International Journal on Software Tools for Technology Transfer (2022) 24:803–819
https://doi.org/10.1007/s10009-022-00673-z

GENERAL

Special Issue: TACAS 2020

Scenario-based verification of uncertain parametric MDPs

Thom Badings[1] · Murat Cubuktepe[2] · Nils Jansen[1] · Sebastian Junges[1] · Joost-Pieter Katoen[3] · Ufuk Topcu[2]

Accepted: 29 August 2022 / Published online: 14 September 2022
© The Author(s) 2022

Abstract
We consider parametric Markov decision processes (pMDPs) that are augmented with unknown probability distributions over parameter values. The problem is to compute the probability to satisfy a temporal logic specification with any concrete MDP that corresponds to a sample from these distributions. As solving this problem precisely is infeasible, we resort to sampling techniques that exploit the so-called scenario approach. Based on a finite number of samples of the parameters, the proposed method yields high-confidence bounds on the probability of satisfying the specification. The number of samples required to obtain a high confidence on these bounds is independent of the number of states and the number of random parameters. Experiments on a large set of benchmarks show that several thousand samples suffice to obtain tight and high-confidence lower and upper bounds on the satisfaction probability.

Keywords Markov decision processes · Uncertainty · Verification · Scenario optimization

1 Introduction

MDPs Markov decision processes (MDPs) model sequential decision-making problems in stochastic dynamic environments [48]. They are widely used in areas such as planning [50], reinforcement learning [54], formal verification [7], and robotics [40]. Mature model checking tools such as PRISM [37] and Storm [25] employ efficient algorithms to verify the correctness of MDPs against temporal logic specifications [45], provided all transition probabilities and cost functions are exactly known. In many applications, however, this assumption may be unrealistic, as certain system parameters are typically not exactly known and under control by external sources.

Uncertainty on MDPs A common approach to deal with unknown system parameters is to describe transition probabilities of an MDP using intervals [26,29,47] or generalizations to a class of uncertain MDPs [42,55,57]. Solution approaches rely on the limiting assumption that the uncertainty at different states of the MDP is independent from each other. As an example, consider a simple motion planning scenario where an unmanned aerial vehicle (UAV) is tasked to transport a certain payload to a target location. The problem is to compute a *strategy* (or *policy*) for the UAV to successfully deliver the payload while taking into account the weather conditions. External factors such as the wind strength or direction may affect the movement of the UAV. The assumption that such weather conditions are independent between the different possible states of UAV is unrealistic and may yield pessimistic verification results.

Illustrative examples We stress that the same situation appears in various systems. For example, in the verification of network protocols, we typically do not precisely know the channel quality (i.e., the loss rate). However, the loss rate is independent of the question of whether we are, e.g., probing or actually sending useful data over the network. A typical verification task would be to show that the protocol yields a sufficiently high quality of service. A verification approach that pessimistically assumes that the channel quality depends on the protocol state may be too pessimistic and fail to establish that the protocol provides the required quality of service.

This work was partially funded by NWO grant NWA.1160.18.238 (PrimaVera), the ERC AdG 787914 (FRAPPANT), NSF 1652113, ONR N000141613165, NASA NNX17AD04G and AFRL FA8650-15-C-2546.

✉ Thom Badings
thom.badings@ru.nl

1 Radboud University, Nijmegen, The Netherlands

2 The University of Texas at Austin, Austin, TX, USA

3 RWTH Aachen University, Aachen, Germany

Parametric models Parametric Markov models allow to explicitly describe that some probabilities are unknown but explicitly related [23,30,36]. In a parametric MDP, one uses variables (*parameters*) that encode, e.g., the probability of wind gusts affecting a UAV, or the probability of packet loss in a network. Transition probabilities are then given as expressions over these parameters. Substituting the parameters with concrete values yields an *induced* (standard) MDP. A variety of parameter synthesis methods have been devised, see the related work in Sect. 8 for details. A typical verification query concerns feasibility, that is, *whether there exist parameter values such that the induced model satisfies a specification*, which implicitly assumes that the parameters are controllable. Another query is to ask *whether for all parameter values the induced model satisfies a specification*. The latter can lead to pessimistic verification results: a UAV may be able to fly during most weather conditions, but it may be impossible to find a satisfying strategy for flying during a rare storm.

Uncertain parametric models Rather than asking about the *existence* of parameter values, we want to analyze a system by considering the *typical* parameter values. In terms of our examples, this means that we want to investigate the typical weather conditions and the typical channel qualities. Similar to [51], we, therefore, assume that the parameters are random variables. For instance, weather data in the form of probability distributions may provide additional information on potential changes during the mission, or a measurement series may provide typical channel qualities. For weather data, such probability distributions may be derived from historical data of, for example, the wind speed [43].

Problem statement We study a setting where the uncertain parameters are random variables that are defined on an arbitrary (joint) probability space over all parameters. We assume that we can sample *independent and identically distributed* parameter values from this distribution and solve the following problem.

Problem statement. Given a parametric MDP and a distribution over the parameter values, compute the probability with which any randomly drawn parameter values yield an induced MDP that satisfies a given specification.

We call this probability the *satisfaction probability*. The intuition is that the question of whether all (or some) parameter values satisfy a specification—as is often done in parameter synthesis [36]—is replaced by the question of *how much we expect the (sampled) model to satisfy a specification*. For example, a satisfaction probability of 80% tells that, if we randomly sample the parameters, with a probability of 80% there exists a strategy for the resulting MDP satisfying the specification. Importantly, we thus assume that the parameter values are *observable*, and hence known when synthesizing a strategy. In every concrete MDP, we may use a different strategy. This is in contrast to a robust strategy synthesis approach, where a single strategy is sought that is robust against all (or a portion of the) parameter valuations.

Scenario-based verification In this paper, we devise a method that answers the problem statement up to a user-specified confidence level. That is, we aim to solve the problem statement up to a statistical guarantee. To achieve this, we resort to *sampling-based* algorithms that yield a confidence (probability) on the bounds of the satisfaction probability. In doing so, *we do not make any assumptions* about the distribution over the parameter values. Referring back to the UAV example, we want to compute a confidence bound on the probability for the UAV to successfully finish its mission for some strategy. To derive confidence bounds, we first formulate the problem of (exactly) computing the satisfaction probability as a *chance-constrained optimization program*. However, this problem is very hard to solve [16], especially because we do not assume any knowledge on the probability distribution of the parameters. We, therefore, use a technique known as *scenario optimization* (also called the *scenario approach*), which provides guarantees on the satisfaction probability via sampling techniques [13,15]. The basic idea is to consider a finite set of samples from the distribution over the parameters and restrict the problem to these samples only. This so-called *scenario optimization problem* can be solved efficiently [11]. The solution to the scenario program is, *with a high confidence probability*, a solution to the previously mentioned chance-constrained program.

Our approach For our setting, we first sample a finite number of parameter instantiations, each of which induces a concrete MDP. We can check the satisfaction of the specification for these concrete MDPs efficiently using, e.g. , a probabilistic model checker. Based on the results, we compute an estimate of the satisfaction probability, which is a lower bound on the true satisfaction probability with the desired confidence probability. For example, we may obtain a lower bound on the satisfaction probability of 80%, which holds with a confidence probability of at least 90%. We show that the probability of an incorrect lower bound on the satisfaction probability diminishes to zero *exponentially rapidly* with an increasing sample size. Moreover, the number of required samples depends on neither the size of the state space nor the number of random parameters. Finally, we show that we can use the same technique to additionally compute *upper bounds* on the satisfaction probability.

Empirical evaluation In our experiments, we validate the theoretical results using several MDPs that have different sizes of state and parameter spaces. We demonstrate experimentally that the required number of parameter samples is indeed not sensitive to the dimension of the state and parameter space. In addition, we show the effectiveness of our

method with a new dedicated case study based on the aforementioned UAV example which incorporates 900 random parameters.

Contributions This paper revises an earlier conference paper [19] as follows. Due to new results in [28] that lift some assumptions required for the scenario approach, we can simplify and generalize our approach by a simplified chance-constrained program. This program can be used in conjunction with all (standard temporal) properties on parametric MDPs. This change in the approach yields completely revised technical sections of the paper. Furthermore, this paper fixes a technical error in [19]. The (new) bounds in Theorem 1 of this paper are less pessimistic. The (new) bounds in Theorem 2 are now correct at the cost of being slightly more conservative.

2 Motivating example

We consider the previously mentioned UAV motion planning example in more detail, where the objective is to transport a payload from one end of a valley to the other. A specification for the UAV would be that it (with at least probability x) realizes this objective. The typical approach to verify the UAV against this specification is to create a model that faithfully captures its dynamics.

However, the dynamics of the UAV depend on the weather conditions, which may be different on each day. Thus, each weather condition induces a distinct model for the UAV. In line with the problem statement, we assume that the weather conditions are deterministically observed on the day itself, and we can adapt the strategy accordingly. When designing the UAV, we may require that the expected number of days per year on which the UAV can satisfy a mission objective is sufficiently high. Concretely, this translates to the requirement that the UAV shall satisfy the specification above on, e.g., at least 90% of the days. More abstractly, this requirement implies we want to show that the probability of a random day yielding weather conditions on which a specification of the corresponding model is satisfied is at least 90%. To this end, we assume that we have historical data that describe a distribution over weather conditions.

Model construction Planning scenarios like the UAV example are naturally modeled by MDPs, where the actions determine the high-level control actions and the action outcomes are stochastic due to the influences of the environment. While this planning problem is, to some degree, continuous, high-level planners often discretize the world. We thus obtain the following grid world in which the UAV can decide to fly in either of the six cardinal directions (N, W, S, E, up, down). States encode the position of the UAV, the current weather condition (sunny, stormy), and the general wind direction in the valley. In this particular scenario, we assume that the

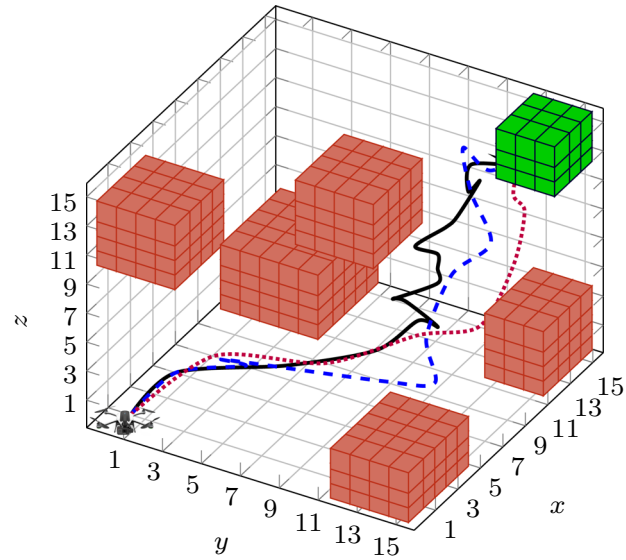


Fig. 1 An example of a 3D UAV benchmark with obstacles (red boxes) and a target area (green box)

probabilistic outcomes are (only) determined by the wind in the valley and the control action. Concretely, we assume that an action moves the UAV one cell in the corresponding direction. Moreover, the wind moves the UAV one cell in the wind direction with a probability p, which depends on the wind speed. Furthermore, we assume that the weather and wind-conditions change during the day and are described by a stochastic process.

