*, *24*, 107–138.

Puterman, M. L. (1994). *Markov decision processes: Discrete stochastic dynamic programming*. John Wiley & Sons, Inc.

Richards, A., & How, J. P. (2002). Aircraft trajectory planning with collision avoidance using mixed integer linear programming. In *Proceedings of the American Control Conference*, Vol. 3, pp. 1936–1941.

Sandberg, I. W. (2001). Gaussian radial basis functions and inner product spaces. *Circuits, Systems and Signal Processing*, *20*(6), 635–642.

Steinmetz, M., Hoffmann, J., & Buffet, O. (2016). Goal probability analysis in probabilistic planning: Exploring and enhancing the state of the art. *Journal of Artificial Intelligence Research*, *57*, 229–271.

Summers, S. J., Kamgarpour, M., Tomlin, C., & Lygeros, J. (2013). Stochastic system controller synthesis for reachability specifications encoded by random sets. *Automatica*, *49*(9), 2906–2910.

Summers, S. J., & Lygeros, J. (2010). Verification of discrete time stochastic hybrid systems: A stochastic reach-avoid decision problem. *Automatica*, *46*(12), 1951–1961.

Sundaram, R. K. (1996). *A first course in optimization theory*. Cambridge university press.

Van Den Berg, J., Ferguson, D., & Kuffner, J. (2006). Anytime path planning and replanning in dynamic environments. In *Proceedings IEEE International Conference on Robotics and Automation*, pp. 2366–2371.

Vianna, L. G., De Barros, L. N., & Sanner, S. (2015). Real-time symbolic dynamic programming for hybrid MDPs. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pp. 3402–3408. AAAI Press.

Watkins, O., & Lygeros, J. (2003). Stochastic reachability for discrete time systems: An application to aircraft collision avoidance. In *Proceedings of the IEEE Conference on Decision and Control*, Vol. 5, pp. 5314–5319.

Wood, T., Summers, S. J., & Lygeros, J. (2013). A stochastic reachability approach to emergency building evacuation. In *52nd IEEE Conference on Decision and Control (CDC), 2013*, pp. 5722–5727. IEEE.