Probabilities Are Not Enough: Formal Controller Synthesis for Stochastic Dynamical Models with Epistemic Uncertainty

Thom Badings[1], **Licio Romao**[2], **Alessandro Abate**[2], **Nils Jansen**[1]

[1] Radboud University, Nijmegen, the Netherlands
[2] University of Oxford, Oxford, United Kingdom
thom.badings@ru.nl, licio.romao@cs.ox.ac.uk, alessandro.abate@cs.ox.ac.uk, n.jansen@science.ru.nl

Abstract

Capturing uncertainty in models of complex dynamical systems is crucial to designing safe controllers. Stochastic noise causes aleatoric uncertainty, whereas imprecise knowledge of model parameters leads to epistemic uncertainty. Several approaches use formal abstractions to synthesize policies that satisfy temporal specifications related to safety and reachability. However, the underlying models exclusively capture aleatoric but not epistemic uncertainty, and thus require that model parameters are known precisely. Our contribution to overcoming this restriction is a novel abstraction-based controller synthesis method for continuous-state models with stochastic noise and uncertain parameters. By sampling techniques and robust analysis, we capture both aleatoric and epistemic uncertainty, with a user-specified confidence level, in the transition probability intervals of a so-called interval Markov decision process (iMDP). We synthesize an optimal policy on this iMDP, which translates (with the specified confidence level) to a feedback controller for the continuous model with the same performance guarantees. Our experimental benchmarks confirm that accounting for epistemic uncertainty leads to controllers that are more robust against variations in parameter values.

1 Introduction

Stochastic models. Stochastic dynamical models capture complex systems where the likelihood of transitions is specified by probabilities (Kumar and Varaiya 2015). Controllers for stochastic models must act safely and reliably with respect to a desired specification. Traditional control design methods use, e.g., Lyapunov functions and optimization to guarantee stability and (asymptotic) convergence. However, alternative methods are needed to give formal guarantees about richer temporal specifications relevant to, for example, safety-critical applications (Fan et al. 2022).

Finite abstractions. A powerful approach to synthesizing certifiably safe controllers leverages probabilistic verification to provide formal guarantees over specifications of *safety* (always avoid certain states) and *reachability* (reach a certain set of states). A common example is the *reach-avoid* specification, where the task is to maximize the probability of reaching desired goal states while avoiding unsafe states (Fisac et al. 2015). Finite abstractions can make continuous models amenable to techniques and tools from formal verification: by discretizing their state and action spaces, abstractions result in, e.g., finite Markov decision processes (MDPs) that soundly capture the continuous dynamics (Abate et al. 2008). Verification guarantees on the finite abstraction can thus carry over to the continuous model. In this paper, we adopt such an abstraction-based approach to controller synthesis.

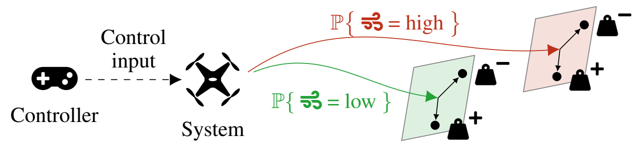

Figure 1: Aleatoric (stochastic) uncertainty in the wind ( ) causes probability distributions over the outcomes of controls, while epistemic uncertainty in the mass ( ) of the drone causes state transitions to be nondeterministic.

Probabilities are not enough. The notion of uncertainty is often distinguished in *aleatoric* (statistical) and *epistemic* (systematic) uncertainty (Fox and Ülkümen 2011; Sullivan 2015). Aleatoric uncertainty is caused by randomness, whereas epistemic uncertainty is caused by a lack of knowledge of, for example, system parameters (Smith 2013). In the absence of a prior distribution over these parameters, a purely probabilistic approach fails to capture epistemic uncertainty (Hüllermeier and Waegeman 2021). In this paper, we aim to reason under both aleatoric and epistemic uncertainty (without a prior distribution) to synthesize provably correct controllers for safety-critical applications. Existing abstraction methods fail to achieve this novel, general goal.

Models with epistemic uncertainty. We consider reach-avoid problems for stochastic dynamical models with continuous state and action spaces under epistemic uncertainty described by parameters that lie within a *convex uncertainty set*. In the simplest case, this uncertainty set is an interval, e.g., a drone whose mass is known to be between 0.75–$1.25\,\mathrm{kg}$. As shown in Fig. 1, the dynamics of the drone depend on uncertain factors, such as the wind and the drone's mass. Assume that for the wind, we can use weather data to estimate the likelihood of state dynamics, i.e., we can derive a probabilistic model. For the mass, however, we may not have information about the likelihood of each value, so employing a probabilistic model is unrealistic. Thus, we treat epistemic uncertainty in such imprecisely known parameters (in this case, the mass) using a *nondeterministic framework* instead.

Problem statement. Our goal is to synthesize a controller that (1) is *robust against nondeterminism* due to parameter uncertainty and (2) *reasons over probabilities* derived from stochastic noise. In other words, the controller must satisfy a given specification *under any possible outcome of the nondeterminism* (robustness) and *with at least a certain probability regarding the stochastic noise* (reasoning over probabilities). We wish to synthesize a controller with a *probably approximately correct (PAC)*-style guarantee to satisfy a reach-avoid specification with at least a desired threshold probability. Thus, we solve the following problem:

> **Problem.** Given a reach-avoid specification for a stochastic model with uncertain parameters, compute a controller and a *lower bound* on the probability that, *under any admissible value of the parameters*, the specification is probabilistically satisfied with this lower bound and *with at least a user-specified confidence probability*.

We solve this problem via a discrete-state abstraction of the continuous model. We generate this abstraction by partitioning the continuous state space and defining actions that induce potential transitions between elements of this partition.

Algorithmically, the closest approach to ours is Badings et al. (2022a), which uses abstractions to synthesize controllers for stochastic models with aleatoric uncertainty of unknown distribution, but with *known parameters*. Our setting is more general, as epistemic uncertainty requires fundamental differences to the technical approach, as explained below.

Robustness to capture nondeterminism. The main contribution that allows us to capture nondeterminism, is that we reason over *sets* of potential transitions (as shown by the boxes in Fig. 1), rather than *precise* transitions, e.g., as in Badings et al. (2022a). Intuitively, for a given action, the aleatoric uncertainty creates a probability distribution over *sets of possible outcomes*. To ensure robustness against epistemic uncertainty, we consider *all possible outcomes* within these sets. We show that, for our class of models, computing these sets of all possible outcomes is computationally tractable. Building upon this reasoning, we provide the following guarantees to solve the above-mentioned problem.

1) PAC guarantees on abstractions. We show that probabilities and nondeterminism can be captured in the transition probability intervals of so-called interval Markov decision processes (iMDPs, Givan, Leach, and Dean 2000). We use sampling methods from scenario optimization (Campi, Carè, and Garatti 2021) and concentration inequalities (Boucheron, Lugosi, and Massart 2013) to compute PAC bounds on these intervals. With a predefined confidence probability, the iMDP correctly captures both aleatoric and epistemic uncertainty.

2) Correct-by-construction control. For the iMDP, we compute a *robust optimal policy* that maximizes the worst-case probability of satisfying the reach-avoid specification. The iMDP policy is automatically translated to a controller for the original, continuous model on the fly. We show that, by construction, the PAC guarantees on the iMDP carry over to the satisfaction of the specification by the continuous model.

Contributions. We develop the first abstraction-based, formal controller synthesis method that simultaneously captures epistemic and aleatoric uncertainty for models with continuous state and action spaces. We also provide results on the PAC-correctness of obtained iMDP abstractions and guarantees on the synthesized controllers for a reach-avoid specification. Our numerical experiments in Sect. 6 confirm that accounting for epistemic uncertainty yields controllers that are more robust against deviations in the parameter values.

Related Work

Uncertainty models. Distinguishing aleatoric from epistemic uncertainty is a key challenge towards trustworthy AI (Thiebes, Lins, and Sunyaev 2021), and has been considered in reinforcement learning (Charpentier et al. 2022), Bayesian neural networks (Depeweg et al. 2018; Loquercio, Segù, and Scaramuzza 2020), and systems modeling (Smith 2013). Dynamical models with set-bounded (epistemic) parameter uncertainty (but *without aleatoric uncertainty*) are considered by Yedavalli (2014), and Geromel and Colaneri (2006). Control of models similar to ours is studied by (Modares 2022), albeit only for stability specifications.