We observe that some probabilities in the system are not fixed but rather a function of the weather. Thus, the model is an uncertain MDP (uMDP) whose transition probabilities depend on the weather. Concretely, parameters describe how the weather affects the UAV in different zones of the valley, and how the weather/wind may change during the day. Historical weather data now induce a distribution over the (joint) parameters. Sampling from this distribution yields a concrete instantiated MDP. The problem is to compute the satisfaction probability, i.e., the probability that *for any sampled MDP, we are able to synthesize a UAV strategy that satisfies the specification.* Figure 1 shows an example environment for the UAV, with the target zone in green and zones to avoid shown in red. The shown trajectories are typical paths under three different weather conditions (we refer to the experiments in Sect. 7 for details).

3 Preliminaries

In the following, we use probability distributions over finite and infinite sets, for which we refer to [9] for details. Let $V = \{x_1, \ldots, x_n\}$ be a finite set of variables *(parameters)* over $\mathbb{R}^n$. The set of polynomials over V, with rational coefficients, is

denoted by $\mathbb{Q}[V]$. We denote the cardinality of a set $\mathcal{U}$ by $|\mathcal{U}|$.

3.1 Parametric models

We introduce parametric Markov decision processes. Note that we omit reward models, but our methods are directly applicable to reward measures.

Definition 1 (*pMDP*) A parametric Markov decision process (pMDP) $\mathcal{M}$ is a tuple $\mathcal{M} = (S, Act, s_I, V, \mathcal{P})$ with a finite set S of *states*, a finite set Act of *actions*, an *initial state* $s_I \in S$, a finite set V of *parameters*, and a *transition function* $\mathcal{P}\colon S \times Act \times S \to \mathbb{Q}[V]$.

The set $ActS(s)$ of *enabled* actions at state $s \in S$ is $ActS(s) = \{\alpha \in Act \mid \exists s' \in S, \mathcal{P}(s, \alpha, s') \neq 0\}$. Without loss of generality, we require $ActS(s) \neq \emptyset$ for all $s \in S$. If $|ActS(s)| = 1$ for all $s \in S$, $\mathcal{M}$ is a *parametric discrete-time Markov chain (pMC)* and we denote its transition function by $\mathcal{P}(s, s') \in \mathbb{Q}[V]$.

Example 1 Consider the pMC in Fig. 2 with parameter $V = \{v\}$, initial state s_0, and target set $T = \{s_3\}$ (used later). Transitions are annotated with polynomials over the parameter v.

A pMDP $\mathcal{M}$ is a *Markov decision process (MDP)* if the transition function yields *well-defined* probability distributions, that is, $\mathcal{P}\colon S \times Act \times S \to [0, 1]$ and $\sum_{s' \in S} \mathcal{P}(s, \alpha, s') = 1$ for all $s \in S$ and $\alpha \in ActS(s)$. We denote the *parameter space of* $\mathcal{M}$ by $\mathcal{V}_{\mathcal{M}}$, which consists of functions $V \to \mathbb{R}$ that map parameters to concrete values. Applying an *instantiation* $u \in \mathcal{V}_{\mathcal{M}}$ to a pMDP $\mathcal{M}$ yields the *instantiated* MDP $\mathcal{M}[u]$ by replacing each $f \in \mathbb{Q}[V]$ in $\mathcal{M}$ by $f[u]$. An instantiation u is *well defined* for $\mathcal{M}$ if the resulting model $\mathcal{M}[u]$ is an MDP. In the remainder, we assume that all parameter instantiations in $\mathcal{V}_{\mathcal{M}}$ yield well-defined MDPs. We call u *graph-preserving* if for all $s, s' \in S$ and $\alpha \in Act$ it holds that $\mathcal{P}(s, \alpha, s') \neq 0 \Rightarrow \mathcal{P}(s, \alpha, s')[u] \in (0, 1]$.

Assumption 1 We consider only parameter instantiations for upMDPs that are graph-preserving.

To define measures on MDPs, nondeterministic choices are resolved by a *strategy* $\sigma\colon S \to Act$ with $\sigma(s) \in ActS(s)$. The set of all strategies over $\mathcal{M}$ is $Str^{\mathcal{M}}$. For the specifications we consider in this paper, *memoryless deterministic strategies* are sufficient [7]. Applying a strategy σ to an MDP $\mathcal{M}$ yields an *induced* Markov chain (MC) $\mathcal{M}[\sigma]$ where all nondeterminism is resolved.

Measures For an MC $\mathcal{D}$, the *reachability probability* $\mathbf{Pr}_{\mathcal{D}}(\Diamond T)$ describes the (time unbounded) reachability probability of reaching a set $T \subseteq S$ of target states from the initial state s_I [7]. Similar definitions can be given for the step-bounded reachability probability of reaching a set T from the initial state within k steps, and—given rewards for every state—the expected rewards accumulated until reaching the target states or the long-run average, and so forth.

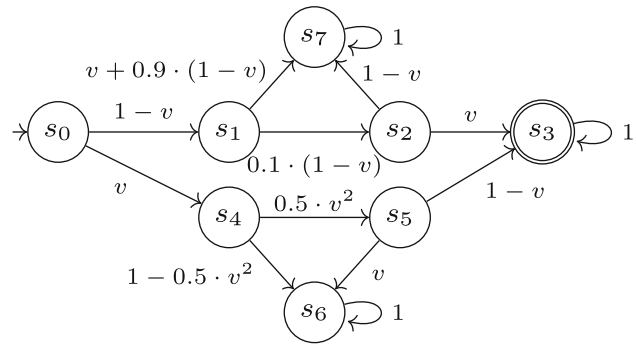


Fig. 2 A pMC/upMC with parameter v

For an MDP $\mathcal{M}$, these measures are typically lifted. The *maximum reachability probability* $\mathbf{Pr}^{\max}_{\mathcal{M}}(\Diamond T)$ is the maximum reachability probability in all induced Markov chains (for all strategies $\sigma \in Str^{\mathcal{M}}$), i.e., $\mathbf{Pr}^{\max}_{\mathcal{M}}(\Diamond T) = \max_{\sigma} \mathbf{Pr}_{\mathcal{M}[\sigma]}(\Diamond T)$. Similar definitions hold for the minimums and the other measures described above. Our approach is directly applicable to more general measures, e.g., measures on paths described by LTL properties [45].

Specifications A specification φ combines a measure, a threshold λ, and a comparison operator from $\{<, \leq, \geq, >\}$. For example, the specification $\varphi = \mathbf{P}_{\leq\lambda}(\Diamond T) = (\mathbf{Pr}^{\max}_{\mathcal{M}}(\Diamond T), \leq \lambda)$ specifies that the maximal reachability probability $\mathbf{Pr}^{\max}_{\mathcal{M}}(\Diamond T)$ is at most λ for the MDP $\mathcal{M}$. If this statement is true for $\mathcal{M}$, we say that $\mathcal{M}$ satisfies the specification, written as $\mathcal{M} \models \varphi$. For an MC $\mathcal{D}$, $\mathbf{P}_{\leq\lambda}(\Diamond T)$ is defined for the measure $\mathbf{Pr}_{\mathcal{D}}(\Diamond T, \leq \lambda)$.

3.2 Solution functions

Recall that every parameter instantiation $u \in \mathcal{V}_{\mathcal{M}}$ for pMDP $\mathcal{M}$ induces a concrete MDP $\mathcal{M}[u]$. For this MDP, we may then compute any of the measures described above. We exploit this relationship to create a direct mapping from parameter instantiations to real values.

Definition 2 (*Solution function*) A solution function $\mathsf{sol}_{\mathcal{M}}\colon \mathcal{V}_{\mathcal{M}} \to \mathbb{R}$ for pMDP $\mathcal{M}$ is a function that maps a parameter instantiation $u \in \mathcal{V}_{\mathcal{M}}$ to a value $\mathsf{sol}_{\mathcal{M}}(u)$, called the *solution of* u.

In particular, we are interested in solution functions that map a parameter instantiation to the solution of computing a particular measure.[1] For instance, we may consider a solution function $\mathsf{sol}_{\mathcal{M}}$ that maps a parameter instantiation u to the

[1] Notice that we later assume the existence of the integral in Def. 4, which excludes some esoteric functions.

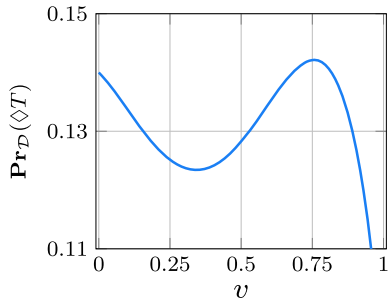


Fig. 3 The solution function for the reachability probability of s_3 for the pMC in Fig. 2

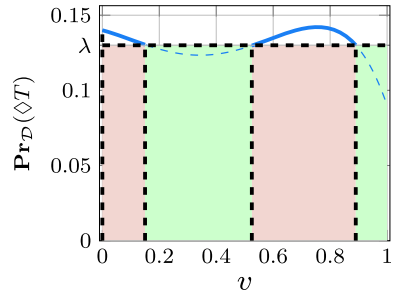


Fig. 4 The probability of satisfying the reachability specification $\varphi = \mathbf{P}_{\leq\lambda}(\Diamond T)$ for the upMC in Fig. 2, versus the value of the parameter v. Intervals that satisfy φ are green, intervals that violate φ are red

probability $\mathbf{Pr}^{\min}_{\mathcal{M}[u]}(\Diamond T)$. In that case, we say that u has solution $\mathbf{Pr}^{\min}_{\mathcal{M}[u]}(\Diamond T)$. Figure 3 depicts a solution function for the reachability probability $\mathbf{Pr}_{\mathcal{M}}(\Diamond T)$ in the pMC from Fig. 2.

Solution functions for parametric models with reachability and expected rewards measures are well studied, in particular their computation [23,24,30], but also some of their properties [6,56]. Already for pMCs, these functions are typically infeasible to compute. In the context of this paper, the important idea is that we can determine the size of the region where this function exceeds a threshold by sampling, as explained next.

3.3 Uncertain parametric MDPs

We now introduce the setting studied in this paper. Specifically, we use pMDPs whose parameters define the uncertainty in the transition probabilities of an MDP. We add another layer of uncertainty, where each parameter follows a probability distribution. For example, referring back to the UAV example in Sect. 2, each weather condition has a certain probability, and every condition leads to a certain parameter instantiation. Importantly, the probability distribution of the parameters is assumed to be *unknown*, and we just assume that we are able to sample this distribution.

Definition 3 (*upMDP*) An *uncertain pMDP* (upMDP) $\mathcal{M}_{\mathbb{P}}$ is a tuple $\mathcal{M}_{\mathbb{P}} = (\mathcal{M}, \mathbb{P})$ with $\mathcal{M}$ a pMDP, and $\mathbb{P}$ a probability distribution over the parameter space $\mathcal{V}_{\mathcal{M}}$. If $\mathcal{M}$ is a pMC, then we call $\mathcal{M}_{\mathbb{P}}$ a upMC.

Intuitively, a upMDP is a pMDP with an associated distribution over possible parameter instantiations. That is, sampling from $\mathcal{V}_{\mathcal{M}}$ according to $\mathbb{P}$ yields concrete MDPs $\mathcal{M}[u]$ with instantiations $u \in \mathcal{V}_{\mathcal{M}}$ (and $\mathbb{P}(u) > 0$).

Definition 4 (*Satisfaction probability*) Let $\mathcal{M}_{\mathbb{P}} = (\mathcal{M}, \mathbb{P})$ be a upMDP and φ a specification. The (weighted) *satisfaction probability of* φ in $\mathcal{M}_{\mathbb{P}}$ is

$$F(\mathcal{M}_{\mathbb{P}}, \varphi) = \int_{\mathcal{V}_{\mathcal{M}}} I_{\varphi}(u)\, d\,\mathbb{P}(u)$$

with $u \in \mathcal{V}_{\mathcal{M}}$ and $I_{\varphi}\colon \mathcal{V}_{\mathcal{M}} \to \{0, 1\}$ is the indicator for φ, i.e., $I_{\varphi}(u) = 1$ iff $\mathcal{M}[u] \models \varphi$.

