CTL Model Checking of Markov Decision Processes over the Distribution Space

Yulong Gao[1], Duc-An Nguyen[2], Karl H. Johansson[1], and Alessandro Abate[2]

[1] Division of Decision and Control Systems, KTH Royal Institute of Technology, and Digital Futures, Stockholm, Sweden yulongg@kth.se,kallej@kth.se
[2] Department of Computer Science, University of Oxford, UK
an.nguyen@cs.ox.ac.uk,alessandro.abate@cs.ox.ac.uk

Abstract. This work studies CTL model checking for finite-state Markov decision processes (MDPs) over the space of the MDP distributions. Instead of investigating the behavior of paths and properties over states of the MDP, as expressed with PCTL formulae, the focus of this work is on the associated transition system that is induced by the MDP dynamics over (transient) distributions. CTL logic is thus used to specify properties over the space of distributions, which provides an alternative way to express probabilistic specifications or requirements over the given MDP. We discuss the distinctive semantics of CTL formulae over the distribution space, compared to traditional PCTL specifications, and argue that these two alternatives are different, yet related. We then propose reachability-based CTL model checking algorithms over the distribution space, as well as computationally tractable, sample-based procedures for computing the relevant reachable sets: it is in particular shown that, with these procedures, the satisfaction set of any CTL specification in positive normal form can be soundly under-approximated by the union of convex polytopes. Examples and a case study elucidate the new approach.

Keywords: Markov decision processes · Computation tree logic · Reachability analysis · Transient probability distributions

1 Introduction

Probabilistic model checking is a technique for formally verifying properties of stochastic models [12, 13, 15]. It provides a framework for calculating the likelihood of the occurrence of specifications given the probabilistic behaviour of a stochastic model. This technique is of great interest in disparate and diverse applications, such as biological systems [10] and wireless networks [16], and for power management problems [18]. Markov decision processes (MDPs) are amongst the most popular models to formalize decision making under uncertainty.

In this work, we study use of computation tree logic (CTL) to express specifications for finite-state MDPs over their distribution space, and we correspondingly develop new CTL model checking algorithms. Unlike the standard,

existing body of work on model checking MDP over PCTL specifications, which concern requirements over states and trajectories of the MDP, we investigate the behavior of transient state distributions, which is governed by a discrete-time transition system evolving over the space of continuous state distributions. This transition system is endowed with non-determinism, depending on the actions chosen over the MDP, which makes it natural to use CTL formulae to specify properties and/or requirements over transient state distributions. We show that the distribution-specified CTL formulae are semantically different from the traditional probabilistic computation tree logic (PCTL) formulae, and thus can be employed to encode alternative probabilistic specifications. More specifically, the CTL framework proposed in this work can not only express similar quantitative temporal logic specifications over the state space through marginalisations over state distributions, but also encode other interesting requirements, e.g., probabilistic safety, with different semantics than PCTL formulae. In this work we thus extend known results on connections between qualitative PCTL formulae and related CTL specifications on models with non-determinism [2].

The characterisation of Sat sets of CTL specifications over continuous distributions space is attained by a new algorithm that computes backward reachable sets for both both existential and universal quantifiers. Furthermore, in order to practically and scalably compute such Sat sets, we put forward sample-based algorithms: specifically, we show that the satisfaction set of any CTL formula in positive normal form can be under-approximated by the union of convex polytopes.

This paper is organized as follows. Section 2 provides preliminaries on MDP models and (P)CTL logics, and formalises the problems to be investigated. Section 3 compares the semantics of distribution-specified CTL specifications with PCTL formulae over MDPs. Section 4 presents a solution to the CTL model checking problem based on reachability over the space of distributions, as well as related sample-based computational procedures. Section 5 discusses a case study to explain the proposed algorithms. Conclusions are drawn in Section 6.

Related Work - CTL [4] is a temporal logic based on a branching notion of time, rather than the linear notion of time used in linear temporal logic (LTL). A major difference with LTL is that CTL formulae allow the expression of properties in existential and universal sense, thus accounting for non-determinism in the model of interest. The CTL model checking boils down the recursive computation of satisfaction sets of sub-formulae, which relies on the backward reachability analysis [5]. The literature has led to a few CTL model checkers, such as EMC [4] and SMV [17]. In this work, we consider the CTL model checking over a continuous distribution space, which restricts the usability of existing model checkers. Instead, we design an approximate - yet sound - algorithm leveraging the sample-based computation of backward distributional reachable sets.

There is a large body of literature on probabilistic model checking, see for example [13] (for discrete-time Markov chain) and [7] (for MDP). Here,

we restrict our attention to the relevant literature on PCTL model checking for MDP. PCTL was first proposed in [9] as an extension of CTL. PCTL model checking for MDP is usually reducible to the computation of maximal or minimal probabilities for constrained reachability properties (e.g., next, bounded or unbounded until) [2]. This can be solved via recursive equations and value iteration for finite-horizon problems, or this relies on the solution to systems of linear equations for infinite-horizon problems [7]. Notable software tools for probabilistic model checking are PRISM [14] and Storm [6].

In [11], linear distribution temporal logic was introduced to integrate the specifications over the hidden states of partial observable MDPs (POMDPs) and the specifications over the belief of the hidden states. Recently, a barrier function-based approach was developed for synthesis over POMDP under linear distribution temporal logic specifications.

2 Preliminaries and Problem Formulation

In this section, we provide preliminaries on MDP models and on (P)CTL logic, and then present two problems studied in this work.