Model-based approaches. Abstractions of stochastic models are well-studied (Abate et al. 2008; Alur et al. 2000), with applications to stochastic hybrid (Cauchi et al. 2019; Lavaei et al. 2022), switched (Lahijanian, Andersson, and Belta 2015), and partially observable systems (Badings et al. 2021; Haesaert et al. 2018). Various tools exist, e.g., StocHy (Cauchi and Abate 2019), ProbReach (Shmarov and Zuliani 2015), and SReachTools (Vinod, Gleason, and Oishi 2019). However, *these papers only reason over uncertainty in a probabilistic way but do not ensure robustness as we do.*

Fan et al. (2022) use optimization for reach-avoid control of linear but *non-stochastic* models with bounded disturbances. Barrier functions are used for cost minimization in stochastic optimal control (Pereira et al. 2020). So-called *funnel libraries* are used by Majumdar and Tedrake (2017) for robust feedback motion planning under set-bounded disturbances. Finally, Zikelic et al. (2022) learn policies together with formal reach-avoid certificates using neural networks for nonlinear systems with only stochastic uncertainty.

Data-driven approaches. Models with (partly) unknown dynamics express epistemic uncertainty about the underlying system. Verification of such models based on data has been done using Bayesian inference (Haesaert, den Hof, and Abate 2017), optimization (Kenanian et al. 2019; Vinod, Israel, and Topcu 2022), and Gaussian process regression (Jackson et al. 2020). Moreover, Knuth et al. (2021), and Chou, Ozay, and Berenson (2021) use neural network models for feedback motion planning for nonlinear deterministic systems with probabilistic safety and reachability guarantees. Data-driven abstractions have been developed for monotone (Makdesi, Girard, and Fribourg 2021) and event-triggered systems (Peruffo and Mazo 2023). By contrast to our setting, *these approaches consider models with non-stochastic dynamics*. A few recent exceptions exist (Salamati and Zamani 2022; Lavaei et al. 2023), albeit requiring more strict assumptions (e.g., discrete input sets) than our model-based approach.

Safe learning. While outside the scope of this paper, our approach fits naturally in a model-based safe learning context (Brunke et al. 2022; García and Fernández 2015). In such a setting, our approach may synthesize controllers that guarantee safe interactions with the system, while techniques from, for example, reinforcement learning (RL, Berkenkamp et al. 2017; Zanon and Gros 2021) or stochastic system identification (Tsiamis and Pappas 2019) can reduce the epistemic uncertainty based on state observations. A risk-sensitive RL scheme providing *approximate* safety guarantees is developed by Geibel and Wysotzki (2005); we instead give *formal* guarantees at the cost of an expensive abstraction.

2 Problem Statement

The cardinality of a discrete set $\mathcal{X}$ is $|\mathcal{X}|$. A probability space is a triple $(\Omega, \mathcal{F}, \mathbb{P})$ of an arbitrary set Ω, sigma algebra $\mathcal{F}$ on Ω, and probability measure $\mathbb{P} \colon \mathcal{F} \to [0,1]$. The convex hull of a polytopic set $\mathcal{X}$ with vertices $v_1, \dots, v_n$ is $\mathrm{conv}(v_1, \dots, v_n)$. The word *controller* relates to continuous models; a *policy* to discrete models.

Stochastic Models with Parameter Uncertainty

To capture parameter uncertainty in a linear time-invariant stochastic system, we consider a model (we extend this model with parameters describing uncertain additive disturbances in Sect. 5) whose continuous state x_k at time $k \in \mathbb{N}$ evolves as

$$x_{k+1} = A(\alpha)x_k + B(\alpha)u_k + \eta_k, \tag{1}$$

where $u_k \in \mathcal{U}$ is the control input, constrained by the convex polytope $\mathcal{U} = \mathrm{conv}(u^1, \dots, u^q) \subset \mathbb{R}^m$ with q vertices, and where the process noise η_k is a stochastic process defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Both the dynamics matrix $A(\alpha) \in \mathbb{R}^{n \times n}$ and the control matrix $B(\alpha) \in \mathbb{R}^{n \times m}$ are a convex combination of a finite set of $r \in \mathbb{N}$ known matrices:

$$A(\alpha) = \sum\nolimits_{i=1}^{r} \alpha_i A_i, \;\; B(\alpha) = \sum\nolimits_{i=1}^{r} \alpha_i B_i, \tag{2}$$

where the *unknown model parameter* $\alpha \in \Gamma$ can be any point in the unit simplex $\Gamma \subset \mathbb{R}^r$:

$$\Gamma = \Big\{ \alpha \in \mathbb{R}^r \colon \alpha_i \geq 0, \; \forall i \in \{1, \dots, r\}, \; \sum\nolimits_{i=1}^{r} \alpha_i = 1 \Big\}.$$

The model in Eq. (1) has set-bounded uncertain parameters through α (which we use to be robust against epistemic uncertainty) and is stochastic due to the process $(\eta_k)_{k \in \mathbb{N}}$ (which we use to reason probabilistically over aleatoric uncertainty).

Assumption 1. *The noise η_k is independent and identically distributed (i.i.d., which is a common assumption) and has density with respect to the Lebesgue measure. However, contrary to most definitions, we allow $\mathbb{P}$ to be* unknown.

Importantly, being distribution-free, our proposed techniques hold for any distribution of η that satisfies Assumption 1.

The matrices A_i and B_i can represent the bounds of intervals over parameters, as illustrated by the following example.

Example 1. *Consider again the drone of Fig. 1. The drone's longitudinal position p_k and velocity v_k are modeled as*

$$x_{k+1} = \begin{bmatrix} p_{k+1} \\ v_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & \tau \\ 0 & 1 - \frac{0.1\tau}{m} \end{bmatrix} x_k + \begin{bmatrix} \frac{\tau^2}{2m} \\ \frac{\tau}{m} \end{bmatrix} u_k + \eta_k,$$

with τ the discretization time, and $\mathcal{U} = [-5, 5]$. Assume that the mass m is only known to lie within $[0.75, 1.25]$. Then, we obtain a model as Eq. (1), with $r = 2$ vertices where A_1, B_1 are obtained for $m := 0.75$, and A_2, B_2 for $m := 1.25$. □

Reach-Avoid Planning Problem

The goal is to steer the state x_k of Eq. (1) to a desirable state within K time steps while always remaining in a safe region. Formally, let the *safe set* $\mathcal{Z}$ be a compact set of $\mathbb{R}^n$, and let $\mathcal{G} \subseteq \mathcal{Z}$ be the *goal set* (see Fig. 2). The control inputs u_k in Eq. (1) are selected by a time-varying *feedback controller*:

Definition 1. *A time-varying feedback* controller $c \colon \mathbb{R}^n \times \mathbb{N} \to \mathcal{U}$ *for Eq.* (1) *is a function that maps a state $x_k \in \mathbb{R}^n$ and a time step $k \in \mathbb{N}$ to a control input $u_k \in \mathcal{U}$.*

The space of admissible feedback controllers is denoted by $\mathcal{C}$. The *reach-avoid probability* $V(x_0, \alpha, c)$ is the probability that Eq. (1), under parameter $\alpha \in \Gamma$ and a controller c, satisfies a reach-avoid planning problem with respect to the sets $\mathcal{Z}$ and $\mathcal{G}$, starting from initial state $x_0 \in \mathbb{R}^n$. Formally:

Definition 2. *The* reach-avoid probability $V \colon \mathbb{R}^n \times \Gamma \times \mathcal{C} \to [0,1]$ *for a given controller $c \in \mathcal{C}$ on horizon K is*

$$\begin{aligned} V(x_0, \alpha, c) = \mathbb{P}\big\{ \eta_k \in \Omega : \; & x_k \textit{ evolves as per Eq. } (1), \\ & \exists k' \in \{0, \dots, K\} \textit{ such that } x_{k'} \in \mathcal{G}, \textit{ and} \\ & x_k \in \mathcal{Z}, \, u_k = c(x_k, k) \; \forall k \in \{0, \dots, k'\} \big\}. \end{aligned}$$

We aim to find a controller for which the reach-avoid probability is above a certain threshold $\lambda \in [0,1]$. However, since parameter α of Eq. (1) is unknown, it is impossible to compute the reach-avoid probability explicitly. We instead take a *robust approach*, thus stating the formal problem as follows:

Formal problem. *Given an initial state x_0, find a controller $c \in \mathcal{C}$ together with a (high) probability threshold $\lambda \in [0,1]$, such that $V(x_0, \alpha, c) \geq \lambda$ holds for all $\alpha \in \Gamma$.*

Thus, we want to find a controller $c \in \mathcal{C}$ that is robust against *any possible instance* of parameter $\alpha \in \Gamma$ in Eq. (1). Due to the aleatoric uncertainty in Eq. (1) of unknown distribution, we solve the problem up to a user-specified confidence probability $\beta \in (0,1)$, as we shall see in Sect. 5.