Note that I_{φ} is measurable for all specifications mentioned in this paper, as it partitions $\mathcal{V}_{\mathcal{M}}$ into a finite union of semi-algebraic sets [8,56]. Moreover, we have that $F(\mathcal{M}_{\mathbb{P}}, \varphi) \in [0, 1]$ and

$$F(\mathcal{M}_{\mathbb{P}}, \varphi) + F(\mathcal{M}_{\mathbb{P}}, \neg\varphi) = 1. \tag{1}$$

Example 2 We expand the pMC in Fig. 2 toward a upMC with a uniform distribution for the parameter v over the interval $[0, 1]$. In Fig. 4, we again plot the solution function for the reachability probability in the pMC from Fig. 2, which was also shown in Fig. 3. Additionally, we compare this probability against a threshold $\lambda = 0.13$ with comparison operator $\leq$, and we plot the satisfying region and its complementary as green and red regions. The satisfying region is given by the union of the intervals $[0.13, 0.525]$ and $[0.89, 1.0]$, and the satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi)$ is $0.395 + 0.11 = 0.505$.

4 Problem statement

Let us now formalize the problem of interest. We aim to compute the satisfaction probability of the parameter space for a specification φ on a upMDP. Equivalently, we thus seek the probability that a randomly sampled instantiation u from the parameter space $\mathcal{V}_{\mathcal{M}}$ induces an MDP $\mathcal{M}[u]$ which satisfies φ. Formally: given a upMDP $\mathcal{M}_{\mathbb{P}} = (\mathcal{M}, \mathbb{P})$, and a specification φ, compute the satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi)$. We *approximate* this probability by sampling the parameters. Such an approach cannot be precise and correct in all cases, because we only have a finite number of samples at our disposal. Instead, we provide the following *probably approximately correct* (PAC) style formulation [33], meaning that we compute a *lower bound* on the satisfaction probability that is correct with *high confidence*:

Formal problem 1. Given a upMDP $\mathcal{M}_{\mathbb{P}} = (\mathcal{M}, \mathbb{P})$, a specification φ, and a confidence probability $\beta \in (0, 1)$, compute a lower bound η on the satisfaction probability, such that $F(\mathcal{M}_{\mathbb{P}}, \varphi) \geq \eta$ holds with a confidence probability of at least β.

Intuitively, given a confidence probability β close below one, we obtain η as a *high-confidence lower bound on the satisfaction probability* $F(\mathcal{M}_{\mathbb{P}}, \varphi)$.

Remark 1 We can also compute an upper bound on the satisfaction probability by exploiting (1) and computing a lower bound for the negated specification $\neg\varphi$.

Furthermore, as is typical in PAC settings, if a specific value for η is desired, we are also able to compute the *confidence that η is indeed a lower bound on the satisfaction probability*. We will exploit both directions, with either given β or η, in the numerical examples in Sect. 7. We illustrate our formal problem by continuing our examples on the upMC in Fig. 2 and the UAV.

Example 3 Let us reconsider the upMC from Example 2 with φ and satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi) = 0.505$. Assume we do not yet know this probability. The problem statement then asks how to compute an η such that with high confidence β, say 0.99, $F(\mathcal{M}_{\mathbb{P}}, \varphi) \geq \eta$.

Example 4 For the UAV motion planning example introduced in Sect. 2, consider the question "*What is a lower bound on the probability that on a given day, there exists a strategy for the UAV to successfully complete the mission?*" Our specification φ for successfully completing a mission could then be that the maximal reachability probability to a target state is above 0.99, or that the expected travel time is below 12 hours. For any such φ, assume that we want to answer the question above with confidence of $\beta = 0.9$. The resulting lower bound on the satisfaction probability could be, e.g., $\eta = 0.81$. This means that with a confidence probability of $\beta = 0.9$, the actual satisfaction probability is indeed at least $\eta = 0.81$. If we change the confidence β to 0.99, the obtained lower bound may reduce to $\eta = 0.78$. Intuitively, the more confidence we want to have, the lower the lower bound.

5 Computing the satisfaction probability

In this section, we introduce our approach for solving the problem presented in Sect. 4. We focus on a practical overview of our approach in this section, while postponing technical details and the derivation of our main results (Theorems 1 and 2) to Sect. 6. First, in Sect. 5.1, we fix some notation for our concrete setup. In particular, we discuss how

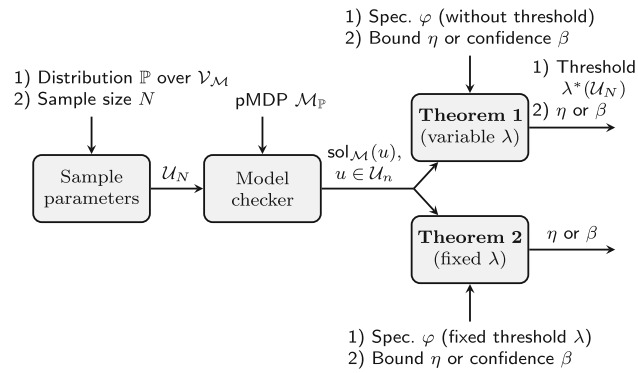


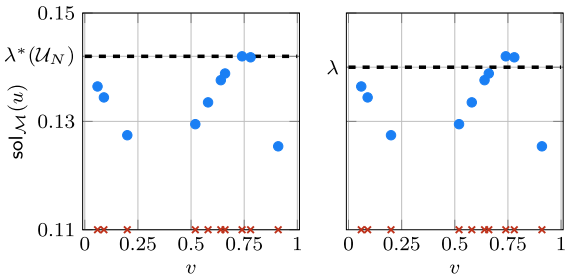
Fig. 5 Schematic overview of our approach. After obtaining the solutions, we choose to apply Theorem 1 (outputting a specification threshold $\lambda^*(\mathcal{U}_N)$ depending on $\mathcal{U}_N$) or Theorem 2 (inputting a fixed threshold λ)

we obtain solutions $\mathsf{sol}_{\mathcal{M}}(u)$ by sampling parameter instantiations u from $\mathcal{V}_{\mathcal{M}}$. In Sect. 5.2, we then first address a simpler yet related problem in which we let the specification φ depend on the set of sampled solutions at hand. In Sect. 5.3, we return to our original problem statement: we introduce our approach in which we keep the specification fixed. An algorithmic overview of both of these methods is shown in Fig. 5. Finally, we discuss the quality of the obtained lower bounds in Sect. 5.4

5.1 Obtaining solutions from parameter samples

We describe how we obtain solutions by sampling from the parameter space. Specifically, we define the set $\mathcal{U}_N = \{u_1, u_2, \ldots, u_N\}$ as the outcome of sampling N parameter instantiations from $\mathcal{V}_{\mathcal{M}}$ according to the probability distribution $\mathbb{P}$. Recall that we assume that these samples are independent and identically distributed. Thus, the set $\mathcal{U}_N$ of N samples is a random element drawn according to the product probability $\mathbb{P}^N = \mathbb{P} \times \cdots \times \mathbb{P}$ (N times) over the parameter probability distribution $\mathbb{P}$. For each sample $u \in \mathcal{U}_N$, we compute the resulting solution $\mathsf{sol}_{\mathcal{M}}(u)$, as shown in the following example.

Example 5 We continue from Example 3 on the upMC in Fig. 2. We sample $N = 10$ parameters from the (uniform) probability distribution of this upMC, which are shown in red on the x-axis in Fig. 6. The resulting solutions $\mathsf{sol}_{\mathcal{M}}(u_1), \ldots, \mathsf{sol}_{\mathcal{M}}(u_{10})$ are depicted as the blue points. As expected, these points indeed lie on the solution function curve shown in Fig. 3. In Fig. 6a and b, we check these solutions against a specification in two different ways: (1) In Fig. 6a, we first observe the solutions and then devise a threshold for the given measure, such that these samples *all satisfy the resulting specification*. That is, the threshold, denoted by $\lambda^*(\mathcal{U}_N)$, depends on the solutions at hand, such that $\mathcal{M}[u] \models \varphi$ for all $u \in \mathcal{U}_N$. For this example, the specifi-



(a) All samples $u \in \mathcal{U}_N$ correspond to $\mathsf{sol}_{\mathcal{M}}(u) \leq \lambda^*(\mathcal{U}_N)$. **(b)** Two samples $u \in \mathcal{U}$ correspond to $\mathsf{sol}_{\mathcal{M}}(u) > \lambda$.

Fig. 6 A set of $N = 10$ parameter instantiations $\mathcal{U}_N = \{u_1, \ldots, u_{10}\}$ (shown as red crosses) for the upMC in Fig. 2 and the solutions $\mathsf{sol}_{\mathcal{M}}(u)$. In Fig. 6a, the specification threshold $\lambda^*(\mathcal{U}_N)$ is chosen after observing the solutions such that all samples are satisfying; Fig. 6b uses a fixed threshold λ and has two violating samples

cation is $\mathbf{P}_{\leq\lambda^*(\mathcal{U}_N)}(\Diamond T)$, and the tightest threshold satisfying this condition is $\lambda^*(\mathcal{U}_N) = \max_{u\in\mathcal{U}_N} \mathsf{sol}_{\mathcal{M}}(u)$. (2) In Fig. 6b, we fix the specification with its threshold first, and then evaluate the number of samples satisfying the specification. This may lead to samples violating the specification (e.g., Fig. 6b has two violating samples).

In both cases, we can partition $\mathcal{U}_N$ into disjoint sets of samples that satisfy ($\mathcal{U}_{N_\varphi}$) or violate ($\mathcal{U}_{N_{\neg\varphi}}$) the specification, i.e., $\mathcal{U}_N = \mathcal{U}_{N_\varphi} \cup \mathcal{U}_{N_{\neg\varphi}}$. Note, that in the first case (Fig. 6a), the set of violating samples is empty by construction, i.e., $\mathcal{U}_{N_{\neg\varphi}} = \varnothing$. Let $N_\varphi = |\mathcal{U}_{N_\varphi}|$ denote the number of satisfying samples and $N_{\neg\varphi} = |\mathcal{U}_{N_{\neg\varphi}}|$ the number of violating samples.

5.2 Restriction to satisfying samples

Before solving the main problem introduced in Sect. 4, we consider a simpler setting to introduce some of the necessary ideas. Intuitively, we want to investigate the case where we adapt the specification (or rather the threshold in this specification) such that $\mathcal{U}_N = \mathcal{U}_{N_\varphi}$, or equivalently $N_\varphi = N$. This simpler setting is shown by Fig. 6a. Here, we do not fix a threshold λ for the specification φ a-priori, but instead derive a threshold $\lambda^*(\mathcal{U}_N)$ from the solutions at hand such that *all samples are satisfying*, i.e., we ensure that

$$\mathcal{M}[u] \models \varphi \text{ for all samples } u \in \mathcal{U}_N. \qquad \text{(Assumption A)}$$

Problem We raise the question: "*What is the probability that, given these N samples and a specification threshold that makes all samples satisfying, the next sampled parameter valuation u (on the x-axis of Fig.* 6a) *with the corresponding solution* $\mathsf{sol}_{\mathcal{M}}(u)$ *will also satisfy this specification?*" This probability is similar to the satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi)$ in Def. 4, but the threshold of specification φ is not fixed a-priori.