2.1 Models - Markov Decision Processes (MDP)

Definition 1. *A finite MDP is a tuple* $\mathsf{M} = (\mathbb{X}, \mathbb{U}, T, \mathcal{AP}_s, L_s)$*, where*

- $\mathbb{X}$ *is a finite state space with cardinality* $|\mathbb{X}| = n$*;*
- $\mathbb{U}$ *is a finite action space with cardinality* $|\mathbb{U}| = m$*;*
- $T : \mathbb{X} \times \mathbb{X} \times \mathbb{U} \to \mathbb{R}$ *is a transition probability, i.e,* $T(y|x,u)$ *assigns a probability from the state* $x \in \mathbb{X}$ *and the action* $u \in \mathbb{U}$ *to the state* $y \in \mathbb{X}$*;*
- *a finite set* $\mathcal{AP}_s$ *of atomic propositions;*
- *a labelling function* $L_s : \mathbb{X} \to 2^{\mathcal{AP}_s}$*.*

For each $x \in \mathbb{X}$, $\mathbb{U}_x \subseteq \mathbb{U}$ is a nonempty set consisting of the admissible actions when the state of the MDP is x. For any $x \in \mathbb{X}$ and $u \in \mathbb{U}_x$, $\sum_{y \in \mathbb{X}} T(y|x,u) = 1$. Let us denote by $\mathcal{P}(\mathbb{X})$ the set of state distributions, which is a simplex in $\mathbb{R}^n$. A state distribution $\pi \in \mathcal{P}(\mathbb{X})$ can be seen as a non-negative row vector $\pi \in \mathbb{R}^n$, such that $\pi \mathbf{1} = 1$, where $\mathbf{1}$ is a column vector with all elements being 1.

Definition 2 (State-Action Path). *For the MDP* M*, an infinite* state-action path *starting from* $x_0 \in \mathbb{X}$ *is a sequence of states* $\boldsymbol{x} = x_0 u_0 x_1 u_1 \ldots x_k u_k x_{k+1} u_{k+1} \ldots$*, with* $T(x_{k+1}|x_k, u_k) > 0$*. Denote by* $\mathsf{SPath}(x_0)$ *the set of all the state-action paths starting from* x_0*.*

In this paper, two kinds of policies are considered.

Definition 3 (Policies). *A policy is a map* $\mu : \mathbb{X} \to \mathcal{P}(\mathbb{U})$*, i.e., for each* $x \in \mathbb{X}$*,* $\sum_{u \in \mathbb{U}_x} \mu(u|x) = 1$*., where* $\mathcal{P}(\mathbb{U})$ *is the set of distributions over* $\mathbb{U}$*. Denote by* $\mathcal{U}$ *this set of randomised policies.*
A deterministic policy is a map $\mu^d : \mathbb{X} \to \mathbb{U}$*, i.e., for each* $x \in \mathbb{X}$*,* $\mu^d(x)$ *selects precisely one* u *from* $\mathbb{U}_x$*. Denote by* $\mathcal{U}^d$ *the set of deterministic policies.*

Note that, for an MDP M, the number of allowable deterministic policies is at most m^n, thus the set $\mathcal{U}^d$ is finite. It follows that a general policy $\mu \in \mathcal{U}$ can be interpreted as a distribution over $\mathcal{U}^d$.

Any policy $\mu \in \mathcal{U}$ (and, in particular, any deterministic policy $\mu^d \in \mathcal{U}^d$) induces from T a row-stochastic matrix as:

$$P^{\mu}(y|x) = \sum_{u \in \mathbb{U}_x} T(y|x,u)\mu(u|x). \tag{1}$$

Given an initial state distribution $\pi_0 \in \mathcal{P}(\mathbb{X})$ and a sequence of time-dependent policies $\{\mu_k \in \mathcal{U}\}_{k\in\mathbb{N}}$, the state distribution evolves over the Euclidean space $\mathbb{R}^n$ as

$$\pi_{k+1} = \pi_k P^{\mu_k} = \sum_{x \in \mathbb{X}} \pi_k(x) P^{\mu_k}(y|x). \tag{2}$$

As much as the MDP dynamics over its states (namely, its state-action paths) are governed by the transition probability matrix, the state distribution of an MDP follows the controlled dynamics in (2). We can thus work with the following MDP-induced transition system over distribution space.

Definition 4. *Given the MDP* $\mathsf{M} = (\mathbb{X}, \mathbb{U}, T, \mathcal{AP}_s, L_s)$*, the MDP-induced transition system* MTS *is a tuple* $\mathsf{MTS} = (\mathcal{P}(\mathbb{X}), \mathcal{U}, \to, \mathcal{AP}_d, L_d)$*, consisting of*

- *the space* $\mathcal{P}(\mathbb{X})$ *of distributions over states, a subset of* $\mathbb{R}^n$*;*
- *the set* $\mathcal{U}$ *of policies for* M*;*
- *the transition relation* $\to \in \mathcal{P}(\mathbb{X}) \times \mathcal{U} \times \mathcal{P}(\mathbb{X})$*, i.e., for* $\pi, \pi' \in \mathcal{P}(\mathbb{X})$ *and* $\mu \in \mathcal{U}$*,* $\pi \xrightarrow{\mu} \pi'$ *if and only if* $\pi' = \pi P^{\mu}$*;*
- *a finite set* $\mathcal{AP}_d$ *of atomic propositions;*
- *a labeling function* $L_d : \mathcal{P}(\mathbb{X}) \to 2^{\mathcal{AP}_d}$*.*

In other words, the model MTS is an uncountable-state dynamical system evolving over a subset of $\mathbb{R}^n$ (the unit simplex), with dynamics that are governed by the difference equation (2). The labeling function L_d and the associated atomic proposition set $\mathcal{AP}_d$ are in general different from L_s and $\mathcal{AP}_s$ in the MDP M. In Section 3, we show that if L_d and $\mathcal{AP}_d$ are appropriately defined based on L_s and $\mathcal{AP}_s$, we can express related properties for M and MTS.