Interval Markov Decision Processes

We solve the problem by generating a finite-state abstraction of the model as an interval Markov decision process (iMDP):

Definition 3. *An* iMDP *is a tuple $\mathcal{M}_{\mathbb{I}} = (S, Act, s_I, \mathcal{P})$ where S is a finite set of states, Act is a finite set of actions, $s_I \in S$ is the initial state, and $\mathcal{P} \colon S \times Act \times S \to \mathbb{I} \cup \{[0,0]\}$ is an uncertain partial probabilistic transition function over intervals $\mathbb{I} = \{[a,b] \mid a, b \in (0,1] \text{ and } a \leq b\}$.*

Note that an interval may not have a lower bound of 0, except for the $[0,0]$ interval. If $\mathcal{P}(s,a,s') = [0,0] \, \forall s' \in S$, action $a \in Act$ is not enabled in state $s \in S$. We can instantiate

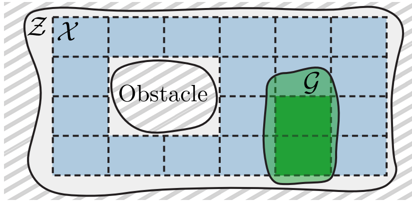

Figure 2: Partition of safe set $\mathcal{X} \subseteq \mathcal{Z}$ (which excludes obstacles) into regions that define iMDP states. A state s_i is a goal state, $s_i \in S_\mathcal{G}$, if its region is contained in the goal region $\mathcal{G}$.

an iMDP into an MDP with a partial transition function $P\colon S \times Act \rightharpoonup Dist(S)$ by fixing a probability $P(s,a)(s') \in \mathcal{P}(s,a,s')$ for each $s, s' \in S$ and for each $a \in Act$ enabled in s, such that $P(s,a)$ is a probability distribution over S. For brevity, we write this instantiation as $P \in \mathcal{P}$ and denote the resulting MDP by $\mathcal{M}_\mathbb{I}[P]$.

A deterministic policy (Baier and Katoen 2008) for an iMDP $\mathcal{M}_\mathbb{I}$ is a function $\pi\colon S^* \to Act$, where S^* is a sequence of states (memoryless policies do not suffice for time-bounded specifications), with $\pi \in \Pi_{\mathcal{M}_\mathbb{I}}$ the admissible policy space. For a policy π and an instantiated MDP $\mathcal{M}_\mathbb{I}[P]$ for $P \in \mathcal{P}$, we denote by $Pr^\pi(\mathcal{M}_\mathbb{I}[P] \models \varphi_{s_I}^K)$ the probability of satisfying a reach-avoid specification[1] $\varphi_{s_I}^K$ (i.e., reaching a goal in $S_\mathcal{G} \subseteq S$ within $K \in \mathbb{N}$ steps, while not leaving a safe set $S_\mathcal{Z} \subseteq S$). A *robust optimal policy* $\pi^\star \in \Pi_{\mathcal{M}_\mathbb{I}}$ *maximizes* this probability under the *minimizing* instantiation $P \in \mathcal{P}$:[2]

$$\pi^\star = \operatorname*{arg\,max}_{\pi \in \Pi} \min_{P \in \mathcal{P}} Pr^\pi(\mathcal{M}_\mathbb{I}[P] \models \varphi_{s_I}^K). \tag{3}$$

We compute an optimal policy in Eq. (3) using a robust variant of value iteration proposed by Wolff, Topcu, and Murray (2012). Note that deterministic policies suffice to obtain optimal values for Eq. (3), see Puggelli et al. (2013).

3 Finite-State Abstraction

To solve the formal problem, we construct a finite-state abstraction of Eq. (1) as an iMDP. We define the actions of this iMDP via backward reachability computations on a so-called *nominal model* that neglects any source of uncertainty. We then compensate for the error caused by this modeling simplification in the iMDP's transition probability intervals.

Nominal Model of the Dynamics

To build our abstraction, we rely on a *nominal model* that neglects both the aleatoric and epistemic uncertainty in Eq. (1), and is thus deterministic. Concretely, we fix any value $\hat{\alpha} \in \Gamma$ and define the nominal model dynamics as

$$\hat{x}_{k+1} = A(\hat{\alpha})x_k + B(\hat{\alpha})u_k. \tag{4}$$

Due to the linearity of the dynamics, we can now express the successor state x_{k+1} *with full uncertainty*, from Eq. (1), as

$$x_{k+1} = \hat{x}_{k+1} + \delta(\alpha, x_k, u_k) + \eta_k, \tag{5}$$

[1]Note that $\varphi_{s_I}^K$ is a reach-avoid specification for an iMDP, while $V(x_0, \alpha, c)$ is the reach-avoid probability on the dynamical model.

[2]Such *min-max* (and *max-min*) problems are common in (distributionally) robust optimization, see Ben-Tal, Ghaoui, and Nemirovski (2009), and Wiesemann, Kuhn, and Sim (2014) for details.

with $\delta(\alpha, x_k, u_k)$ being a new term, called the *epistemic error*, encompassing the error caused by parameter uncertainty:

$$\delta(\alpha, x_k, u_k) = [A(\alpha) - A(\hat{\alpha})]x_k + [B(\alpha) - B(\hat{\alpha})]u_k. \tag{6}$$

In other words, the successor state x_{k+1} is the nominal one, plus the epistemic error, and plus the stochastic noise. Note that for $\alpha = \hat{\alpha}$ (i.e., the true model parameters equal their nominal values), we obtain $\delta(\alpha, x_k, u_k) = 0$. We also impose the next assumption on the nominal model, which guarantees that we can compute the inverse image of Eq. (4) for a given $\hat{x}_{k+1}$, and is used in the proof of Lemma 3 in the extended version of this paper (Badings et al. 2022b, Appendix A).

Assumption 2. *The matrix $A(\hat{\alpha})$ in Eq.* (4) *is non-singular.*

Assumption Assumption 2 is mild as it only requires the existence of a non-singular matrix in $\operatorname{conv}\{A_1, \dots, A_r\}$.

States

We create a partition of a subset $\mathcal{X}$ of the safe set $\mathcal{Z}$ on the continuous state space, see Fig. 2. This partition fully covers the goal set $\mathcal{G}$ but excludes unsafe states, i.e., any $x \notin \mathcal{Z}$.

Definition 4. *A finite collection of subsets $(\mathcal{P}_i)_{i=1}^L$ is called a* partition *of $\mathcal{X} \subseteq \mathcal{Z} \subset \mathbb{R}^n$ if the following conditions hold:*

1. $\mathcal{X} = \bigcup_{i=1}^L \mathcal{P}_i$,
2. $\mathcal{P}_i \bigcap \mathcal{P}_j = \emptyset,\ \forall i, j \in \{1, \dots, L\},\ i \neq j.$

We append to the partition a so-called *absorbing region* $\mathcal{P}_0 = \operatorname{cl}(\mathbb{R}^n \setminus \mathcal{X})$, which is defined as the closure of $\mathbb{R}^n \setminus \mathcal{X}$ and represents any state $x \notin \mathcal{X}$ that is disregarded in subsequent reachability computations. We consider partitions into convex polytopic regions, which will allow us to compute PAC probability intervals in Lemma 1 using results from Romao, Papachristodoulou, and Margellos (2022) on scenario optimization programs with discarded constraints:

Assumption 3. *Each region $\mathcal{P}_i$ is a convex polytope given by*

$$\mathcal{P}_i = \{x \in \mathbb{R}^n \colon H_i x \leq h_i\}, \tag{7}$$

with $H_i \in \mathbb{R}^{p_i \times n}$ and $h_i \in \mathbb{R}^{p_i}$ for some $p_i \in \mathbb{N}$, and the inequality in Eq. (7) *is to be interpreted element-wise.*

We define an iMDP state for each element of $(\mathcal{P}_i)_{i=0}^L$, yielding a set of $L+1$ discrete states $S = \{s_i \mid i = 0, \dots, L\}$. Define $T\colon \mathbb{R}^n \to \{0, 1, \dots, L\}$ as the map from any $x \in \mathcal{X}$ to its corresponding region index i. We say that a continuous state x belongs to iMDP state s_i if $T(x) = i$. State s_0 is a deadlock, such that the only transition leads back to s_0.