Result Using Theorem 1[2], we compute a lower bound η on this satisfaction probability that holds with a user-specified confidence probability β:

Theorem 1 *Let upMDP $\mathcal{M}_{\mathbb{P}}$ and the set $\mathcal{U}_N$ of $N \geq 1$ sampled parameters. For any set $\mathcal{U}_N$, choose threshold $\lambda^*(\mathcal{U}_N)$ of specification φ such that $\mathcal{U}_N = \mathcal{U}_{N_\varphi}$, and fix a confidence probability $\beta \in (0, 1)$. Then, it holds that*

$$\mathbb{P}^N\left\{F(\mathcal{M}_{\mathbb{P}}, \varphi) \geq (1-\beta)^{\frac{1}{N}}\right\} \geq \beta. \tag{2}$$

Applying Theorem 1 to the solutions in Fig. 6a, we compute that the satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi)$ with respect to specification φ with threshold $\lambda^*(\mathcal{U}_N) = 0.142$ is bounded from below by $\eta = 0.794$ (with a confidence probability of at least $\beta = 0.9$) and by $\eta = 0.631$ (with a confidence of at least $\beta = 0.99$).

Sample complexity More generally, Theorem 1 asserts that with a probability of at least β, the next sampled parameter from $\mathcal{V}_{\mathcal{M}}$ will satisfy the specification (with sample-dependent threshold $\lambda^*(\mathcal{U}_N)$) with a probability of at least $(1-\beta)^{\frac{1}{N}}$. Thus, the satisfaction probability is lower bounded by $\eta = (1-\beta)^{\frac{1}{N}}$ with high confidence, given that β is close to one. This high confidence is easily achieved for a sufficiently large number of samples N, as seen from the following corollary.

Corollary 1 *The sample size N necessary to obtain a desired lower bound $\eta \in (0, 1)$ on the satisfaction probability with at least a confidence of $\beta \in (0, 1)$ is*

$$N = ceil\left(\frac{\log(1-\beta)}{\log \eta}\right), \tag{3}$$

where the function ceil(x) rounds its argument $x \in \mathbb{R}$ upwards to the nearest integer.

Corollary 1 states that the sample size N is logarithmic in the confidence probability β. Thus, a significant improvement in β (i.e., closer to one) only requires a marginal increase in N. Similarly, increasing the sample size N improves the lower bound on the satisfaction probability η. For example, when increasing the number of samples in Fig. 6a to $N = 100$ (note that we still assume that $\mathcal{M}[u] \models \varphi$ for all $u \in \mathcal{U}_N$), Theorem 1 concludes that the satisfaction probability is lower bounded by 0.977 (with a confidence of at least $\beta = 0.9$) and by 0.955 (with a confidence of at least $\beta = 0.99$). Next, consider the extreme case, where β is infinitely close to one. We observe from Corollary 1 that such a confidence probability can only be obtained for $N = \infty$. Intuitively, this observation makes sense: we can only be absolutely certain of our lower bound

[2] We derive this theorem using Lemma 1, which is provided later on in Sect. 6.3

on the satisfaction probability, if we have based this estimate on infinitely many samples. In practice, our sample set is finite, and a typical confidence probability may be $\beta = 1 - 10^{-3}$.

5.3 Treatment of violating samples

We return to a setting with a fixed threshold λ and possible violating samples. In other words, we violate (Assumption A) that $\mathcal{M}[u] \models \varphi$ for all samples $u \in \mathcal{U}_N$. Consider again Fig. 6b, where for some of the samples $u \in \mathcal{U}_N$, the value $\mathsf{sol}_{\mathcal{M}}(u)$ exceeds λ, so $\mathcal{U}_N \neq \mathcal{U}_{N_\varphi}$, and Theorem 1 does not apply. Instead, we state Theorem 2[3] as a generalization that uses a fixed threshold λ and is also applicable in the presence of violating samples:

Theorem 2 *Let upMDP $\mathcal{M}_{\mathbb{P}}$ and the set $\mathcal{U}_N$ of $N \geq 1$ sampled parameters. Choose a confidence probability $\beta \in (0, 1)$. Then, it holds that*

$$\mathbb{P}^N \Big\{ F(\mathcal{M}_{\mathbb{P}}, \varphi) \geq t^*(N_{\neg\varphi}) \Big\} \geq \beta, \tag{4}$$

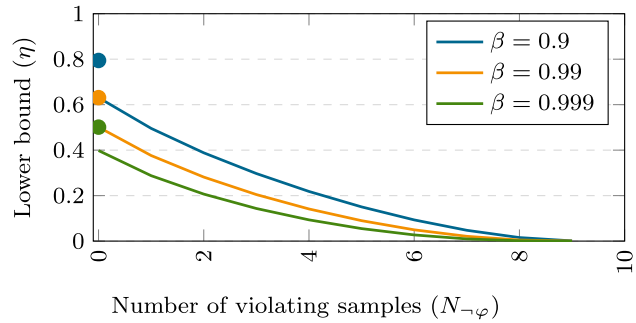
where $t^*(N) = 0$ for $N_{\neg\varphi} = N$, and for $k = 0, \ldots, N-1$, $t^*(k)$ is the solution of

$$\frac{1-\beta}{N} = \sum_{i=0}^{k} \binom{N}{i} (1-t)^i t^{N-i}. \tag{5}$$

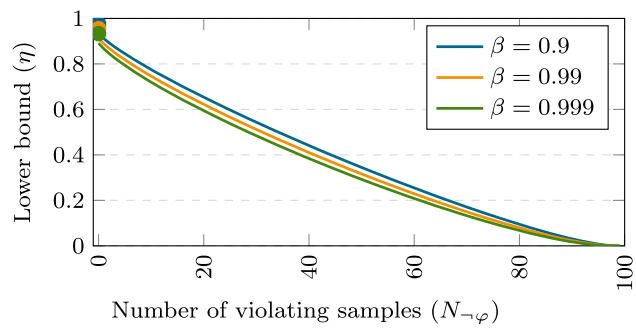
Theorem 2 solves the formal problem stated in Sect. 4. Recall that $N_{\neg\varphi}$ denotes the number of samples whose value $\mathsf{sol}_{\mathcal{M}}(u)$ violates the specification φ. Applying Theorem 2 to the solutions in Fig. 6b (with $N_{\neg\varphi} = 2$), we conclude that the satisfaction probability is bounded from below by 0.388 (with a confidence of at least $\beta = 0.9$) and by 0.282 (with $\beta = 0.99$). When we increase the number of samples to $N = 100$ and assume that $N_{\neg\varphi} = 20$, these results improve to the lower bounds 0.654 (with $\beta = 0.9$) and 0.622 (with $\beta = 0.99$). We note that the intuition in Corollary 1 about the relationships between the sample size N, lower bound η, and the confidence probability β also holds for Theorem 2.

5.4 Quality of the obtained lower bounds

Figure 7 illustrates how the number of violating samples, $N_{\neg\varphi}$, influences the quality of the lower bound on the satisfaction probability. The points at $N_{\neg\varphi} = 0$ are the bounds returned by Theorem 1, while the lines correspond to Theorem 2. Intuitively, the lower bound on the satisfaction probability computed by Theorem 2 decreases with an increased number of violating samples. Moreover, Theorem 1 yields a better lower bound than Theorem 2 (points versus the lines in Fig. 7), at the cost of not using a fixed threshold on the specification, and not being able to deal with violating samples.



(a) Number of samples $N = 10$.



(b) Number of samples $N = 100$.

Fig. 7 Lower bounds on the satisfaction probability as computed by Theorem 1 (shown as points at $N_{\neg\varphi} = 0$) and Theorem 2 (lines for different $N_{\neg\varphi} = 0, \ldots, N$)

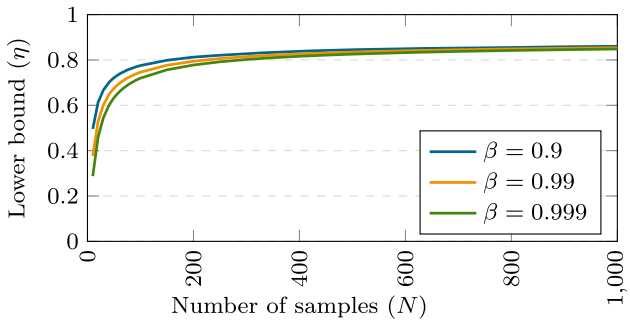
In Fig. 8, we fix the fraction of violating samples $N_{\neg\varphi}/N$ and plot the lower bounds on the satisfaction probability obtained using Theorem 2 for different values of N and β. Note that the lower bounds grow toward the fraction of violation for increased sample sizes. As also shown with Corollary 1, the confidence probability β only has a marginal effect on the obtained lower bounds.

Finally, we make the following remark with respect to the sample complexity of Theorems 1 and 2.

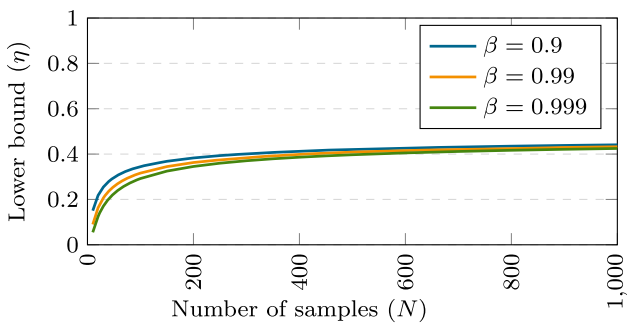
Remark 2 (*Independence to model size*) The number of samples needed to obtain a certain confidence probability in Theorems 1 and 2 is independent of the number of states, transitions, or parameters of the upMDP.[4]

[3] We derive this theorem using Lemma 2, which is provided later on in Sect. 6.4

[4] Despite this independence, note that the time to compute solutions via model checking still depends on the number of states and transitions of the instantiated MDP.



(a) Fraction of violating samples is $N_{\neg\varphi}/N = 10\%$.



(b) Fraction of violating samples is $N_{\neg\varphi}/N = 50\%$.

Fig. 8 Bounds on the satisfaction probability from Theorem 2 for fixed fractions $N_{\neg\varphi}/N$ of violated samples

6 Derivation of the main results

In this section, we explain how we obtain Theorems 1 and 2. Toward proving these theorems, we reformulate our problem statement into the domain of linear programs (LPs). First, we define the case where we account for all but a small fraction of the parameters instantiations $u \in \mathcal{V}_{\mathcal{M}}$ (recall that $\mathcal{V}_{\mathcal{M}}$ typically has infinite cardinality), which we formalize using a so-called *chance-constrained LP*. We remark that solving this chance-constrained LP directly is difficult [16]. Instead, we formalize our sampling-based approach, which is based on *scenario optimization*, and which only considers a finite number of sampled parameters $u \in \mathcal{U}_N$.

6.1 Chance-constrained LP reformulation

Recall from Sect. 4 that the problem is to compute a lower bound η on the satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi)$. In other words, when sampling a parameter instantiation $u \in \mathcal{V}_{\mathcal{M}}$ according to probability measure $\mathbb{P}$, compute a lower bound η on the probability that $\mathcal{M}[u] \models \varphi$. If the specification φ has a comparison operator $\leq$ and a threshold λ (e.g., $\varphi = \mathbf{P}_{\leq\lambda}(\Diamond T)$), then the condition $\mathcal{M}[u] \models \varphi$ is equivalent to $\mathsf{sol}_{\mathcal{M}}(u) \leq \lambda$. As the solution function (Def. 2) is a function of (only) the parameter instantiation, the solution $\mathsf{sol}_{\mathcal{M}}(u)$ is also a random variable with probability measure $\mathbb{P}$. Thus, we can formalize the problem of finding a lower bound η based on the following *chance-constrained LP*:

$$\underset{\tau \geq 0}{\text{minimize}} \quad \tau \tag{6a}$$

$$\text{subject to} \quad \Pr\left\{ u \in \mathcal{V}_{\mathcal{M}} \;\middle|\; \mathsf{sol}_{\mathcal{M}}(u) \leq \tau \right\} \geq \eta. \tag{6b}$$

Then, the satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi)$ is lower bounded by η, given that the optimal solution τ^* to (6) is at most λ.