Definition 5 (Distribution-Policy Path). *For the MDP-induced transition system* MTS*, an infinite* distribution-policy path $\boldsymbol{\pi}$ *starting from* $\pi_0 \in \mathcal{P}(\mathbb{X})$ *is a sequence of state distributions* $\boldsymbol{\pi} = \pi_0\mu_0\pi_1\mu_1 \ldots \pi_k\mu_k \ldots$ *such that* $\forall k \in \mathbb{N}$*,* $\pi_k \xrightarrow{\mu_k} \pi_{k+1}$*. Denote by* $\mathsf{DPath}(\pi_0)$ *the set of infinite distribution-policy paths starting from* π_0*.*

2.2 Specifications - Probabilistic Computation Tree Logic

CTL has a two-stage syntax, namely state and path formulae, defined over a general alphabet $\mathcal{AP}$, and encompasses both propositional and temporal logic operators. CTL *state formulae* are formed according to the following grammar:

$$\Phi ::= \text{true} \mid a \mid \neg\Phi \mid \Phi_1 \wedge \Phi_2 \mid \exists\varphi \mid \forall\varphi,$$

where $a \in \mathcal{AP}$ is an atomic proposition and φ is a path formula. CTL *path formulae* are instead shaped according to the following grammar:

$$\varphi ::= \bigcirc \Phi \mid \Phi_1 \mathsf{U} \Phi_2,$$

where $\bigcirc$ and U denote the "next" and "until" operators, respectively, and Φ, Φ_1, and Φ_2 are state formulae.

A PCTL formula is also defined over a finite alphabet $\mathcal{AP}$: its state formulae Φ are much like CTL, whereas path formulae that in CTL are quantified either existentially or universally (that is, $\exists\varphi$ and $\forall\varphi$) now depend on a probabilistic operator, as

$$\Phi ::= [\ldots] \mid \mathsf{Pr}_{\sim p}(\varphi),$$

where φ is a path formula, but now Pr denotes the probabilistic operator, $\sim \in \{>, <, \geq, \leq\}$, and $p \in [0, 1]$. PCTL path formulae φ are defined similarly as CTL, with the addition of the following operator:

$$\varphi ::= [\ldots] \mid \Phi_1 \mathsf{U}^{\leq k} \Phi_2,$$

where now $k \in \mathbb{N}$ is a finite index. The bounded until operator $\mathsf{U}^{\leq k}$, as natively used in PCTL formulae [2], can be naturally expressed also for CTL formulae by evaluating their semantics over finite paths [21] (see below).

We now taylor the (P)CTL syntax above to the models of interest, namely the MDP M and the MDP-induced transition system MTS, by discussing the associated satisfaction semantics [2]. Notice in particular that the generic alphabet $\mathcal{AP}$ will be tailored to the labelling maps introduced in the definitions of the models above (respectively $\mathcal{AP}_s$ and $\mathcal{AP}_d$), and that the semantics will hinge on state-action paths for M and on distribution-policy paths for MTS, respectively.

Definition 6 (PCTL Semantics). *Consider the MDP* M*. Given an atomic proposition* $a \in \mathcal{AP}_s$*, a state* $x \in \mathbb{X}$*, PCTL state formulae* Φ*,* Φ_1*, and* Φ_2*, and a PCTL path formula* φ*, the satisfaction relation* $\vDash$ *is defined for state formulae as follows:*

$$\begin{aligned}
&x \vDash a \Leftrightarrow a \in L_s(x),\\
&x \vDash \neg\Phi \Leftrightarrow x \nvDash \Phi,\\
&x \vDash \Phi_1 \wedge \Phi_2 \Leftrightarrow x \vDash \Phi_1 \wedge x \vDash \Phi_2,\\
&x \vDash \mathsf{Pr}_{\sim p}(\varphi) \Leftrightarrow \mathsf{Pr}\big(\boldsymbol{x} \in \mathsf{SPath}(x) \mid \boldsymbol{x} \vDash \varphi\big) \sim p.
\end{aligned}$$

Given a state-action path $\boldsymbol{x} = x_0 u_0 x_1 u_1 \ldots x_k u_k \ldots$*, the satisfaction relation* $\vDash$ *is defined for path formulae by*

$$\begin{aligned}
&\boldsymbol{x} \vDash \bigcirc\Phi \Leftrightarrow x_1 \in \Phi,\\
&\boldsymbol{x} \vDash \Phi_1 \mathsf{U}^{\leq k} \Phi_2 \Leftrightarrow \exists j \in \mathbb{N}_{[0,k]} \text{ s.t. } \begin{cases} x_j \vDash \Phi_2, \\ \forall i \in \mathbb{N}_{[0,j-1]}, x_i \vDash \Phi_1, \end{cases}\\
&\boldsymbol{x} \vDash \Phi_1 \mathsf{U} \Phi_2 \Leftrightarrow \exists j \in \mathbb{N} \text{ s.t. } \begin{cases} x_j \vDash \Phi_2, \\ \forall i \in \mathbb{N}_{[0,j-1]}, x_i \vDash \Phi_1. \end{cases}
\end{aligned}$$

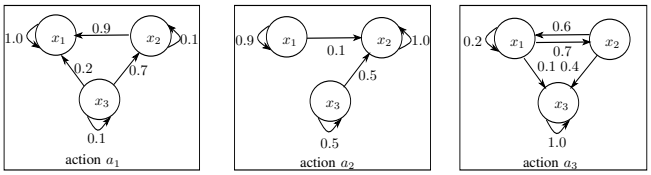

Fig. 1: Graphical representation of the MDP in Example 1.

CTL formulae are instead now taylored to MTS as follows.