Actions

Recall that we define the iMDP actions via backward reachability computations under the nominal model in Eq. (4). Let $\mathcal{T} = \{\mathcal{T}_1, \dots, \mathcal{T}_M\}$ be a finite collection of *target sets*, each of which is a convex polytope, $\mathcal{T}_\ell = \operatorname{conv}\{t^1, \dots, t^d\} \subset \mathbb{R}^n$. Every target set corresponds to an iMDP action, yielding the set $Act = \{a_\ell \mid \ell = 1, \dots, M\}$ of actions. Action $a_\ell \in Act$ represents a transition to $\hat{x}_{k+1} \in \mathcal{T}_\ell$ that is feasible under the nominal model. The one-step *backward reachable set* $\mathcal{R}_{\hat{\alpha}}^{-1}(\mathcal{T}_\ell)$, shown in Fig. 3, represents precisely these continuous states from which such a direct transition to $\mathcal{T}_\ell$ exists:

$$\mathcal{R}_{\hat{\alpha}}^{-1}(\mathcal{T}_\ell) = \big\{x \in \mathbb{R}^n \mid \exists u \in \mathcal{U},\ A(\hat{\alpha})x + B(\hat{\alpha})u \in \mathcal{T}_\ell\big\}.$$

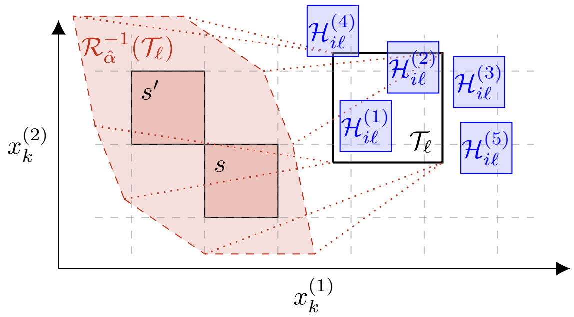

Figure 3: Action a_ℓ to target set $\mathcal{T}_\ell$ is enabled in states $s, s' \in S$, as their regions are contained in the backward reachable set $\mathcal{R}_{\hat{\alpha}}^{-1}(\mathcal{T}_\ell)$. The successor state sets $\mathcal{H}_{i\ell}^{(1)}, \dots, \mathcal{H}_{i\ell}^{(5)}$ for five noise samples, overapproximated as boxes, are shown in blue.

Intuitively, we enable action a_ℓ in state s_i only if $\hat{x}_{k+1} \in \mathcal{T}_\ell$ can be realized from any $x_k \in \mathcal{P}_i$ in the associated region, or in other words, region $\mathcal{P}_i$ must be contained in the backward reachable set. As such, we obtain the following definition:

Definition 5. *Given a fixed $\hat{\alpha} \in \Gamma$, an action $a_\ell \in Act$ is enabled in a state $s_i \in S$ if $\mathcal{P}_i \subseteq \mathcal{R}_{\hat{\alpha}}^{-1}(\mathcal{T}_\ell)$. The set $Act_{\hat{\alpha}}(s_i)$ of enabled actions in state s_i under $\hat{\alpha} \in \Gamma$ is defined as*

$$Act_{\hat{\alpha}}(s_i) = \{a_\ell \in Act \colon \mathcal{P}_i \subseteq \mathcal{R}_{\hat{\alpha}}^{-1}(\mathcal{T}_\ell)\}. \tag{8}$$

Computing backward reachable sets. To apply Def. 5, we must compute $\mathcal{R}_{\hat{\alpha}}^{-1}(\mathcal{T}_\ell)$ for each action $a_\ell \in Act$ with associated target set $\mathcal{T}_\ell$. In the extended version of this paper (Badings et al. 2022b, Appendix A, Lemma 3), we show that $\mathcal{R}_{\hat{\alpha}}^{-1}(\mathcal{T}_\ell)$ is a polytope characterized by the vertices of $\mathcal{U}$ and $\mathcal{T}_\ell$, which is computationally tractable to compute.

Transition Probability Intervals

We wish to compute the probability $P(s_i, a_\ell)(s_j)$ that taking action $a_\ell \in Act$ in a continuous state $x_k \in \mathcal{P}_i$ yields a successor state $x_{k+1} \in \mathcal{P}_j$ that belongs to state $s_j \in S$. This conditional probability is defined using Eq. (5) as

$$P(s_i, a_\ell)(s_j) = \mathbb{P}\{x_{k+1} \in \mathcal{P}_j \mid a_\ell \in Act_{\hat{\alpha}}(s_i)\}. \tag{9}$$

Note the outcome of an action a_ℓ is the same for any origin state s_i in which a_ℓ is enabled. In the remainder, we thus drop the conditioning on $a_\ell \in Act_{\hat{\alpha}}(s_i)$ in Eq. (9) for brevity.

Two factors prevent us from computing Eq. (9): 1) the nominal successor state $\hat{x}_{k+1}$ and the term $\delta(\alpha, x_k, u_k)$ are nondeterministic, and 2) the distribution of the noise η_k is unknown. We deal with the former in the following paragraph while addressing the stochastic noise in Sect. 4.

Capturing nondeterminism. As a key step, we capture the nondeterminism caused by epistemic uncertainty. First, recall that by construction, we have $\hat{x}_{k+1} \in \mathcal{T}_\ell$. Second, we write the set Δ_i of all possible epistemic errors in Eq. (6) as

$$\Delta_i = \{\delta(\alpha, x_k, u_k) \colon \alpha \in \Gamma, x_k \in \mathcal{P}_i, u_k \in \mathcal{U}\}. \tag{10}$$

Based on these observations, we obtain that the successor state x_{k+1} is an element of a set that we denote by $\mathcal{H}_{i\ell}$:

$$x_{k+1} \in \mathcal{T}_\ell + \Delta_i + \eta_k = \mathcal{H}_{i\ell}. \tag{11}$$

Crucially, we show in Lemma 4 of the extended version of this paper (Badings et al. 2022b, Appendix A) that we can compute an *overapproximation of* Δ_i based on sets $\mathcal{P}_i$ and $\mathcal{U}$, and the model dynamics. Based on the set $\mathcal{H}_{i\ell}$, we bound the probability in Eq. (9) as follows:

$$\mathbb{P}\{\mathcal{H}_{i\ell} \subseteq \mathcal{P}_j\} \leq \mathbb{P}\{x_{k+1} \in \mathcal{P}_j\} \leq \mathbb{P}\{\mathcal{H}_{i\ell} \cap \mathcal{P}_j \neq \emptyset\}. \tag{12}$$

Both inequalities follow directly by definition of Eq. (11). The lower bound holds, since if $\mathcal{H}_{i\ell} \subseteq \mathcal{P}_j$, then $x_{k+1} \in \mathcal{P}_j$ for any $x_{k+1} \in \mathcal{H}_{i\ell}$. The upper bound holds, since by Eq. (11) we have that $x_{k+1} \in \mathcal{H}_{i\ell}$, and thus, if $x_{k+1} \in \mathcal{P}_j$, then the intersection $\mathcal{H}_{i\ell} \cap \mathcal{P}_j$ must be nonempty.

4 PAC Probability Intervals via Sampling

The interval in Eq. (12) still depends on the noise η_k, whose density function is unknown. We show how to compute PAC bounds on this interval, by sampling a set of $N \in \mathbb{N}$ samples of the noise, denoted by $\eta_k^{(1)}, \dots, \eta_k^{(N)}$. Recall from Assumption 1 that these sample are i.i.d. elements from $(\Omega, \mathcal{F}, \mathbb{P})$. Each sample $\eta_k^{(\iota)}$ yields a set $\mathcal{H}_{i\ell}^{(\iota)}$ (see Fig. 3) that contains the successor state under that value of the noise, i.e.,

$$x_{k+1}^{(\iota)} \in \mathcal{T}_\ell + \Delta_i + \eta_k^{(\iota)} = \mathcal{H}_{i\ell}^{(\iota)}. \tag{13}$$

For reasons of computational performance, we overapproximate each set $\mathcal{H}_{i\ell}^{(\iota)}$ as the smallest hyperrectangle in $\mathbb{R}^n$, by taking the point-wise min. and max. over the vertices of $\mathcal{H}_{i\ell}^{(\iota)}$.