Similarly, if the specification φ has a lower bound comparison operator $\geq$ and a threshold λ, we consider the following chance-constrained LP:

$$\underset{\tau \geq 0}{\text{maximize}} \quad \tau \tag{7a}$$

$$\text{subject to} \quad \Pr\left\{ u \in \mathcal{V}_{\mathcal{M}} \;\middle|\; \mathsf{sol}_{\mathcal{M}}(u) \geq \tau \right\} \geq \eta. \tag{7b}$$

Note that the differences between (6) and (7) are the optimization direction and the operator within the chance constraint. Solving these chance-constrained problems is very hard in general, in particular because the probability distribution of the parameters is unknown [16].

In what follows, we introduce our sampling-based approach to solve these problems with high confidence. For brevity, we assume specifications with a lower bound comparison operator as in (6), but modifying our results for other operators is straightforward.

6.2 Scenario optimization program

Instead of solving the chance-constrained LP in (6) directly, we compute probably approximately correct lower bounds on the satisfaction probability based on *scenario optimization* [12,13]. Specifically, we replace the chance constraint (6b), which asks for the satisfaction of a certain fraction of a set of infinitely many constraints, with a finite number of *hard constraints* that are induced by the sampled parameters $u \in \mathcal{U}_N$. The resulting optimization problem is called a *scenario program* [16] and is formulated as follows:

$$\underset{\tau \geq 0}{\text{minimize}} \quad \tau \tag{8a}$$

$$\text{subject to} \quad \mathsf{sol}_{\mathcal{M}}(u) \leq \tau \quad \forall u \in \mathcal{U}_N. \tag{8b}$$

Upon solving scenario program (8), we obtain a unique optimal solution $\tau^* = \max_{u \in \mathcal{U}_N} \mathsf{sol}_{\mathcal{M}}(u)$. In Sect. 6.3, we show that Theorem 1 follows from solving program (8) directly, while Theorem 2 corresponds to a setting where we deal with samples that violate the specification.

6.3 Restriction to satisfying samples

Consider the case where we directly solve the scenario program (8). In this case, the following theorem, which is based on [13, Theorem 2.4], enables us to compute a high-confidence lower bound on the satisfaction probability.

Lemma 1 *Let uMDP $\mathcal{M}_{\mathbb{P}}$, a specification φ with operator $\leq$, and the set $\mathcal{U}_N$ of $N \geq 1$ sampled parameters. Let τ^* be the optimal solution of (8), and choose a confidence probability $\beta \in (0, 1)$. Then, it holds that*

$$\mathbb{P}^N \left\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \mathsf{sol}_{\mathcal{M}}(u) \leq \tau^*\} \geq (1-\beta)^{\frac{1}{N}} \right\} \geq \beta. \quad (9)$$

Lemma 1 states that with a probability of *at least* β, the probability that $\mathsf{sol}_{\mathcal{M}}(u) \leq \tau^*$ for the next sampled parameter $u \in \mathcal{V}_{\mathcal{M}}$ is at least $(1-\beta)^{\frac{1}{N}}$. To derive Theorem 1 from Lemma 1, we choose the (sample-dependent) specification threshold $\lambda(\mathcal{U}_N) \geq \tau^*$ after solving the scenario program. Then, Theorem 1 follows directly by observing that $F(\mathcal{M}_{\mathbb{P}}, \varphi) \geq \Pr\{u \in \mathcal{V}_{\mathcal{M}} \,|\, \mathsf{sol}_{\mathcal{M}}(u) \leq \tau^*\}$, since $\lambda(\mathcal{U}_N) \geq \tau^*$. We provide the proof of Lemma 1, and thus of Theorem 1, in Appendix A.

6.4 Treatment of violating samples

We now derive Theorem 2, which assumes a fixed threshold λ. In this case, we cannot guarantee a-priori that $\lambda \geq \tau^*$, because some samples may induce a reachability probability above λ, as in Fig. 6b. Recall that $N_{\neg\varphi}$ is the number of samples that violate the specification. Loosely speaking, we relax the constraints for these $N_{\neg\varphi}$ samples, and compute the maximum probability over the remaining samples $\mathcal{U}_{N_\varphi} \subseteq \mathcal{U}_N$, which we write as $\tau^+ = \max_{u \in \mathcal{U}_{N_\varphi}} \mathsf{sol}_{\mathcal{M}}(u)$. The following theorem is adapted from [15, Theorem 2.1] and computes a high-confidence lower bound on the satisfaction probability, using the values of $N_{\neg\varphi}$ and τ^+.

Lemma 2 *Let uMDP $\mathcal{M}_{\mathbb{P}}$, a specification φ with operator $\leq$, and the set $\mathcal{U}_N$ of $N \geq 1$ sampled parameters. Fix a confidence probability $\beta \in (0, 1)$. Then, it holds that*

$$\mathbb{P}^N \left\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} | \mathsf{sol}_{\mathcal{M}}(u) \leq \tau^+\} \geq t^*(N_{\neg\varphi}) \right\} \geq \beta, \quad (10)$$

where $t^*(N) = 0$ for $N_{\neg\varphi} = N$, and for $k = 0, \ldots, N-1$, $t^*(k)$ is the solution of

$$\frac{1-\beta}{N} = \sum_{i=0}^{k} \binom{N}{i} (1-t)^i t^{N-i}. \quad (11)$$

Lemma 2 asserts that with a probability of *at least* β, the probability that $\mathsf{sol}_{\mathcal{M}}(u) \leq \tau^+$ for the next sampled parameter $u \in \mathcal{V}_{\mathcal{M}}$ is at least $t^*(N_{\neg\varphi})$, given that $N_{\neg\varphi}$ samples violate the specification φ. Theorem 2 follows directly from Lemma 2, by observing that by construction, $\tau^+ \leq \lambda$. We provide the proof of Lemma 2, and thus of Theorem 2, in Appendix A..

We note that (11) is the cumulative distribution function of a beta distribution with $N_{\neg\varphi} + 1$ and $N - N_{\neg\varphi}$ degrees of freedom [16], which can easily be solved numerically for t. Moreover, we can speed up the computations at run-time, by tabulating the solutions to (11) for all relevant values of N, β and $N_{\neg\varphi}$ up front.

7 Numerical examples

We implemented our approach in Python using the model checker Storm [25] to construct and analyze samples of MDPs. Our implementation is available at https://doi.org/10.5281/zenodo.6674059 All computations ran on a computer with 32 3.7 GHz cores, and 64 GB of RAM.

First, we apply our method to the UAV motion planning example introduced in Sect. 2. Thereafter, we report on a set of well-known benchmarks used in parameter synthesis [36] that are, for instance, available on the website of the tools PARAM [30] or part of the PRISM benchmark suite [38]. We demonstrate that with our method, we can either specify a lower bound η on the satisfaction probability and compute with what confidence probability β we can guarantee this lower bound, or we can do this in the opposite direction (i.e., specify η and compute β).

7.1 UAV motion planning

Setup Recall the example from Sect. 2 of a UAV which needs to deliver a payload while avoiding obstacles. The weather conditions are uncertain, and this uncertainty is reflected in the parameters of the uMDP. For the distributions over parameter values, that is, over weather conditions, we consider the following three cases:

1. we assume a uniform distribution over the different weather conditions in each zone;
2. the probability for a weather condition inducing a wind direction that pushes the UAV northbound (i.e., into the positive y-direction) is twice as likely as in other directions;
3. it is twice as likely to push the UAV westbound (i.e., into the negative x-direction).

Trajectories We depict example trajectories of the UAV for these three cases in Fig. 1. The trajectory depicted by the black line represents a simulated trajectory for the first case (uniform distribution), taking a direct route to reach the target area. Similarly, the trajectories shown by the dotted

Table 1 Lower bounds η on the (un)satisfaction probability for the UAV benchmark with $N = 5\,000$ samples

Confidence probability	$\beta = 0.9$		$\beta = 0.99$		$\beta = 0.999$		$\beta = 0.9999$	
Weather condition	η, sat	η, unsat	η, sat	η, unsat	η, sat	η, unsat	η, sat	η, unsat
1. Uniform distribution	0.91138	0.0583	0.90928	0.0567	0.90735	0.05528	0.90555	0.05398
2. Stronger northbound wind	0.77878	0.17483	0.77577	0.17217	0.77302	0.16978	0.77048	0.1676
3. Stronger westbound wind	0.7768	0.17664	0.77378	0.17397	0.77103	0.17157	0.76847	0.16938

purple and dashed blue lines are simulated trajectories for the second (stronger northbound wind, i.e., positive x-direction) and third cases (stronger westbound wind, i.e., positive y-direction), respectively. Under these two weather conditions, the UAV takes different paths toward the goal in order to account for the stronger wind. In particular, for the case with northbound wind, we observe that the UAV is able to fly close to the obstacle at the right bottom. By contrast, for the case with westbound wind, the UAV avoids to get close to this obstacle, as the wind may push the UAV into the obstacles, and then reaches the target area.

Bounds on satisfaction probabilities We sample $N = 1000$ parameters for each case and consider different confidence probabilities β between 0.9 and 0.9999. Specifically, we consider the specification $\varphi = \mathbf{P}_{\geq 0.9}(\Diamond T)$, which is satisfied if the probability to safely reach the goal region is at least 90%. For all three weather conditions, we compute the lower bounds η on both the satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi)$ and the unsatisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \neg\varphi)$, using Theorem 2.

The results are presented in Table 1. The highest lower bound on the satisfaction probability is given by the first weather condition, and is $\eta = 0.911$ (for $\beta = 0.9$) and $\eta = 0.906$ (for $\beta = 0.9999$). In other words, under a uniform distribution over the weather conditions, the UAV will (with a confidence of at least $\beta = 0.9999$) satisfy the specification on at least 90.6% of the days. The second and third weather conditions lead to $\eta = 0.770$ and $\eta = 0.768$ (for $\beta = 0.9999$), respectively, showing that it is harder to navigate around the obstacles with non-uniform probability distributions over the parameters. The average time to run a full iteration of our approach on this uMDP with 900 parameters and around 10 000 states (i.e., performing the sampling, model checking, and computing the lower bounds η) with 5 000 parameter samples is 9.5 minutes.

7.2 Parameter synthesis benchmarks

Setup In our second set of benchmarks, we adopt parametric MDPs and pMCs from [49]. We adapt the *Consensus* protocol [3] and the *Bounded Retransmission Protocol* (brp) [22,34] to uMDPs; the *Crowds Protocol* (crowds) [52] and the *NAND Multiplexing* benchmark (nand) [31] become uMCs. Essentially, the PLA technique from [49] allows to approximate the percentage of instantiations that satisfy (or do not satisfy) a specification, while assuming a uniform distribution over the parameter space. Table 2 lists the specification checked (φ) and the number of parameters, states, and transitions for all benchmarks. Note that the satisfying regions reported in Table 2 approximate $F(\mathcal{M}_{\mathbb{P}}, \varphi)$, while the unsatisfying regions approximate $F(\mathcal{M}_{\mathbb{P}}, \neg\varphi)$. We provide these numbers as a baseline only: the computation via PLA cannot scale to more than tens of parameters [49] and cannot cope with unknown distributions. For all benchmarks, we assume a uniform distribution over the parameters.