Definition 7 (CTL semantics). *Consider the MDP-induced transition system* MTS*, a state distribution $\pi \in \mathcal{P}(\mathbb{X})$, an atomic proposition $a \in \mathcal{AP}_d$, CTL state formulae Φ, Φ_1, and Φ_2, and a CTL path formula φ. The satisfaction relation $\vDash$ is defined for CTL state formulae in the same manner over their propositional fragment, whereas*

$$\pi \vDash \exists\varphi \Leftrightarrow \boldsymbol{\pi} \vDash \varphi \text{ for some } \boldsymbol{\pi} \in \mathsf{DPath}(\pi),$$
$$\pi \vDash \forall\varphi \Leftrightarrow \boldsymbol{\pi} \vDash \varphi \text{ for all } \boldsymbol{\pi} \in \mathsf{DPath}(\pi).$$

Given a distribution-policy path $\boldsymbol{\pi} = \pi_0\mu_0\pi_1\mu_1 \ldots \pi_k\mu_k \ldots$, the satisfaction relation $\vDash$ is defined for path formulae by

$$\boldsymbol{\pi} \vDash \bigcirc\Phi \Leftrightarrow \pi_1 \in \Phi,$$
$$\boldsymbol{\pi} \vDash \Phi_1 \mathsf{U} \Phi_2 \Leftrightarrow \exists j \in \mathbb{N} \text{ s.t. } \begin{cases} \pi_j \vDash \Phi_2, \\ \forall i \in \mathbb{N}_{[0,j-1]}, \pi_i \vDash \Phi_1. \end{cases}$$

We have commented that the bounded-until operator can be naturally expressed for CTL similarly [21]. The following example discusses two CTL formulae specified over the distribution space.

Example 1. Consider the MDP with $\mathbb{X} = \{x_1, x_2, x_3\}, \mathbb{U} = \{a_1, a_2, a_3\}$, and the corresponding transition probability matrices displayed in Fig. 1. In order to define the MDP-induced transition system MTS, let us introduce two sets of state distributions, as shown in blue and red in Fig. 2. The set in red is $\{\pi \in \mathcal{P}(\mathbb{X}) \mid \|\pi - [1/3\ 1/3\ 1/3]\|_\infty \leq 0.1\}$ and the set in blue is $\{\pi \in \mathcal{P}(\mathbb{X}) \mid \|\pi - [0.1\ 0.2\ 0.7]\|_\infty \leq 0.05\}$. The label function L_d maps the distributions in the blue region to the atomic proposition a and the distributions in the red region to b. Let us consider two distribution-specified CTL formulae as $\Phi_1 = \exists(\neg a \mathsf{U} b)$ and $\Phi_2 = \forall(\bigcirc\neg a)$, which will be used throughout this work. □

Remark 1. The choice of CTL, a temporal logic with branching semantics, is motivated by the presence of non-determinism in the dynamics of the transition system MTS. It should be clear that the verification question for existentially-quantified formulae elicits a synthesis one, as we would be interested in generating the policy associated to satisfying traces. We shall see that the CTL model checking algorithms we put forward do produce such policy. □

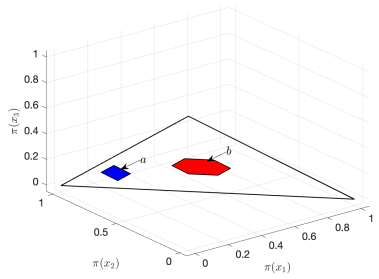

Fig. 2: Two set of state distributions (in blue and red), with labels a and b respectively, for the MDP of Example 1.

2.3 Statement of Problems under Study

The two perspectives on the MDP M and on the transition model MTS allow to introduce two alternative classes of probabilistic temporal requirements: one is by the widely-used PCTL logic, while another is with CTL formulae defined over state distributions. In order to emphasise their differences and to highlight the usefulness of introducing distribution-specified CTL formulae, we posit the following first problem.

Problem 1. Given the MDP M and the MDP-induced transition system MTS, formally relate the semantics of CTL formulae for MTS with that of PCTL specifications for M.

The second goal is to solve the CTL model checking problem for MTS.

Problem 2. Given an MDP M with initial distribution $\pi_0 \in \mathcal{P}(\mathbb{X})$ and the MDP-induced transition system MTS, and a CTL formula Φ, verify whether $\pi_0 \models \Phi$.

Note that, as CTL formulae in this work are defined over a continuous space of distributions, the standard CTL model checking algorithms for finite-state transition systems ought to be re-developed for this setup. It is remarkable that the control literature has not developed algorithms for these models.

3 Problem 1 - Comparison of Distribution-specified CTL Formulae with State-based PCTL Formulae

In this section, we compare the semantics of specific CTL formulae over the MDP distribution space with that of PCTL ones over its state space.