Lower Bounds from the Scenario Approach

We interpret the lower bound in Eq. (12) within the *sampling-and-discarding* scenario approach (Campi and Garatti 2011). Concretely, let $R \subseteq \{1, \dots, N\}$ be a subset of the noise samples and consider the following convex program:

$$\begin{aligned} \mathfrak{L}_R \colon \ & \underset{\lambda \geq 0}{\text{minimize}} \ \lambda \\ & \text{subject to} \ \ \mathcal{H}_{i\ell}^{(\iota)} \subseteq \mathcal{P}_j(\lambda) \quad \forall \iota \in R, \end{aligned} \tag{14}$$

where $\mathcal{P}_j(\lambda)$ is a version of $\mathcal{P}_j$ scaled by a factor λ around an arbitrary point $x \in \mathcal{P}_j$, such that $\mathcal{P}_j(0) = x$, and $\mathcal{P}_j(\lambda_1) < \mathcal{P}_j(\lambda_2)$ for $\lambda_1 < \lambda_2$; see Badings et al. (2022a, Appendix A) for details. The optimal solution $\lambda_R^\star$ to $\mathfrak{L}_R$ results in a region $\mathcal{P}_j(\lambda_R^\star)$ such that, for all $\iota \in R$, the set $\mathcal{H}_{i\ell}^{(\iota)}$ for noise sample $\eta_k^{(\iota)}$ is contained in $\mathcal{P}_j(\lambda_R^\star)$. We ensure that $\mathcal{P}_j(\lambda_R^\star) \subseteq \mathcal{P}_j$, by choosing R as the set of samples being a subset of $\mathcal{P}_j$, i.e.,

$$R := \{\iota \in \{1, \dots, N\} \colon \mathcal{H}_{i\ell}^{(\iota)} \subseteq \mathcal{P}_j\}, \tag{15}$$

We use the results in Romao, Papachristodoulou, and Margellos (2022, Theorem 5) to lower bound the probability that a random sample $\eta_k \in \Omega$ yields $\mathcal{H}_{i\ell} \subseteq \mathcal{P}_j(\lambda_R^\star)$. This leads to the following lower bound on the transition probability:[3]

Lemma 1. *Fix a region $\mathcal{P}_j$ and confidence probability $\beta \in (0, 1)$. Given sets $(\mathcal{H}_{i\ell}^{(\iota)})_{\iota=1}^N$, compute R. Then, it holds that*

$$\mathbb{P}^N \Big\{ \mathbb{P}\{\eta_k \in \Omega \colon \mathcal{H}_{i\ell} \subseteq \mathcal{P}_j\} \geq \underline{p} \Big\} \geq 1 - \beta, \tag{16}$$

[3] One can readily show that the technical requirements stated in Romao, Papachristodoulou, and Margellos (2022) are satisfied for the scenario program Eq. (14). Details are omitted here for brevity.

where $\underline{p} = 0$ *if* $|R| = 0$*, and otherwise,* $\underline{p}$ *is the solution to*

$$\frac{\beta}{N} = \sum\nolimits_{i=0}^{N-|R|} \binom{N}{i} (1-\underline{p})^i \underline{p}^{N-i}. \tag{17}$$

More details and the proof of Lemma 1 are in the extended version, Badings et al. (2022b, Appendix A).

Upper Bounds via Hoeffding's Inequality

The scenario approach might lead to conservative estimates of the upper bound in Eq. (12); see Badings et al. (2022b, Appendix A) for details. Thus, we instead apply Hoeffding's inequality (Boucheron, Lugosi, and Massart 2013) to infer an upper bound $\bar{p}$ of the probability $\mathbb{P}\{\mathcal{H}_{i\ell} \cap \mathcal{P}_j \neq \emptyset\}$. Concretely, this probability describes the parameter of a Bernoulli random variable, which has value 1 if $\mathcal{H}_{i\ell} \cap \mathcal{P}_j \neq \emptyset$ and 0 otherwise. The sample sum $\tilde{R}$ of this random variable is given by the number of sets $\mathcal{H}_{i\ell}^{(\iota)}$ that intersect with region $\mathcal{P}_j$, i.e.,

$$\tilde{R} \coloneqq \big\{\iota \in \{1, \dots, N\} \colon \mathcal{H}_{i\ell}^{(\iota)} \cap \mathcal{P}_j \neq \emptyset\big\}. \tag{18}$$

Using Hoeffding's inequality, we state the following lemma to bound the upper bound transition probability in Eq. (12).

Lemma 2. *Fix region* $\mathcal{P}_j$ *and confidence probability* $\beta \in (0,1)$*. Given sets* $(\mathcal{H}_{i\ell}^{(\iota)})_{\iota=1}^N$*, compute* $\tilde{R}$*. Then, it holds that*

$$\mathbb{P}^N \Big\{ \mathbb{P}\{\eta_k \in \Omega \colon \mathcal{H}_{i\ell} \cap \mathcal{P}_j \neq \emptyset\} \leq \bar{p} \Big\} \geq 1 - \beta, \tag{19}$$

where the upper bound $\bar{p}$ *is computed as*

$$\bar{p} = \min \left\{ 1, \; \frac{\tilde{R}}{N} + \sqrt{\frac{1}{2N} \log\left(\frac{1}{\beta}\right)} \right\}. \tag{20}$$

We provide the proof in Badings et al. (2022b, Appendix A).

Probability Intervals with PAC Guarantees

We apply Lemmas 1 and 2 as follows to compute a PAC probability interval for a specific transition of the iMDP.

Theorem 1 (PAC probability interval)**.** *Fix a region* $\mathcal{P}_j$ *and a confidence probability* $\beta \in (0,1)$*. For the collection* $(\mathcal{H}_{i\ell}^{(\iota)})_{\iota=1}^N$*, compute* $\underline{p}$ *and* $\bar{p}$ *using Lemmas 1 and 2. Then, the transition probability* $P(s_i, a_\ell)(s_j)$ *is bounded by*

$$\mathbb{P}^N \Big\{ \underline{p} \leq P(s_i, a_\ell)(s_j) \leq \bar{p} \Big\} \geq 1 - 2\beta. \tag{21}$$

Proof. Theorem 1 follows directly by combining Lemmas 1 and 2 via the union bound[4] with the probability interval in Eq. (12), which asserts that these bounds are both correct with a probability of at least $1 - 2\beta$. □

Counting samples. The inputs to Theorem 1 are the sample counts R (fully contained in $\mathcal{P}_j$) and $\tilde{R}$ (at least partially contained in $\mathcal{P}_j$), and the confidence probability β. Thus, the problem of computing probability intervals reduces to a *counting problem* on the samples. In Badings et al. (2022b, Appendix B), we describe a procedure that significantly speeds up this counting process in a sound way by merging (and overapproximating) sets $\mathcal{H}^{(\iota)}$ that are very similar.

[4]The union bound (Boole's inequality) states that the probability that at least one of a finite set of events happens, is upper bounded by the sum of these events' probabilities (Casella and Berger 2021).

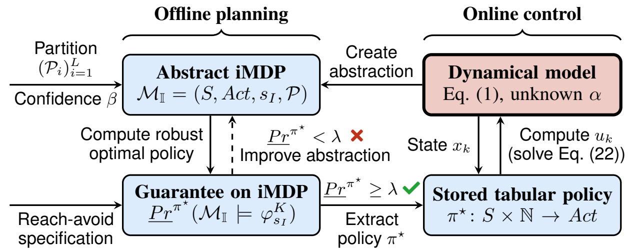

Figure 4: Our overall approach to solve the formal problem based on the abstraction method outlined in Sects. 3 and 4.

5 Overall Abstraction Method

We provide an algorithm to solve the formal problem based on the proposed abstraction method. The approach is shown in Fig. 4 and consists of an *offline planning* phase, in which we create the iMDP and compute a robust optimal policy, and an *online control* phase in which we automatically derive a provably-correct controller for the continuous model.

1) Create abstraction. Given a model as in Eq. (1), a partition $(\mathcal{P}_i)_{i=1}^L$ of the state space as defined by Def. 4, and a confidence level $\beta \in (0,1)$ as inputs, we create an abstract iMDP by applying the techniques presented in Sects. 3 and 4.

2) Compute robust optimal policy. We compute a robust optimal policy $\pi^\star$ for the iMDP using Eq. (3). Recall that the problem is to find a controller together with a lower bound λ on the reach-avoid probability. If this condition holds, we output the policy and proceed to step 3; otherwise, we attempt to improve the abstraction in one of the following ways.

First, we can refine the partition at the cost of a larger iMDP, as shown in Sect. 6. Second, using more samples N yields an improved iMDP through tighter intervals (see, e.g., Badings et al. (2022a) for such trade-offs). Finally, the uncertainty in $\alpha \in \Gamma$ may be too large, meaning we need to reduce set Γ using learning techniques (see the related work).