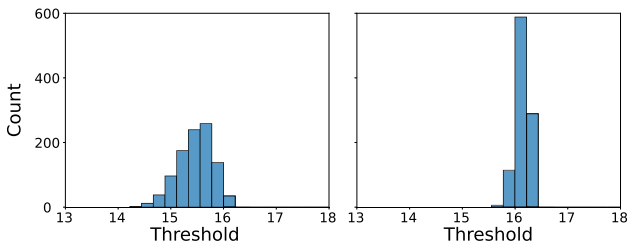
Specifications with variable thresholds λ We demonstrate Theorem 1 on brp (16,5) with a variable threshold $\lambda^*(\mathcal{U}_N)$ in specification $\varphi = \mathbb{E}_{\leq \lambda^*(\mathcal{U})N)}(\Diamond T)$. We use either $N = 1\,000$ or 10 000 parameter samples and compute the tightest threshold $\lambda^*(\mathcal{U}_N)$ such that all samples are satisfying. As explained in Example 5, this tightest threshold is $\lambda^*(\mathcal{U}_N) = \max_{u \in \mathcal{U}_N} \mathsf{sol}_{\mathcal{M}}(u)$. We repeat both cases ten thousand times and show a histogram of the obtained thresholds in Fig. 9. The corresponding lower bounds on the satisfaction probability (which only depend on N and β) are $\eta = 0.9954$ (for $N = 1\,000$) and $\eta = 0.9995$ (for $N = 10\,000$). We observe from Fig. 9 that for a higher number of samples, the thresholds are, on average, higher and their variability is lower. These results are explainable, since the threshold is computed as the maximum of all solutions.

Computing β for a given η We use Theorem 2 to compute the confidence probabilities β that the approximate satisfying regions in Table 2 are indeed a lower bound on the actual satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi)$ (and on $F(\mathcal{M}_{\mathbb{P}}, \neg\varphi)$ for the unsatisfying regions). Thus, we let η be the approximate (un)satisfying regions in Table 2, and sample a desired number of parameters (between $N = 1\,000$ and 25 000). For every instance, we then compute the confidence probability β using Theorem 2.

For every benchmark and sample size, we report the average confidence probabilities β over 10 full iterations of the same benchmark in Table 3. Furthermore, we list the run time (in seconds) per 1 000 samples for each instance. The results in Table 3 show that we can conclude with high confidence that the (un)satisfying regions are indeed a lower bound on the actual (un)satisfaction probabilities. In partic-

Table 2 Information for the benchmark instances and the approximate (un)satisfaction probabilities taken from [49]

Benchmark	Instance	φ	#pars	Model size		Approximate (un)satisfying regions	
				#states	#trans	Satisfying region	Unsatisfying region
brp	(256,5)	$\mathbf{P}_{\leq 0.5}(\Diamond T)$	2	19 720	26 627	0.055	0.898
	(16,5)	$\mathbb{E}_{\leq 3}(\Diamond T)$	4	1 304	1 731	0.275	0.676
	(32,5)	$\mathbb{E}_{\leq 3}(\Diamond T)$	4	2 600	3 459	0.232	0.718
crowds	(10,5)	$\mathbf{P}_{\leq 0.9}(\Diamond T)$	2	104 512	246 082	0.537	0.413
	(15,7)	$\mathbf{P}_{\leq 0.9}(\Diamond T)$	2	8 364 409	25 108 729	0.411	0.539
nand	(10,5)	$\mathbf{P}_{\geq 0.05}(\Diamond T)$	2	35 112	52 647	0.218	0.733
	(25,5)	$\mathbf{P}_{\geq 0.05}(\Diamond T)$	2	865 592	1 347 047	0.206	0.744
consensus	(2,2)	$\mathbf{P}_{\geq 0.25}(\Diamond T)$	2	272	492	0.280	0.669
	(4,2)	$\mathbf{P}_{\geq 0.25}(\Diamond T)$	4	22 656	75 232	0.063	0.888



(a) $N = 1\,000$ samples. **(b)** $N = 10\,000$ samples.

Fig. 9 Histograms of the obtained thresholds $\lambda^*(\mathcal{U}_N)$

ular for $N = 25\,000$ samples, most confidence probabilities are very close to one. For example, for the crowds benchmark, instance (10,5) with $N = 25\,000$, we obtain a confidence probability of $\beta = 0.99943$ on the unsatisfying region of size 0.413. Thus, the probability that the approximate unsatisfying region of 0.413 in Table 2 is *not* a lower bound on the actual unsatisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \neg\varphi)$ is less than $1 - \beta = 0.00057$. Moreover, in line with Remark 2, larger models do not (in general) lead to worse confidence bounds (although model checking times do typically increase with the size of the MDP, cf. Table 3).

The instance for which we obtained the worst confidence probability is the unsatisfying probability of nand (10,5), namely $\beta = 0.975$. Recomputing the results of [49] with a much smaller tolerance revealed that the approximate unsatisfying region of 0.733 was already a very tight lower bound (the best bound we were able to compute was 0.747). As such, we could only conclude with a confidence of $\beta = 0.975$ that $\eta = 0.733$ is a correct lower bound (as shown in Table 3, for $N = 25\,000$).

Computing η for a given β Conversely, we can also use Theorem 2 to compute the best lower bound η on the (un)satisfaction probability that holds with at least a confidence probability β. For each benchmark, we sample $N = 25\,000$ parameters and apply Theorem 2 for increasing confidence probabilities β. We report the resulting bounds η in Table 4. We observe that the obtained values of η are slightly more conservative (i.e., lower) for higher values of β. This observation is indeed intuitive: to reduce our chance $1 - \beta$ of obtaining an incorrect bound on the (un)satisfaction probability, the value of η must be more conservative. Moreover, increasing the confidence probability β only marginally reduces the obtained lower bound η. For example, the obtained lower bound on the satisfaction probability for brp (256,5) with $\beta = 0.9$ is $\eta = 0.07244$, while for $\beta = 0.9999$, it is only reduced to $\eta = 0.07036$ (a reduction of only 0.21%). This observation confirms the important result of Corollary 1: a high confidence probability β can typically be obtained without sacrificing the tightness of the obtained lower bound η.

Recall that, based on Table 3, we can only confirm the validity of the lower bound $\eta = 0.733$ on the unsatisfying region for nand (10,5) with a confidence probability of $\beta = 0.975$. Interestingly, Table 4 shows that we can guarantee a marginally weaker lower bound of $\eta = 0.73271$ with a confidence probability $\beta = 0.9999$. In other words, by weakening our lower bound η by an almost negligible amount of 0.0003, we increase the confidence on the results from 97.51% to a remarkable 99.99%. This highlights that the confidence probability β is extremely sensitive for the lower bound η, especially for high sample sizes N.

8 Discussion and related work

The so-called *parameter synthesis* problem considers computing parameter values such that the induced nonparametric MDP satisfies the specification for some strategy. Most works on parameter synthesis focus on finding one parameter value that satisfies the specification. The approaches involve computing a rational function of the reachability probabilities [6,23,27,30], utilizing convex optimization [20,21], and sampling-based methods [18,41]. The problem of whether

Table 3 Average confidence probabilities β for different sample sizes N, and run times per 1 000 samples

# samples	Instance	1 000		2 500		5 000		10 000		25 000		
Benchmark		β, sat	β, unsat	β, sat	β, unsat	β, sat	β, unsat	β, sat	β, unsat	β, sat	β, unsat	Time (s)
brp	(256,5)	0.42586	0.14955	0.91278	0.53890	0.99927	0.91447	1.00000	0.99957	1.00000	1.00000	1.296
	(16,5)	0.05699	0.03247	0.22192	0.10293	0.62397	0.35332	0.95612	0.92778	1.00000	1.00000	0.341
	(32,5)	0.05126	0.07365	0.21862	0.27669	0.50731	0.64205	0.89816	0.91258	1.00000	1.00000	0.344
crowds	(10,5)	0.04770	0.03339	0.22113	0.07921	0.58451	0.31034	0.94009	0.65727	1.00000	0.99943	0.119
	(15,7)	0.06568	0.02839	0.15451	0.05446	0.51860	0.31260	0.95223	0.71819	1.00000	1.00000	0.174
nand	(10,5)	0.18263	0.01101	0.62057	0.02775	0.97510	0.07370	1.00000	0.37567	1.00000	0.97509	4.097
	(25,5)	0.02938	0.20327	0.14312	0.51272	0.40151	0.82369	0.62267	0.99994	0.99884	1.00000	156.654
consensus	(2,2)	0.06282	0.02833	0.23683	0.14101	0.65357	0.44217	0.98990	0.93097	1.00000	1.00000	0.450
	(4,2)	0.13668	0.41820	0.48546	0.90556	0.86663	0.99999	0.99998	1.00000	1.00000	1.00000	26.575

Table 4 Lower bounds η on the (un)satisfaction probability for $N = 25\,000$ samples

Confidence probability	Instance	$\beta = 0.9$		$\beta = 0.99$		$\beta = 0.999$		$\beta = 0.9999$	
Benchmark		η, sat	η, unsat	η, sat	η, unsat	η, sat	η, unsat	η, sat	η, unsat
brp	(256,5)	0.07244	0.91221	0.07168	0.91135	0.07099	0.91056	0.07036	0.90982
	(16,5)	0.28787	0.68619	0.28653	0.68481	0.28531	0.68353	0.28417	0.68234
	(32,5)	0.24356	0.73176	0.24229	0.73044	0.24113	0.72922	0.24005	0.72808
crowds	(10,5)	0.55106	0.42091	0.54957	0.41945	0.54821	0.41810	0.54695	0.41685
	(15,7)	0.42397	0.54798	0.42250	0.54650	0.42115	0.54514	0.41990	0.54387
nand	(10,5)	0.23909	0.73637	0.23783	0.73506	0.23668	0.73384	0.23561	0.73271
	(25,5)	0.20979	0.76673	0.20858	0.76546	0.20748	0.76430	0.20647	0.76321
consensus	(2,2)	0.29383	0.68009	0.29248	0.67870	0.29125	0.67742	0.29010	0.67622
	(4,2)	0.07367	0.91086	0.07291	0.91000	0.07221	0.90921	0.07157	0.90846

there exists a value in the parameter space that satisfies a reachability specification is ETR-complete[5] [56], and finding a satisfying parameter value is exponential in the number of parameters.

The work in [4] considers the analysis of Markov models in the presence of uncertain rewards, utilizing statistical methods to reason about the probability of a parametric MDP satisfying an expected cost specification. This approach is restricted to reward parameters and does not explicitly compute confidence bounds. The work in [46] obtains data-driven bounds on the parameter ranges and then uses parameter synthesis techniques to validate properties for all parameter values in this range. Paper [39] computes bounds on the long-run probability of satisfying a specification with probabilistic uncertainty for Markov chains. Other related techniques include multi-objective model checking to maximize the average performance with probabilistic uncertainty sets [51], sampling-based methods which minimize the *regret* with uncertainty sets [1], and Bayesian reasoning to compute parameter values that satisfy a metric temporal logic specification on a continuous-time Markov chain (CTMC) [10]. Sampling-based methods similar to ours for verifying CTMCs with uncertain rates are developed in [5]. Finally, the work in [2] considers a variant of the problem in this paper where parameter values cannot be observed and thus must be learned. The paper formulates the strategy synthesis problem as a computationally harder partially observable Markov decision process (POMDP) synthesis problem and uses off-the-shelf point-based POMDP methods [17,44].