Table 1: Connections between specific CTL and PCTL formulae for MDPs

	PCTL formula Φ_{PCTL}	CTL formula Φ_{CTL}	Condition	Connection
(1)	$\mathsf{Pr}_{\sim p}(\bigcirc a_s)$	$\forall(\bigcirc a_d)$	$\mathbb{X}_d = \Pi^{\sim p}(\mathbb{X}_s)$	$x_0 \vDash \Phi_{\text{PCTL}}$ if and only if $e_{x_0} \vDash \Phi_{\text{CTL}}$[1]
(2)	$\mathsf{Pr}_{\geq p}(a_s \vee \bigcirc a_s)$	$\forall(a_d \vee \bigcirc a_d)$	$\mathbb{X}_d = \Pi^{\geq p}(\mathbb{X}_s)$	$x_0 \vDash \Phi_{\text{PCTL}}$ if and only if $e_{x_0} \vDash \Phi_{\text{CTL}}$
(3)	$\mathsf{Pr}_{\geq p}(a_{s_1} \mathsf{U} a_{s_2})$	$\forall(a_{d_1} \mathsf{U} a_{d_2})$	$\mathbb{X}_{d_1} = \Pi^{=1}(\mathbb{X}_{s_1})$ $\mathbb{X}_{d_2} = \Pi^{\geq p}(\mathbb{X}_{s_2})$	$x_0 \vDash \Phi_{\text{PCTL}}$ if $e_{x_0} \vDash \Phi_{\text{CTL}}$
(4)	$\mathsf{Pr}_{\geq p}(a_{s_1} \mathsf{U}^{\leq k} a_{s_2})$	$\forall(a_{d_1} \mathsf{U}^{\leq k} a_{d_2})$	$\mathbb{X}_{d_1} = \Pi^{=1}(\mathbb{X}_{s_1})$ $\mathbb{X}_{d_2} = \Pi^{\geq p}(\mathbb{X}_{s_2})$	$x_0 \vDash \Phi_{\text{PCTL}}$ if $e_{x_0} \vDash \Phi_{\text{CTL}}$
(5)	$\mathsf{Pr}_{=1}(\Diamond a_s)$	$\forall(\Diamond a_d)$	$\mathbb{X}_d = \Pi^{=1}(\mathbb{X}_s)$	$x_0 \vDash \Phi_{\text{PCTL}}$ if $e_{x_0} \vDash \Phi_{\text{CTL}}$
(6)	$\mathsf{Pr}_{\geq p}(\Box a_s)$	$\forall(\Box a_d)$	$\mathbb{X}_d = \Pi^{\geq p}(\mathbb{X}_s)$	$e_{x_0} \vDash \Phi_{\text{CTL}}$ if $x_0 \vDash \Phi_{\text{PCTL}}$
(7)	$\mathsf{Pr}_{=1}(\Box a_s)$	$\forall(\Box a_d)$	$\mathbb{X}_d = \Pi^{=1}(\mathbb{X}_s)$	$x_0 \vDash \Phi_{\text{PCTL}}$ if and only if $e_{x_0} \vDash \Phi_{\text{CTL}}$

[1] e_{x_0} is a vector with the x_0-th element equal to 1 and all the others set to 0.

We first remark that, given the different semantics of CTL and PCTL formulae, as provided in Section 2, these two sets of formulae are in general not comparable. In particular, there exist distribution-specified CTL formulae for which no corresponding PCTL formulae exist. For example, assume that there exists an atomic proposition a_d such that $L_d^{-1}(a_d)$ is *a subset of the interior* of the distribution space $\mathcal{P}(\mathbb{X})$. The meaningful CTL formulae $\exists \bigcirc a_d$, $\forall \bigcirc a_d$, $\exists \Diamond a_d$, and $\forall \Diamond a_d$ are such that any satisfying distribution in $L_d^{-1}(a_d)$ has a domain corresponding to the whole state space $\mathbb{X}$, for which a corresponding PCTL requirement is vacuous.

Still, despite their differences in syntax, we observe that there can be connections between distribution-specified CTL formulae and standard PCTL specifications. Consider the MDP $\mathsf{M} = (\mathbb{X}, \mathbb{U}, T, \mathcal{AP}_s, L_s)$ and the MDP-induced transition system $\mathsf{MTS} = (\mathcal{P}(\mathbb{X}), \mathcal{U}, \rightarrow, \mathcal{AP}_d, L_d)$. Let $\mathcal{AP}_s = \{a_s, a_{s1}, a_{s2}\}$, $\mathbb{X}_s = L_s^{-1}(a_s)$, $\mathbb{X}_{s1} = L_s^{-1}(a_{s1})$, and $\mathbb{X}_{s2} = L_s^{-1}(a_{s2})$. Similarly, let $\mathcal{AP}_d = \{a_d, a_{d1}, a_{d2}\}$, $\mathbb{X}_d = L_d^{-1}(a_d)$, $\mathbb{X}_{d1} = L_d^{-1}(a_{d1})$, and $\mathbb{X}_{d2} = L_d^{-1}(a_{d2})$. For a set $\mathbb{Y} \subseteq \mathbb{X}$, define $\Pi^{\sim p}(\mathbb{Y}) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid \sum_{x \in \mathbb{Y}} \pi(x) \sim p\}$, where $\sim \in \{>, <, \geq, \leq, =\}$ and $p \in [0, 1]$ is a given probability level. Table 1 summarises connections between specific CTL and PCTL formulae, emphasising required connections between the sets $\mathbb{X}_s$, $\mathbb{X}_{s1}$, and $\mathbb{X}_{s2}$ for PCTL, and $\mathbb{X}_d$, $\mathbb{X}_{d1}$, and $\mathbb{X}_{d2}$ for CTL, respectively. Let us discuss these connections in detail next.

For (1), $x_0 \vDash \mathsf{Pr}_{\sim p}(\bigcirc a_s)$ if and only if $\sum_{y\in\mathbb{X}_s} P^{\mu}(y|x_0) \sim p$ for all $\mu \in \mathcal{U}$, which can further rewritten as $e_{x_0}P^{\mu} \in \mathbb{X}_d \triangleq \Pi^{\sim p}(\mathbb{X}_s)$, for all $\mu \in \mathcal{U}$. Thus, $x_0 \vDash \mathsf{Pr}_{\sim p}(\bigcirc a_s)$ if and only if $e_{x_0} \vDash \forall(\bigcirc a_d)$. Similar arguments apply to (2).