3) Online control. The stored policy is a time-varying map from iMDP states to actions. Recall that an action $a_\ell \in Act$ corresponds to applying a control u_k such that the nominal state $\hat{x}_{k+1} \in \mathcal{T}_\ell$. In view of Eq. (4) and Def. 5, such a $u_k \in \mathcal{U}$ for the original model *exists by construction* and is obtained as the solution to the following convex optimization program:

$$\begin{aligned} c_\ell(x_k) = \operatorname*{arg\,min}_{u \in \mathcal{U}} \quad & \|A(\hat{\alpha})x_k + B(\hat{\alpha})u - \tilde{t}_\ell\|_2 \\ \text{subject to} \quad & A(\hat{\alpha})x_k + B(\hat{\alpha})u \in \mathcal{T}_\ell, \end{aligned} \tag{22}$$

with $\tilde{t}_\ell \in \mathcal{T}_\ell$ a representative point, which indicates a point to which we want to steer the nominal state (in practice, we choose $\tilde{t}_\ell$ as the center of $\mathcal{T}_\ell$, but the theory holds for any such point). Thus, upon observing the current continuous state x_k, we determine the optimal action a_ℓ in the corresponding iMDP state and apply the control input $u_k = c_\ell(x_k) \in \mathcal{U}$.

Correctness of the Abstraction

We lift the confidence probabilities on individual transitions obtained from Theorem 1 to a correctness guarantee on the whole iMDP. The following theorem states this key result:

Theorem 2 (Correctness of the iMDP)**.** *Generate an iMDP abstraction $\mathcal{M}_{\mathbb{I}}$ and compute the robust reach-avoid probability $\underline{Pr}^{\pi^\star}(\mathcal{M}_{\mathbb{I}} \models \varphi_{s_I}^K)$ under optimal policy $\pi^\star$. Under the controller c defined by Eq. (22) for each $k \leq K$, it holds that*

$$\mathbb{P}\left\{ V(x_0, \alpha, c) \geq \underline{Pr}^{\pi^\star}(\mathcal{M}_{\mathbb{I}} \models \varphi_{s_I}^K) \right\} \geq 1 - 2\beta LM. \quad (23)$$

The proof of Theorem 2, which we provide in Badings et al. (2022b, Appendix A), uses the union bound with the fact that the iMDP has at most LM unique probability intervals. By tuning $\beta \in (0,1)$, we thus obtain an abstraction that is correct with a user-specified confidence level of $1 - \tilde{\beta} = 1 - 2\beta LM$.

Crucially, we note that Theorem 2 is, with a probability of at least $1 - 2\beta LM$, a solution to the formal problem stated in Sect. 2 with threshold $\lambda = \underline{Pr}^{\pi^\star}(\mathcal{M}_{\mathbb{I}} \models \varphi_{s_I}^K)$.

Sample complexity. The required sample size N depends logarithmically on the confidence level, cf. Lemmas 1 and 2. Moreover, the number of unique intervals is often lower than the worst-case of LM, so the bound in Theorem 2 can be conservative. In particular, the number of intervals only depends on the state and action definitions of the iMDP, and is thus observable before we apply Theorem 1. To compute less conservative probability intervals, we replace LM in Eq. (23) with the observed number of unique intervals.

Uncertain Additive Disturbance

We extend the generality of our models with an additive parameter representing an external disturbance $q_k \in \mathcal{Q}$ that, in the spirit of this paper, belongs to a convex set $\mathcal{Q} \subset \mathbb{R}^n$ (Blanchini and Miani 2008). The resulting model is

$$x_{k+1} = A(\alpha)x_k + B(\alpha)u_k + q_k + \eta_k. \quad (24)$$

This additional parameter q_k models uncertain disturbances *that are not stochastic* (and can thus not be captured by η_k), and that are independent of the state x_k and control u_k (and can thus not be captured in $A(\alpha)$ or $B(\alpha)$). To account for parameter q_k (which creates another source of epistemic uncertainty), we expand Eq. (6) and Lemma 4 of Badings et al. (2022b, Appendix A), *to be robust against any* $q_k \in \mathcal{Q}$. While this extension increases the size of the sets $\mathcal{H}_{i\ell}^{(\iota)}, \iota = 1, \dots, N$, the procedure outlined in Sect. 4 to compute probability intervals remains the same.

Generality of the model. The parameter q_k expands the applicability of our approach significantly. Consider, e.g., a building temperature control problem, where only the temperatures of adjacent rooms affect each other. We can decompose the model dynamics into the individual rooms by capturing any possible influence between rooms into $q_k \in \mathcal{Q}$, as is common in assume-guarantee reasoning (Bobaru, Pasareanu, and Giannakopoulou 2008). We apply this extension to a large building temperature control problem in Sect. 6.

6 Numerical Experiments

We perform experiments to answer the question: "*Can our method synthesize controllers that are robust against epistemic uncertainty in parameters?*" In this section, we focus

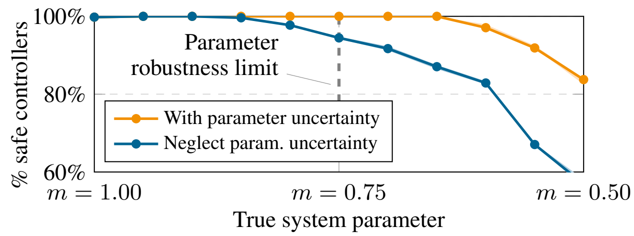

Figure 5: Percentage of initial states with safe performance guarantees (i.e., the simulated reach-avoid probability is above the optimum of Eq. (3) on the iMDP). Our approach that accounts for epistemic uncertainty is *100% safe up to the parameter robustness limit*; neglecting uncertainty is not.

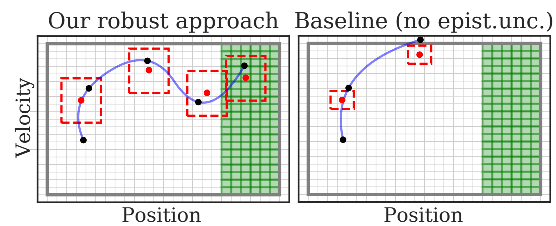

Figure 6: With our approach, the system safely reaches the goal (in green), while the baseline neglecting epistemic uncertainty leaves the safe set (gray box), as it underestimates the successor state sets $\mathcal{H}_{i\ell}$ defined in Eq. (11) (red boxes).

on problems from motion planning and temperature control, and we discuss an additional experiment on a variant of the automated anesthesia delivery benchmark from Abate et al. (2018) in the extended version of this paper (Badings et al. 2022b, Appendix C). All experiments ran single-threaded on a computer with 32 3.7GHz cores and 64GB RAM. A Python implementation of our approach is available at https://github.com/LAVA-LAB/DynAbs, using the probabilistic model checker PRISM (Kwiatkowska, Norman, and Parker 2011) to compute optimal iMDP policies.

Longitudinal Drone Dynamics

We revisit Example 1 of a drone with an uncertain mass $m \in [0.75, 1.25]$. We fix the nominal value of the mass as $\hat{m} = 1$. To purely show the effect of epistemic uncertainty, we set the covariance of the aleatoric uncertainty in η_k (being a Gaussian distribution) to almost zero. The specification is to reach a position of $p_k \geq 8$ before time $K = 12$, while avoiding speeds of $|v_k| \geq 10$. Thus, the safe set is $\mathcal{Z} = [-\infty, \infty] \times [-10, 10]$, of which we create a partition covering $\mathcal{X} = [-10, 14] \times [-10, 10]$ and into 24×20 regions. We use 20K samples of the noise to estimate probability intervals. We compare against a baseline that builds an iMDP for the nominal model only, thus neglecting parameter uncertainty.

Neglecting epistemic uncertainty is unsafe. We solve the formal problem stated in Sect. 2 for every initial state $x_0 \in \mathcal{X}$, resulting in a threshold λ for each of those states. The run time for solving this benchmark is around 3 s. For each x_0, we say that the controller c at a parameter value $\alpha \in \Gamma$ is *unsafe*,

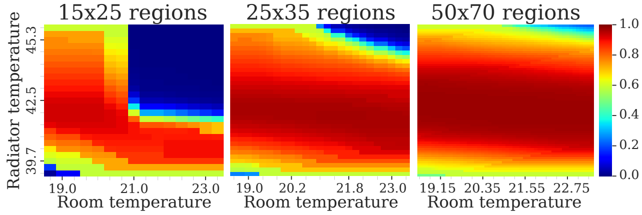

Figure 7: Specification satisfaction probabilities on the iMDP for a single room of the temperature control problem.

if the reach-avoid probability $V(x_0, \alpha, c)$ (estimated using Monte Carlo simulations) is below $\lambda = \underline{Pr}^{\pi^\star}(\mathcal{M}_{\mathbb{I}} \models \varphi_{s_I}^K)$ on the iMDP, as per Eq. (3). In Fig. 5, we show the deviation of the actual mass m from its nominal value, versus the average percentage of states with a safe controller (over 10 repetitions). The *parameter robustness limit* represents the extreme values of the parameter against which our approach is guaranteed to be robust ($m = 0.75$ and 1.25 in this case).