The works [47,57] consider the verification of MDPs with convex uncertainties. However, the uncertainty sets for different states in an MDP are restricted to be independent, which does not hold in our problem setting where we have parameter dependencies.

Uncertainties in MDPs have received quite some attention in the artificial intelligence and planning literature. Interval MDPs [29,47] use probability intervals in the transition probabilities. Dynamic programming, robust value iteration and robust strategy iteration have been developed for MDPs with uncertain transition probabilities whose parameters are statistically independent, also referred to as rectangular, to find a strategy ensuring the highest expected total reward at a given confidence level [42,57]. The work in [55] relaxes

[5] The ETR satisfiability problem is to decide if there exists a satisfying assignment to the real variables in a Boolean combination of a set of polynomial inequalities. It is known that NP $\subseteq$ ETR $\subseteq$ PSPACE.

this independence assumption a bit and determines a strategy that satisfies a given performance with a pre-defined confidence provided an observation history of the MDP is given by using conic programming. State-of-the art exact methods can handle models of up to a few hundred of states [35]. Multi-model MDPs [53] treat distributions over probability and cost parameters and aim at finding a single strategy maximizing a weighted value function. This problem is NP-hard for deterministic strategies and PSPACE-hard for history-dependent strategies.

9 Conclusion

We have presented a new sampling-based approach to analyze uncertain Markov models. Theoretically, we have shown how to effectively and efficiently bound the probability that any randomly drawn sample satisfies a temporal logic specification. Furthermore, we have shown the computational tractability of our approaches by means of well-known benchmarks and a new, dedicated case study. In the future, we plan to exploit our approaches for more involved models such as Markov automata [32]. Another line of future work will be a closer integration with a parameter synthesis framework.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

A Proofs

In this section, we provide the proofs of our main theoretical contributions. Since our theorems and lemmas are tailored to specifications φ with comparison operator $\leq$, we also use these assumptions throughout the proofs. The proofs for the case where we have a specification φ with comparison operator $\geq$ are analogous to the difference between LPs (6b) and (7b): we use maximize instead of minimize, and the operator in the constraint changes sign.

A.1 Proof of Lemma 1 and Theorem 1

We first prove Lemma 1, and then show that Theorem 1 follows directly. Let us rewrite LP (8) in a more compact way.

To this end, for a parameter sample $u \in \mathcal{U}_N$, let C_u denote the interval of values for τ, for which constraint (8b) is satisfied. Note that for a specification φ with comparison operator $\leq$, we have $C_u = [\mathsf{sol}_{\mathcal{M}}(u),\ +\infty)$, i.e., C_u is lower bounded by the solution $\mathsf{sol}_{\mathcal{M}}(u)$. Using C_u, we reformulate the scenario program (8) as the equivalent program

$$\begin{aligned} \underset{\tau \geq 0}{\text{minimize}} \quad & \tau \\ \text{subject to} \quad & \tau \in \bigcap_{u \in \mathcal{U}_N} C_u. \end{aligned} \tag{12}$$

Note that (12) is exactly in the form of the scenario program formulated in [13]. Let τ^* denote the optimal value to the scenario program with respect to sample set $\mathcal{U}_N$, and let u be an independently sampled parameter from $\mathcal{V}_{\mathcal{M}}$ according to $\mathbb{P}$. Then, Theorem 2.4 of [13] states that the *cumulative distribution function* of the probability for the set C_u associated with sampled parameter u to *violate* the optimal solution τ^*, that is $\tau^* \notin C_u$, is written as follows:

$$\begin{aligned} &\mathbb{P}^N \Big\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \tau^* \notin C_u\} > \epsilon \Big\} \\ &\qquad \leq \sum_{i=0}^{d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}, \end{aligned} \tag{13}$$

where $\epsilon \in (0, 1)$ bounds the violation probability, and d is the number of decision variables of (12). Since we have $d = 1$, and we are after the satisfaction probability (rather than the violation probability), we simplify (13) as

$$\mathbb{P}^N \Big\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \tau^* \in C_u\} < 1-\epsilon \Big\} \leq (1-\epsilon)^N \tag{14}$$

$$\mathbb{P}^N \Big\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \tau^* \in C_u\} \geq 1-\epsilon \Big\} \geq 1-(1-\epsilon)^N. \tag{15}$$

Let $\beta = 1-(1-\epsilon)^N$, which implies that $\epsilon = 1-(1-\beta)^{\frac{1}{N}}$. Moreover, the event that $\tau^* \in C_u$ is equivalent to $\mathsf{sol}_{\mathcal{M}}(u) \leq \tau^*$, so (15) reduces to

$$\mathbb{P}^N \Big\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \mathsf{sol}_{\mathcal{M}}(u) \leq \tau^*\} \geq (1-\beta)^{\frac{1}{N}} \Big\} \geq \beta, \tag{16}$$

which equals the desired expression in (9) for Lemma 1.

Finally, we show that Theorem 1 follows from Lemma 1. Because $\tau^* \leq \lambda^*(\mathcal{U}_N)$, with $\lambda^*(\mathcal{U}_N)$ the sample-dependent threshold of specification φ, the inner probability in (16) is a lower bound on the satisfaction probability $F(\mathcal{M}_{\mathbb{P}}, \varphi)$:

$$F(\mathcal{M}_{\mathbb{P}}, \varphi) \geq \Pr(u \in \mathcal{V}_{\mathcal{M}} \mid \tau^* \in C_u). \tag{17}$$

By combining (16) with (17), we find that

$$\mathbb{P}^N \Big\{ F(\mathcal{M}_{\mathbb{P}}, \varphi) \geq (1-\beta)^{\frac{1}{N}} \Big\} \geq \beta, \tag{18}$$

and thus, the claim in Theorem 1 follows.

A.2 Proof of Lemma 2 and Theorem 2

We first prove Lemma 2, and then show that Theorem 2 follows directly. We modify the scenario program (8) as a scenario program with *discarded samples* [15], which allows for the removal of undesirable constraints:

$$\underset{\tau \geq 0}{\text{minimize}} \quad \tau \tag{19a}$$

$$\text{subject to} \quad \mathsf{sol}_{\mathcal{M}}(u) \leq \tau \quad \forall u \in \mathcal{U}_N \backslash \mathcal{Q}, \tag{19b}$$

where we introduced the sample removal set $\mathcal{Q}$, which accounts for a subset of samples whose constraints have been discarded. We explicitly write the dependency of the optimal solution $\tau^*_{|\mathcal{Q}|}$ on the number of discarded samples $|\mathcal{Q}|$. Critically, samples are removed based on the following rule:

Lemma 3 *The sample removal set $\mathcal{Q} \subseteq \{1, \ldots, N\}$ is obtained by iteratively removing the* active constraints *from (19), i.e., the samples $u \in \mathcal{U}_N$ for which $\mathsf{sol}_{\mathcal{M}}(u) = \tau^*_{|\mathcal{Q}|}$.*

Note that the active constraint may not be unique, e.g., if the solution $\mathsf{sol}_{\mathcal{M}}(u_1) = \mathsf{sol}_{\mathcal{M}}(u_2)$ for $u_1 \neq u_2$. In that case, a suitable *tie-break rule* may be used to select a constraint to discard, as discussed in [14].

The main difference between programs (19) and (8) is that instead of enforcing the constraint for every sample $u \in \mathcal{U}_N$, we only enforce the constraint for a subset of samples $u \in \mathcal{U}_N \backslash \mathcal{Q}$. Based on the solution to (19), Theorem 2.1 of [15] bounds the *cumulative distribution function* of the violation probability, in a similar manner as the guarantees given by (13):

$$\begin{aligned} &\mathbb{P}^N \left\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \tau^*_{|\mathcal{Q}|} \notin C_u\} > \epsilon \right\} \\ &\quad \leq \binom{\mathcal{Q}+d-1}{\mathcal{Q}} \sum_{i=0}^{|\mathcal{Q}|+d-1} \binom{N}{I} \epsilon^i (1-\epsilon)^{N-i} \\ &\quad = \sum_{i=0}^{|\mathcal{Q}|} \binom{N}{I} \epsilon^i (1-\epsilon)^{N-i}, \end{aligned} \tag{20}$$

where $\epsilon \in (0, 1)$ bounds the violation probability, $d = 1$ is the number of decision variables of (19), and $|\mathcal{Q}|$ is the cardinality of the sample removal set. As our goal is to bound the satisfaction probability (rather than the violation probability), we define $t = 1 - \epsilon$, and rewrite (20) as

$$\begin{aligned} &\mathbb{P}^N \left\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \tau^*_{|\mathcal{Q}|} \in C_u\} < 1 - \epsilon \right\} \\ &\qquad \leq \sum_{i=0}^{|\mathcal{Q}|} \binom{N}{I} \epsilon^i (1-\epsilon)^{N-i} \end{aligned} \tag{21}$$

$$\begin{aligned} &\mathbb{P}^N \left\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \tau^*_{|\mathcal{Q}|} \in C_u\} \geq t \right\} \\ &\qquad \geq 1 - \sum_{i=0}^{|\mathcal{Q}|} \binom{N}{I} (1-t)^i t^{N-i}. \end{aligned} \tag{22}$$

We equate the right-hand side of (22) to $1 - \frac{1-\beta}{N}$, where $\beta \in (0, 1)$ is a confidence probability (typically close to one):

$$\mathbb{P}^N \left\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \tau^*_{|\mathcal{Q}|} \in C_u\} \geq t^*(|\mathcal{Q}|) \right\} \geq 1 - \frac{1-\beta}{N}, \tag{23}$$

where $t^*(|\mathcal{Q}|)$ is the solution to

$$1 - \frac{1-\beta}{N} = 1 - \sum_{i=0}^{|\mathcal{Q}|} \binom{N}{I} (1-t)^i t^{N-i}. \tag{24}$$

We divide the confidence level by N to account for all N possible values for $|\mathcal{Q}|$, ranging from 0 to $N-1$. The value of $|\mathcal{Q}|$ that is actually needed depends on the sample set at hand, and is, therefore, not known a-priori (i.e., before observing the actual samples). For brevity, denote by $\mathcal{A}_n$ the event that

$$Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \tau^*_n \in C_u\} \geq t^*(n). \tag{25}$$

The probability for this event to hold is $\mathbb{P}^N\{\mathcal{A}_n\} \geq 1 - \frac{1-\beta}{N}$, and its complement $\mathcal{A}'_n$ has a probability of $\mathbb{P}^N\{\mathcal{A}'_n\} \leq \frac{1-\beta}{N}$. Based on Boole's inequality, it holds that

$$\mathbb{P}^N \left\{ \bigcup_{i=0}^{N-1} \mathcal{A}'_n \right\} \leq \sum_{i=0}^{N-1} \mathbb{P}^N \{\mathcal{A}'_n\} \leq \frac{1-\beta}{N} N = 1 - \beta. \tag{26}$$

Thus, the probability of the intersection of all events is

$$\mathbb{P}^N \left\{ \bigcap_{i=0}^{N-1} \mathcal{A}_n \right\} = 1 - \mathbb{P}^N \left\{ \bigcup_{i=0}^{N-1} \mathcal{A}'_n \right\} \geq \beta. \tag{27}$$

In other words, the bounds on the satisfaction probability given by (23) hold *simultaneously for all values* $|\mathcal{Q}| = 0, \ldots, N-1$ with a confidence probability of at least β.