For (3), $e_{x_0} \vDash \forall(a_{d_1} \mathsf{U} a_{d_2})$ implies that for any distribution-policy path $\boldsymbol{\pi} = e_{x_0}\mu_0\pi_1\mu_1 \ldots \pi_k\mu_k \ldots$, there exists $j \in \mathbb{N}$ such that $\pi_j \in \mathbb{X}_{d_2}$ and for all $i \in \mathbb{N}_{[0,j-1]}$, $\pi_i \in \mathbb{X}_{d_1}$. For $\boldsymbol{\pi} = e_{x_0}\mu_0\pi_1\mu_1 \ldots \pi_k\mu_k \ldots$, let $\widehat{\mathsf{SPath}}(\boldsymbol{\pi}) = \{\boldsymbol{x} \in \mathsf{SPath}(x_0) \mid \boldsymbol{x} = x_0u_0x_1u_1 \ldots x_ku_k \ldots, \mu_k(u_k|x_k) > 0, P^{\mu_k}(x_{k+1}|x_k) > 0, \forall k \in \mathbb{N}\}$. Given the definitions of $\mathbb{X}_{d_1}$ and $\mathbb{X}_{d_2}$, we have that $\mathsf{Pr}\big(\boldsymbol{x} \in \mathsf{SPath}(x_0)|\boldsymbol{x} \vDash \varphi\big) \geq \mathsf{Pr}\big(\boldsymbol{x} \in \widehat{\mathsf{SPath}}(\boldsymbol{\pi})|\boldsymbol{\pi} \in \mathsf{DPath}(e_{x_0}), e_{x_0} \vDash \forall(a_{d_1}\mathsf{U}a_{d_2})\big) \geq p$. Thus, $x_0 \vDash \mathsf{Pr}_{\geq p}(a_{s_1}\mathsf{U}a_{s_2})$ if $e_{x_0} \vDash \forall(a_{d_1}\mathsf{U}a_{d_2})$.

From (3), similar arguments apply to the *bounded until* formula in (4) and to the *eventually operator* in (5). It should be highlighted that in (5), we show that there exists a distribution-specified CTL formula that corresponds to the qualitative PCTL formula $\mathsf{Pr}_{=1}(\Diamond a_s)$. However, it is known [2] that there exists no state-based CTL formula that is equivalent to this PCTL formula.

Different from the cases in (1)–(5), where CTL formulae give sufficient (or necessary and sufficient) conditions to verify corresponding PCTL formulae, we show that in (6) the CTL-specified probabilistic safety is weaker than the PCTL-specified probabilistic safety. Recall $\mathsf{Pr}_{\geq p}(\Box a_s) = \mathsf{Pr}_{\leq p}(\Diamond\neg a_s)$. It has been shown in [20, 8] that this property holds for some state x_0 and $p > 0$ only if the set $\mathbb{X}_s$ contains an absorbing set, which raises a strong assumption on $\mathbb{X}_s$. Instead, we can use the CTL formula $\forall(\Diamond a_d)$ to express a different probabilistic safety requirement. Since $\mathbb{X}_d = \Pi^{\geq p}(\mathbb{X}_s)$, this CTL formula specifies the property of staying in the set $\mathbb{X}_s$ with probability no less than p at each time step, which is a weaker property than $\mathsf{Pr}_{\geq p}(\Box a_s)$. Thus, $e_{x_0} \vDash \Phi_{\mathrm{CTL}}$ if $x_0 \vDash \Phi_{\mathrm{PCTL}}$. Then, when it comes to a qualitative property, i.e., $p = 1$, we have $e_{x_0} \vDash \Phi_{\mathrm{CTL}}$ if and only if $x_0 \vDash \Phi_{\mathrm{PCTL}}$, as shown in (7) of Table 1.

Finally, we highlight that CTL model checking over the distribution space returns an informative satisfaction set that is a subset of the distribution space, unlike PCTL model checking, which results in a satisfaction set that is a subset of the state space. Different from the PCTL model checking algorithm, which boils down to the evaluation of the minimal or maximal probability of satisfying a formula, the new CTL model checking algorithm in this paper leverages reachability analysis over distribution space, as detailed in the next section.

4 Problem 2 - CTL Model Checking on Distribution Space

This section will provide a reachability-based solution to Problem 2, i.e., a characterisation of CTL model checking for MDPs over their distribution space. We begin by defining two backward-reachability operators with respect to existential and universal quantifiers, respectively. We then adapt the standard CTL model checking algorithm based on reachability analysis for finite transition systems to one for MDP-induced transition systems, which are instead endowed with a continuous state space. In view of this key feature, we finally provide

an approximate method for facilitating the computations associated to the new algorithm.

In this section, we focus our attention to the CTL formulae expressed in positive norm form (PNF), for which negations can only occur to basic atomic propositions. This assumption, which is clearly not restrictive, will facilitate the computation of under-approximations of the satisfaction sets required to verify general CTL formulae. It is well known [2] that each CTL formula can be equivalently expressed as one in PNF. The syntax of CTL state formulae in PNF is given by $\Phi ::= \text{true} \mid \text{false} \mid a \mid \neg a \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \exists\varphi \mid \forall\varphi$, while the CTL path formulae in PNF is defined by $\varphi ::= \bigcirc\Phi \mid \Phi_1 \mathsf{U} \Phi_2 \mid \Phi_1 \mathsf{W} \Phi_2$, where W denotes the "weak-until" operator. The semantics of W is the following: given a distribution-policy path $\boldsymbol{\pi} = \pi_0\mu_0 \ldots \pi_k\mu_k \ldots$, we say that $\boldsymbol{\pi} \vDash \Phi_1 \mathsf{W} \Phi_2$ if $\boldsymbol{\pi} \vDash \Phi_1 \mathsf{U} \Phi_2$ or $\boldsymbol{\pi} \vDash \Box\Phi_1$, where $\Box$ denotes the "always" operator, whereby $\boldsymbol{\pi} \vDash \Box\Phi$ if $\forall j \in \mathbb{N}, \pi_j \vDash \Phi$.