Our approach yields *100% safe controllers* for deviations well over the robustness limit, while the baseline yields *6% unsafe controllers* at this limit. We show simulated trajectories under an actual mass $m = 0.75$ in Fig. 6. These trajectories confirm that our approach safely reaches the goal region while the baseline does not, as it neglects epistemic uncertainty, which is equivalent to assuming $\Delta_i = 0$ in Eq. (11).

Multiple uncertain parameters. To show that our contributions hold independently of the uncertain parameter, we consider a case in which, in addition to the uncertain mass, we have an uncertain friction coefficient. The results of this experiment, presented in Badings et al. (2022b, Appendix C), show that we obtain controllers with similar correctness guarantees, irrespective of the number of uncertain parameters.

Building Temperature Control

We consider a temperature control problem for a 5-room building with Gaussian process noise, each with a dedicated radiator that has an uncertain power output of $\pm 10\%$ around its nominal value (see Badings et al. 2022b, Appendix C for details). The 10D state of this model captures the temperatures of 5 rooms and 5 radiators. The goal is to maintain a temperature within $21 \pm 2.5\,°\mathrm{C}$ for 15 steps of $20\,\mathrm{min}$.

Interactions between rooms as nondeterminism. Since a direct partitioning of the 10D state space is infeasible, we use the procedure from Sect. 5 to capture *any possible thermodynamic interaction* between rooms in the uncertain parameter $q_k \in \mathcal{Q}$. In particular, we show in Badings et al. (2022b, Appendix C) that the set $\mathcal{Q}_i$ for room $i \in \{1, \dots, 5\}$ is characterized by the maximal temperature difference between room i and all adjacent rooms. For the partition used in this particular reach-avoid problem, this maximal temperature difference is $23.5 - 18.5 = 5\,°\mathrm{C}$. Following this procedure, we can easily derive a set-bounded representation of $\mathcal{Q}$, allowing us to decouple the dynamics into the individual rooms.

Refining partitions improves results. We apply our method with an increasingly more fine-grained state-space partition. In Fig. 7, we present, for three different partitions, the thresholds λ of satisfying the specification under the robust optimal iMDP policy as per Eq. (3), from any initial state $x_0 \in \mathcal{X}$. These results confirm the idea from Sect. 5 that partition refinement can lead to controllers with better performance guarantees. A more fine-grained partition leads to more actions enabled in the abstraction, which in turn improves the robust lower bound on the reach-avoid probability.

Scalability. We report run times and model sizes in Badings et al. (2022b, Appendix C, Table 1). The run times vary between $5.6\,\mathrm{s}$ and $8\,\mathrm{min}$ for the smallest (15×25) and largest (70×100) partitions, respectively. Without the decoupling procedure, even a 2-room version on the smallest partition *leads to a memory error*. By contrast, our approach with decoupling procedure has *linear complexity in the number of rooms*. We observe that accounting for epistemic uncertainty yields iMDPs with more transitions and slightly higher run times (for the largest partition: 82 instead of 52 million transitions and 8 instead of $5\,\mathrm{min}$). This is due to larger successor state sets $\mathcal{H}_{i\ell}^{(\iota)}$ in Eq. (11) caused by the epistemic error Δ_i.

7 Conclusions and Future Work

We have presented a novel abstraction-based controller synthesis method for dynamical models with aleatoric and epistemic uncertainty. The method captures those different types of uncertainties in order to ensure certifiably safe controllers. Our experiments show that we can synthesize controllers that are robust against uncertainty and, in particular, against deviations in the model parameters.

Generality of our approach. We stress that the models in Eqs. (1) and (24) can capture many common sources of uncertainty. As we have shown, our approach simultaneously deals with epistemic uncertainty over one or multiple parameters, as well as aleatoric uncertainty due to stochastic noise of an unknown distribution. Moreover, the additive parameter q_k introduced in Sect. 5 enables us to generate abstractions that faithfully capture any error term represented by a bounded set (as we have done for the temperature control problem). For example, we plan to apply our method to *nonlinear systems*, such as non-holonomic robots (Thrun, Burgard, and Fox 2005). Concretely, we may apply our abstraction method on a linearized version of the system while treating linearization errors as nondeterministic disturbances $q_k \in \mathcal{Q}$ in Eq. (24). The main challenge is then to obtain this set-bounded representation $\mathcal{Q}$ of the linearization error.

Scalability. Enforcing robustness against epistemic uncertainty hampers the scalability of our approach, especially compared to similar non-robust abstraction methods, such as Badings et al. 2022a. To reduce the computational complexity, we restrict partitions to be rectangular, but the theory is valid for any convex partition. A finer partition yields larger iMDPs but also improves the guarantees on controllers.

Safe learning. Finally, we wish to integrate the abstractions in a *safe learning framework* (Brunke et al. 2022) by, as discussed in the related work in Sect. 1, applying our approach to guarantee safe interactions with the system.

Acknowledgments

This research has been partially funded by NWO grant NWA.1160.18.238 (PrimaVera), by EPSRC IAA Award EP/X525777/1, and by the ERC Starting Grant 101077178 (DEUCE).

References

Abate, A.; Blom, H. A. P.; Cauchi, N.; Haesaert, S.; Hartmanns, A.; Lesser, K.; Oishi, M.; Sivaramakrishnan, V.; Soudjani, S.; Vasile, C. I.; and Vinod, A. P. 2018. ARCH-COMP18 Category Report: Stochastic Modelling. In *ARCH@ADHS*, volume 54 of *EPiC Series in Computing*, 71–103. EasyChair.

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

Alur, R.; Henzinger, T. A.; Lafferriere, G.; and Pappas, G. J. 2000. Discrete abstractions of hybrid systems. *Proc. IEEE*, 88(7): 971–984.

Badings, T. S.; Abate, A.; Jansen, N.; Parker, D.; Poonawala, H. A.; and Stoelinga, M. 2022a. Sampling-Based Robust Control of Autonomous Systems with Non-Gaussian Noise. In *AAAI*, 9669–9678. AAAI Press.

Badings, T. S.; Jansen, N.; Poonawala, H. A.; and Stoelinga, M. 2021. Filter-Based Abstractions with Correctness Guarantees for Planning under Uncertainty. *CoRR*, abs/2103.02398.

Badings, T. S.; Romao, L.; Abate, A.; and Jansen, N. 2022b. Probabilities Are Not Enough: Formal Controller Synthesis for Stochastic Dynamical Models with Epistemic Uncertainty. Technical report, CoRR, abs/2210.05989.

Baier, C.; and Katoen, J. 2008. *Principles of model checking*. MIT Press.

Ben-Tal, A.; Ghaoui, L. E.; and Nemirovski, A. 2009. *Robust Optimization*, volume 28 of *Princeton Series in Applied Mathematics*. Princeton University Press.

Berkenkamp, F.; Turchetta, M.; Schoellig, A. P.; and Krause, A. 2017. Safe Model-based Reinforcement Learning with Stability Guarantees. In *NIPS*, 908–918.

Blanchini, F.; and Miani, S. 2008. *Set-theoretic methods in control*, volume 78. Springer.

Bobaru, M. G.; Pasareanu, C. S.; and Giannakopoulou, D. 2008. Automated Assume-Guarantee Reasoning by Abstraction Refinement. In *CAV*, volume 5123 of *Lecture Notes in Computer Science*, 135–148. Springer.

Boucheron, S.; Lugosi, G.; and Massart, P. 2013. *Concentration Inequalities - A Nonasymptotic Theory of Independence*. Oxford University Press.

Brunke, L.; Greeff, M.; Hall, A. W.; Yuan, Z.; Zhou, S.; Panerati, J.; and Schoellig, A. P. 2022. Safe Learning in Robotics: From Learning-Based Control to Safe Reinforcement Learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 5(1): 411–444.

Campi, M. C.; Carè, A.; and Garatti, S. 2021. The scenario approach: A tool at the service of data-driven decision making. *Annu. Rev. Control.*, 52: 1–17.

Campi, M. C.; and Garatti, S. 2011. A Sampling-and-Discarding Approach to Chance-Constrained Optimization: Feasibility and Optimality. *J. Optim. Theory Appl.*, 148(2): 257–280.

Casella, G.; and Berger, R. L. 2021. *Statistical inference*. Cengage Learning.

Cauchi, N.; and Abate, A. 2019. StocHy: Automated Verification and Synthesis of Stochastic Processes. In *TACAS (2)*, volume 11428 of *Lecture Notes in Computer Science*, 247–264. Springer.

Cauchi, N.; Laurenti, L.; Lahijanian, M.; Abate, A.; Kwiatkowska, M.; and Cardelli, L. 2019. Efficiency through uncertainty: scalable formal synthesis for stochastic hybrid systems. In *HSCC*, 240–251. ACM.