After observing the samples at hand, we determine the actual value of $N_{\neg\varphi}$ and plug it as $|\mathcal{Q}|$ into (23). The probability that this expression holds cannot be smaller than that of the intersection of all events in (27). Hence, we obtain

$$\mathbb{P}^N \left\{ \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \mathsf{sol}_{\mathcal{M}}(u) \leq \tau^*_{|\mathcal{Q}|}\} \geq t^*(N_{\neg\varphi}) \right\} \geq \beta. \tag{28}$$

Recall from Sect. 6.4 that $\tau^+ = \max_{u \in \mathcal{U}_{N_\varphi}} \mathsf{sol}_{\mathcal{M}}(u)$, which is, by construction, equivalent to $\tau^*_{|\mathcal{Q}|}$ under $|\mathcal{Q}|$ removed

samples. Thus, (28) is equivalent to (10), and the definition of $t^*(|\mathcal{Q}|)$ in (24) equals (11), so the claim in Lemma 2 follows.

Finally, to show that Theorem 2 follows directly from Lemma 2, we note that $\tau^+ = \tau^*_{|\mathcal{Q}|} \leq \lambda$, so it must hold that

$$F(\mathcal{M}_{\mathbb{P}}, \varphi) \geq \Pr\{u \in \mathcal{V}_{\mathcal{M}} \mid \mathsf{sol}_{\mathcal{M}}(u) \leq \tau^+\}. \tag{29}$$

By combining (28) with (29), we find that

$$\Pr\left\{F(\mathcal{M}_{\mathbb{P}}, \varphi) \geq t^*(N_{\neg\varphi})\right\} \geq \beta, \tag{30}$$

and thus, we conclude the proof of Theorem 2.

References

1. Ahmed, A., Varakantham, P., Lowalekar, M., Adulyasak, Y., Jaillet, P.: Sampling based approaches for minimizing regret in uncertain Markov decision processes (MDPs). J. Artif. Intell. Res. **59**, 229–264 (2017)
2. Arming, S., Bartocci, E., Chatterjee, K., Katoen, J.P., Sokolova, A.: Parameter-Independent Strategies for pMDPs via POMDPs. In: QEST, pp. 53–70. Springer (2018)
3. Aspnes, J., Herlihy, M.: Fast randomized consensus using shared memory. J. Algorithms **15**(1), 441–460 (1990)
4. Bacci, G., Hansen, M., Larsen, K.G.: Model Checking Constrained Markov Reward Models with Uncertainties. In: QEST, pp. 37–51 (2019)
5. Badings, T.S., Jansen, N., Junges, S., Stoelinga, M., Volk, M.: Sampling-based verification of ctmcs with uncertain rates. In: International Conference on Computer Aided Verification (to appear). Springer (2022)
6. Baier, C., Hensel, C., Hutschenreiter, L., Junges, S., Katoen, J.P., Klein, J.: Parametric Markov chains: PCTL complexity and fraction-free Gaussian elimination. Inf. Comput. **272**, 104504 (2020)
7. Baier, C., Katoen, J.P.: Principles of Model Checking. MIT Press, London (2008)
8. Basu, S., Pollack, R., Roy, M.: Algorithms in Real Algebraic Geometry. Springer, Berlin (2010)
9. Bertsekas, D.P., Tsitsiklis, J.N.: Introduction to Probability. Athena Scientinis, London (2000)
10. Bortolussi, L., Silvetti, S.: Bayesian statistical parameter synthesis for linear temporal properties of stochastic models. In: TACAS (2), Lecture Notes in Computer Science, vol. 10806, pp. 396–413. Springer (2018)
11. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, New York (2004)
12. Calafiore, G.C., Campi, M.C.: The scenario approach to robust control design. IEEE Trans. Autom. Contr. **51**(5), 742–753 (2006)
13. Campi, M.C., Garatti, S.: The exact feasibility of randomized solutions of uncertain convex programs. SIAM J. Optim. **19**(3), 1211–1230 (2008)
14. Campi, M.C., Garatti, S.: The exact feasibility of randomized solutions of uncertain convex programs. SIAM J. Optim. **19**(3), 1211–1230 (2008)
15. Campi, M.C., Garatti, S.: A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. J. Optim. Theory Appl. **148**(2), 257–280 (2011)
16. Campi, M.C., Garatti, S.: Introduction to the scenario approach. SIAM **2**, 996 (2018)
17. Cassandra, A., Littman, M.L., Zhang, N.L.: Incremental Pruning: A Simple, Fast, Exact Method for Partially Observable Markov Decision Processes. In: UAI, pp. 54–61 (1997)
18. Chen, T., Hahn, E.M., Han, T., Kwiatkowska, M., Qu, H., Zhang, L.: Model Repair for Markov Decision Processes. In: TASE, pp. 85–92. IEEE CS (2013)
19. Cubuktepe, M., Jansen, N., Junges, S., Katoen, J., Topcu, U.: Scenario-based verification of uncertain mdps. In: TACAS (1), Lecture Notes in Computer Science, vol. 12078, pp. 287–305. Springer (2020)
20. Cubuktepe, M., Jansen, N., Junges, S., Katoen, J.P., Papusha, I., Poonawala, H.A., Topcu, U.: Sequential Convex Programming for the Efficient Verification of Parametric MDPs. In: TACAS (2), LNCS, vol. 10206, pp. 133–150 (2017)
21. Cubuktepe, M., Jansen, N., Junges, S., Katoen, J.P., Topcu, U.: Synthesis in pMDPs: A tale of 1001 parameters. In: ATVA, LNCS, vol. 11138, pp. 160–176. Springer (2018)
22. D'Argenio, P.R., Jeannet, B., Jensen, H.E., Larsen, K.G.: Reachability analysis of probabilistic systems by successive refinements. In: PAPM-PROBMIV, LNCS, vol. 2165, pp. 39–56. Springer (2001)
23. Daws, C.: Symbolic and Parametric Model Checking of Discrete-Time Markov chains. In: ICTAC, LNCS, vol. 3407, pp. 280–294. Springer (2004)
24. Dehnert, C., Junges, S., Jansen, N., Corzilius, F., Volk, M., Bruintjes, H., Katoen, J., Ábrahám, E.: PROPhESY: A PRObabilistic ParamEter SYnthesis Tool. In: CAV (1), LNCS, vol. 9206, pp. 214–231. Springer (2015)
25. Dehnert, C., Junges, S., Katoen, J., Volk, M.: A Storm is Coming: A Modern Probabilistic Model Checker. In: CAV (2), LNCS, vol. 10427, pp. 592–600. Springer (2017)
26. Delahaye, B., Larsen, K.G., Legay, A., Pedersen, M.L., Wasowski, A.: Decision problems for interval Markov chains. In: LATA, LNCS, vol. 6638, pp. 274–285. Springer (2011)
27. Gainer, P., Hahn, E.M., Schewe, S.: Incremental verification of parametric and reconfigurable Markov chains. In: QEST, LNCS, vol. 11024, pp. 140–156. Springer (2018)
28. Garatti, S., Campi, M.C.: The risk of making decisions from data through the lens of the scenario approach. IFAC-PapersOnLine **54**(7), 607–612 (2021)
29. Givan, R., Leach, S., Dean, T.: Bounded-parameter Markov decision processes. Artif. Intell. **122**(1–2), 71–109 (2000)
30. Hahn, E.M., Hermanns, H., Zhang, L.: Probabilistic reachability for parametric Markov models. STTT **13**(1), 3–19 (2010)
31. Han, J., Jonker, P.: A system architecture solution for unreliable nanoelectronic devices. IEEE Trans. Nanotechnol. **1**, 201–208 (2002)
32. Hatefi, H., Hermanns, H.: Model checking algorithms for Markov automata. ECEASST **53**, 88890 (2012)
33. Haussler, D.: Probably approximately correct learning. In: AAAI, pp. 1101–1108. AAAI Press (1990)
34. Helmink, L., Sellink, M., Vaandrager, F.: Proof-Checking a Data Link Protocol. In: TYPES, LNCS, vol. 806, pp. 127–165. Springer (1994)
35. Ho, C.P., Petrik, M.: Fast Bellman Updates for Robust MDPs. In: ICML (2018)
36. Junges, S., Ábrahám, E., Hensel, C., Jansen, N., Katoen, J., Quatmann, T., Volk, M.: Parameter Synthesis for Markov Models. CoRR arXiv:1903.07993 (2019)
37. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: Verification of Probabilistic Real-Time Systems. In: CAV, LNCS, vol. 6806, pp. 585–591. Springer (2011)
38. Kwiatkowska, M., Norman, G., Parker, D.: The PRISM Benchmark Suite. In: QEST, pp. 203–204. IEEE CS (2012)
39. Llerena, Y.R.S., Böhme, M., Brünink, M., Su, G., Rosenblum, D.S.: Verifying the Long-run Behavior of Probabilistic System Models

in the Presence of Uncertainty. In: ESEC/SIGSOFT FSE, pp. 587–597. ACM (2018)
40. McAllister, R., Peynot, T., Fitch, R., Sukkarieh, S.: Motion Planning and Stochastic Control with Experimental Validation on a Planetary Rover. In: IROS, pp. 4716–4723. IEEE (2012)
41. Meedeniya, I., Moser, I., Aleti, A., Grunske, L.: Evaluating probabilistic models with uncertain model parameters. Softw. Syst. Model. **13**(4), 1395–1415 (2014)
42. Nilim, A., El Ghaoui, L.: Robust control of Markov decision processes with uncertain transition matrices. Oper. Res. **53**(5), 780–798 (2005)
43. Papaefthymiou, G., Klöckl, B.: MCMC for wind power simulation. IEEE Trans. Energy Convers. **23**(1), 234–240 (2008)
44. Pineau, J., Gordon, G., Thrun, S.: Point-Based Value Iteration: an Anytime Algorithm for POMDPs. In: IJCAI, pp. 1025–1030 (2003)
45. Pnueli, A.: The Temporal Logic of Programs. In: FOCS, pp. 46–57 (1977)
46. Polgreen, E., Wijesuriya, V.B., Haesaert, S., Abate, A.: Data-efficient bayesian verification of parametric markov chains. In: QEST, Lecture Notes in Computer Science, vol. 9826, pp. 35–51. Springer (2016)
47. Puggelli, A., Li, W., Sangiovanni-Vincentelli, A.L., Seshia, S.A.: Polynomial-Time Verification of PCTL Properties of MDPs with Convex Uncertainties. In: CAV, pp. 527–542. Springer (2013)
48. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley, London (2014)
49. Quatmann, T., Dehnert, C., Jansen, N., Junges, S., Katoen, J.P.: Parameter Synthesis for Markov Models: Faster Than Ever. In: ATVA, LNCS, vol. 9938, pp. 50–67 (2016)
50. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach, 4th edn. Pearson, New York (2020)
51. Scheftelowitsch, D., Buchholz, P., Hashemi, V., Hermanns, H.: Multi-Objective Approaches to Markov Decision Processes with Uncertain Transition Parameters. In: VALUETOOLS, pp. 44–51 (2017)
52. Shmatikov, V.: Probabilistic analysis of an anonymity system. J. Comput. Secur. **12**(3–4), 355–377 (2004)
53. Steimle, L.N., Kaufman, D.L., Denton, B.T.: Multi-Model Markov Decision Processes. Optimization Online (2018)
54. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, London (2018)
55. Wiesemann, W., Kuhn, D., Rustem, B.: Robust Markov decision processes. Math. Oper. Res. **38**(1), 153–183 (2013)
56. Winkler, T., Junges, S., Pérez, G.A., Katoen, J.P.: On the complexity of reachability in parametric Markov decision processes. In: CONCUR, LIPIcs, vol. 140, pp. 14:1–14:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2019)
57. Wolff, E.M., Topcu, U., Murray, R.M.: Robust control of uncertain Markov decision processes with temporal logic specifications. In: CDC, pp. 3372–3379. IEEE (2012)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.