4.1 Reachability-based CTL Model Checking over Distributions

Consider the MDP $\mathsf{M} = \{\mathbb{X}, \mathbb{U}, T, \mathcal{AP}_s, L_s\}$ and the MDP-induced transition system $\mathsf{MTS} = (\mathcal{P}(\mathbb{X}), \mathcal{U}, \to, \mathcal{AP}_d, L_d)$. Define the set-valued map $\mathcal{BR}_\exists : 2^{\mathcal{P}(\mathbb{X})} \to 2^{\mathcal{P}(\mathbb{X})}$ as

$$\mathcal{BR}_\exists(\Pi) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid \ \exists\mu \in \mathcal{U}, \pi P^\mu \in \Pi\}, \tag{3}$$

as well as the set-valued map $\mathcal{BR}_\forall : 2^{\mathcal{P}(\mathbb{X})} \to 2^{\mathcal{P}(\mathbb{X})}$ as

$$\mathcal{BR}_\forall(\Pi) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid \ \forall\mu \in \mathcal{U}, \pi P^\mu \in \Pi\}, \tag{4}$$

where $\Pi \subseteq \mathcal{P}(\mathbb{X})$. The set $\mathcal{BR}_\exists(\Pi)$ collects all the state distributions which can be steered to the set Π under some policy $\mu \in \mathcal{U}$, while the set $\mathcal{BR}_\forall(\Pi)$ collects all the state distributions which can be steered to the set Π under all policies $\mu \in \mathcal{U}$.

Let us introduce the Post Set $\mathsf{Post}(\pi)$, comprising the direct successors of $\pi \in \mathcal{P}(\mathbb{X})$: this is defined by $\mathsf{Post}(\pi) = \{\pi' \in \mathbb{S} \mid \exists\mu \in \mathcal{U}, \pi \xrightarrow{\mu} \pi'\}$. The above two maps $\mathcal{BR}_\exists$ and $\mathcal{BR}_\forall$ can then be rewritten, respectively, as

$$\mathcal{BR}_\exists(\Pi) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid \ \mathsf{Post}(\pi) \cap \Pi \neq \emptyset\},$$
$$\mathcal{BR}_\forall(\Pi) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid \ \mathsf{Post}(\pi) \subseteq \Pi\}.$$

Based on the two maps defined above, and leveraging the standard CTL model checking algorithms for finite-state models [2], we introduce a new CTL model checking for MDPs over their distributions space.

Let us denote by $\mathsf{Sat}(\Phi) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid \pi \vDash \Phi\}$ the satisfaction set of any CTL state formula Φ. Consider an atomic proposition $a \in \mathcal{AP}_d$ and three CTL state formulae Φ, Φ_1, and Φ_2 in PNF. For the propositional fragment of CTL formulae, it is straightforward to see that:

- $\mathsf{Sat}(\text{true}) = \mathcal{P}(\mathbb{X})$ and $\mathsf{Sat}(\text{false}) = \emptyset$;

- $\mathsf{Sat}(a) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid a \in L_d(\pi)\}$ and $\mathsf{Sat}(\neg a) = \mathcal{P}(\mathbb{X}) \setminus \mathsf{Sat}(a)$;
- $\mathsf{Sat}(\Phi_1 \wedge \Phi_2) = \mathsf{Sat}(\Phi_1) \cap \mathsf{Sat}(\Phi_2)$ and $\mathsf{Sat}(\Phi_1 \vee \Phi_2) = \mathsf{Sat}(\Phi_1) \cup \mathsf{Sat}(\Phi_2)$.

Furthermore, from the one-step operators introduced above, it follows that

- $\mathsf{Sat}(\exists \bigcirc \Phi) = \mathcal{BR}_\exists(\mathsf{Sat}(\Phi))$ and $\mathsf{Sat}(\forall \bigcirc \Phi) = \mathcal{BR}_\forall(\mathsf{Sat}(\Phi))$.

Finally, the following proposition characterises how to compute the satisfaction sets for the until and weak-until operators in CTL.

Proposition 1. *The following statements hold:*

$$\mathsf{Sat}\left(\exists\left(\Phi_1 \mathsf{U} \Phi_2\right)\right) = \mathbb{T}_\infty, \quad \mathsf{Sat}\left(\exists\left(\Phi_1 \mathsf{W} \Phi_2\right)\right) = \mathbb{T}_\infty \cup \mathbb{P}_\infty;$$
$$\mathsf{Sat}\left(\forall\left(\Phi_1 \mathsf{U} \Phi_2\right)\right) = \mathbb{S}_\infty, \quad \mathsf{Sat}\left(\forall\left(\Phi_1 \mathsf{W} \Phi_2\right)\right) = \mathbb{S}_\infty \cup \mathbb{Q}_\infty;$$

where

$$\mathbb{T}_\infty = \lim_{k\to\infty} \mathbb{T}_i = \mathrm{cl}(\bigcup_{i\in\mathbb{N}} T_i), \mathbb{T}_{i+1} = \mathbb{T}_i \cup (\mathsf{Sat}(\Phi_1) \cap \mathcal{BR}_\exists(\mathbb{T}_i)) \text{ with } \mathbb{T}_0 = \mathsf{Sat}(\Phi_2);$$
$$\mathbb{S}_\infty = \lim_{k\to\infty} \mathbb{S}_i = \mathrm{cl}(\bigcup_{i\in\mathbb{N}} S_i), \mathbb{S}_{i+1} = \mathbb{S}_i \cup (\mathsf{Sat}(\Phi_1) \cap \mathcal{BR}_\forall(\mathbb{S}_i)) \text{ with } \mathbb{S}_0 = \mathsf{Sat}(\Phi_2);$$
$$\mathbb{P}_\infty = \lim_{k\to\infty} \mathbb{P}_i = \bigcap_{i\in\mathbb{N}} \mathrm{cl}(\mathbb{P}_i), \mathbb{P}_{i+1} = \mathbb{P}_i \cap \mathcal{BR}_\exists(\mathbb{P}_i) \text{ with } \mathbb{P}_0 = \mathsf{Sat}(\Phi_1);$$
$$\mathbb{Q}_\infty = \lim_{k\to\infty} \mathbb{Q}_i = \bigcap_{i\in\mathbb{N}} \mathrm{cl}(\mathbb{Q}_i), \mathbb{Q}_{i+1} = \mathbb{Q}_i \cap \mathcal{BR}_\forall(\mathbb{Q}_i) \text{ with } \mathbb{Q}_0 = \mathsf{Sat}(\Phi_1).$$