Charpentier, B.; Senanayake, R.; Kochenderfer, M. J.; and Günnemann, S. 2022. Disentangling Epistemic and Aleatoric Uncertainty in Reinforcement Learning. *CoRR*, abs/2206.01558.

Chou, G.; Ozay, N.; and Berenson, D. 2021. Model Error Propagation via Learned Contraction Metrics for Safe Feedback Motion Planning of Unknown Systems. In *CDC*, 3576–3583. IEEE.

Depeweg, S.; Hernández-Lobato, J. M.; Doshi-Velez, F.; and Udluft, S. 2018. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-sensitive Learning. In *ICML*, volume 80 of *Proceedings of Machine Learning Research*, 1192–1201. PMLR.

Fan, C.; Qin, Z.; Mathur, U.; Ning, Q.; Mitra, S.; and Viswanathan, M. 2022. Controller Synthesis for Linear System With Reach-Avoid Specifications. *IEEE Trans. Autom. Control.*, 67(4): 1713–1727.

Fisac, J. F.; Chen, M.; Tomlin, C. J.; and Sastry, S. S. 2015. Reach-avoid problems with time-varying dynamics, targets and constraints. In *HSCC*, 11–20. ACM.

Fox, C. R.; and Ülkümen, G. 2011. Distinguishing two dimensions of uncertainty. In *Perspectives on Thinking, Judging, and Decision Making*. Universitetsforlaget.

García, J.; and Fernández, F. 2015. A comprehensive survey on safe reinforcement learning. *J. Mach. Learn. Res.*, 16: 1437–1480.

Geibel, P.; and Wysotzki, F. 2005. Risk-Sensitive Reinforcement Learning Applied to Control under Constraints. *J. Artif. Intell. Res.*, 24: 81–108.

Geromel, J. C.; and Colaneri, P. 2006. Robust stability of time varying polytopic systems. *Systems & Control Letters*, 55(1): 81–85.

Givan, R.; Leach, S. M.; and Dean, T. L. 2000. Bounded-parameter Markov decision processes. *Artif. Intell.*, 122(1-2): 71–109.

Haesaert, S.; den Hof, P. M. J. V.; and Abate, A. 2017. Data-driven and model-based verification via Bayesian identification and reachability analysis. *Autom.*, 79: 115–126.

Haesaert, S.; Nilsson, P.; Vasile, C. I.; Thakker, R.; Agha-mohammadi, A.; Ames, A. D.; and Murray, R. M. 2018. Temporal Logic Control of POMDPs via Label-based Stochastic Simulation Relations. In *ADHS*, volume 51 of *IFAC-PapersOnLine*, 271–276. Elsevier.

Hüllermeier, E.; and Waegeman, W. 2021. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Mach. Learn.*, 110(3): 457–506.

Jackson, J.; Laurenti, L.; Frew, E. W.; and Lahijanian, M. 2020. Safety Verification of Unknown Dynamical Systems via Gaussian Process Regression. In *CDC*, 860–866. IEEE.

Kenanian, J.; Balkan, A.; Jungers, R. M.; and Tabuada, P. 2019. Data driven stability analysis of black-box switched linear systems. *Autom.*, 109.

Knuth, C.; Chou, G.; Ozay, N.; and Berenson, D. 2021. Planning With Learned Dynamics: Probabilistic Guarantees on Safety and Reachability via Lipschitz Constants. *IEEE Robotics Autom. Lett.*, 6(3): 5129–5136.

Kumar, P. R.; and Varaiya, P. 2015. *Stochastic systems: Estimation, identification, and adaptive control*. SIAM.

Kwiatkowska, M. Z.; Norman, G.; and Parker, D. 2011. PRISM 4.0: Verification of Probabilistic Real-Time Systems. In *CAV*, volume 6806 of *Lecture Notes in Computer Science*, 585–591. Springer.

Lahijanian, M.; Andersson, S. B.; and Belta, C. 2015. Formal Verification and Synthesis for Discrete-Time Stochastic Systems. *IEEE Trans. Autom. Control.*, 60(8): 2031–2045.

Lavaei, A.; Soudjani, S.; Abate, A.; and Zamani, M. 2022. Automated verification and synthesis of stochastic hybrid systems: A survey. *Automatica*, 146: 110617.

Lavaei, A.; Soudjani, S.; Frazzoli, E.; and Zamani, M. 2023. Constructing MDP Abstractions Using Data With Formal Guarantees. *IEEE Control. Syst. Lett.*, 7: 460–465.

Loquercio, A.; Segù, M.; and Scaramuzza, D. 2020. A General Framework for Uncertainty Estimation in Deep Learning. *IEEE Robotics Autom. Lett.*, 5(2): 3153–3160.

Majumdar, A.; and Tedrake, R. 2017. Funnel libraries for real-time robust feedback motion planning. *Int. J. Robotics Res.*, 36(8): 947–982.

Makdesi, A.; Girard, A.; and Fribourg, L. 2021. Data-Driven Abstraction of Monotone Systems. In *L4DC*, volume 144 of *Proceedings of Machine Learning Research*, 803–814. PMLR.

Modares, H. 2022. Data-driven Safe Control of Linear Systems Under Epistemic and Aleatory Uncertainties. *CoRR*, abs/2202.04495.

Pereira, M.; Wang, Z.; Exarchos, I.; and Theodorou, E. A. 2020. Safe Optimal Control Using Stochastic Barrier Functions and Deep Forward-Backward SDEs. In *CoRL*, volume 155 of *Proceedings of Machine Learning Research*, 1783–1801. PMLR.

Peruffo, A.; and Mazo, M. 2023. Data-Driven Abstractions With Probabilistic Guarantees for Linear PETC Systems. *IEEE Control. Syst. Lett.*, 7: 115–120.

Puggelli, A.; Li, W.; Sangiovanni-Vincentelli, A. L.; and Seshia, S. A. 2013. Polynomial-Time Verification of PCTL Properties of MDPs with Convex Uncertainties. In *CAV*, volume 8044 of *Lecture Notes in Computer Science*, 527–542. Springer.

Romao, L.; Papachristodoulou, A.; and Margellos, K. 2022. On the exact feasibility of convex scenario programs with discarded constraints. *IEEE Transactions on Automatic Control (to appear)*.

Salamati, A.; and Zamani, M. 2022. Safety Verification of Stochastic Systems: A Repetitive Scenario Approach. *IEEE Control Systems Letters*.

Shmarov, F.; and Zuliani, P. 2015. ProbReach: verified probabilistic delta-reachability for stochastic hybrid systems. In *HSCC*, 134–139. ACM.

Smith, R. C. 2013. *Uncertainty quantification: theory, implementation, and applications*, volume 12. Siam.

Sullivan, T. J. 2015. *Introduction to uncertainty quantification*, volume 63. Springer.

Thiebes, S.; Lins, S.; and Sunyaev, A. 2021. Trustworthy artificial intelligence. *Electron. Mark.*, 31(2): 447–464.

Thrun, S.; Burgard, W.; and Fox, D. 2005. *Probabilistic robotics*. Intelligent robotics and autonomous agents. MIT Press.

Tsiamis, A.; and Pappas, G. J. 2019. Finite Sample Analysis of Stochastic System Identification. In *CDC*, 3648–3654. IEEE.

Vinod, A. P.; Gleason, J. D.; and Oishi, M. M. K. 2019. SReachTools: a MATLAB stochastic reachability toolbox. In *HSCC*, 33–38. ACM.

Vinod, A. P.; Israel, A.; and Topcu, U. 2022. On-the-Fly Control of Unknown Nonlinear Systems With Sublinear Regret. *IEEE Transactions on Automatic Control*, 1–13.

Wiesemann, W.; Kuhn, D.; and Sim, M. 2014. Distributionally Robust Convex Optimization. *Oper. Res.*, 62(6): 1358–1376.

Wolff, E. M.; Topcu, U.; and Murray, R. M. 2012. Robust control of uncertain Markov Decision Processes with temporal logic specifications. In *CDC*, 3372–3379. IEEE.

Yedavalli, R. K. 2014. Robust control of uncertain dynamic systems. *AMC*, 10: 12.

Zanon, M.; and Gros, S. 2021. Safe Reinforcement Learning Using Robust MPC. *IEEE Trans. Autom. Control.*, 66(8): 3638–3652.

Zikelic, D.; Lechner, M.; Henzinger, T. A.; and Chatterjee, K. 2022. Learning Control Policies for Stochastic Systems with Reach-avoid Guarantees. *CoRR*, abs/2210.05308.