Proof. See Appendix B. □

To summarise, CTL model checking over distribution space reduces to the computation of backward reachable sets $\mathcal{BR}_\exists(\Pi)$ and $\mathcal{BR}_\forall(\Pi)$ from a given set $\Pi \subset \mathcal{P}(\mathbb{X})$. Let us emphasise that, for general sets Π (in particular, non-convex), it can be difficult to manipulate $\mathcal{BR}_\exists(\Pi)$ and $\mathcal{BR}_\forall(\Pi)$. Next, we will develop a sample-based method for facilitating this numerical computation.

4.2 A Sample-based Approximate Method to Compute Polytopic Backward Reachable Sets

We restrict our attention to the problem of computing backward reachable sets $\mathcal{BR}_\exists(\Pi)$ and $\mathcal{BR}_\forall(\Pi)$ whenever $\Pi \subseteq \mathcal{P}(\mathbb{X})$ is a convex polytope. Let us recall that a convex polytope $\mathbb{Y} \subset \mathbb{R}^n$ can be expressed in a (vertex) V-representation, i.e., $\mathbb{Y} = \mathrm{conv}(\{v_1, \ldots, v_N\})$; or alternatively in a (face, or half-space) H-representation, namely $\mathbb{Y} = \{z \in \mathbb{R}^n \mid Ax \leq b\}$, where $v_i \in \mathbb{R}^n$, $i = 1, \ldots, N$, $N \in \mathbb{N}$, $A \in \mathbb{R}^{l\times n}$, $b \in \mathbb{R}^l$, and l is the number of half-spaces.

Focussing on an MDP model M and the associated MTS, the following result provides a different way to represent $\mathcal{BR}_\exists(\Pi)$ and $\mathcal{BR}_\forall(\Pi)$ if $\Pi \subseteq \mathcal{P}(\mathbb{X})$ is asumed to be a convex polytope.

Proposition 2. *If $\Pi \subseteq \mathcal{P}(\mathbb{X})$ is a convex polytope, then*

– $\mathcal{BR}_{\exists}(\Pi)$ *is a convex polytope and can be rewritten as*

$$\mathcal{BR}_{\exists}(\Pi) = \left\{ (Q\mathbf{1})^T \in \mathcal{P}(\mathbb{X}) \;\middle|\; \begin{array}{l} Q \in \mathbb{R}^{n\times m},\ Q \geq 0,\ (Q\mathbf{1})^T \in \mathcal{P}(\mathbb{X}), \\ \pi \in \Pi,\ \forall y \in \mathbb{X},\ \pi(y) = \\ \sum\limits_{x\in\mathbb{X}} \sum\limits_{u\in\mathbb{U}_x} T(y|x,u)Q(x,u) \end{array} \right\}. \quad (5)$$

– $\mathcal{BR}_{\forall}(\Pi)$ *is a convex polytope and can be rewritten as*

$$\mathcal{BR}_{\forall}(\Pi) = \left\{ \pi \in \mathcal{P}(\mathbb{X}) \;\middle|\; \forall \mu^d \in \mathcal{U}^d, \pi P^{\mu^d} \in \Pi \right\}. \quad (6)$$

Proof. See Appendix A. □

Remark 2. The matrix Q to define the existential backward reachable set in (6) is called the occupation measure in the literature, which enables to recover a policy $\mu \in \bar{\mathcal{U}}$ by

$$\mu(u|x) = \begin{cases} \frac{Q(x,u)}{\sum_{v\in\mathbb{U}_x} Q(x,v)} & \text{if } \sum\limits_{v\in\mathbb{U}_x} Q(x,v) > 0, \\ \frac{1}{|\mathbb{U}_x|}, & \text{if } \sum\limits_{v\in\mathbb{U}_x} Q(x,v) = 0 \ \&\ u \in \mathbb{U}_x. \end{cases}$$

The use of occupation measures allows to reformulate a constrained MDP problem as a linear program [1], which solution can be used to recover a sequence of (time-dependent) policies for a finite-horizon problem (cf. case study). □

From Proposition 2 we can observe that, even if the set Π is a polytope in the form of either V-representation or H-representation, it can still be quite challenging computationally to exactly compute the polytopic sets $\mathcal{BR}_{\exists}(\Pi)$ and $\mathcal{BR}_{\forall}(\Pi)$, particularly whenever the MDP M has a large number n of states. The main reason is that both $\mathcal{BR}_{\exists}(\Pi)$ and $\mathcal{BR}_{\forall}(\Pi)$ depend on extra variables (e.g., the matrix Q in (5)), and that the necessary set projection in high-dimensional spaces can be computationally heavy. In the following we discuss Algorithm 1, a scalable, sample-based method to under-approximate sets $\mathcal{BR}_{\exists}(\Pi)$ and $\mathcal{BR}_{\forall}(\Pi)$.

The input to Algorithm 1 consists of two convex polytopes Π and Γ with $\Pi \subseteq \mathcal{P}(\mathbb{X}) \subset \text{int}(\Gamma)$, and the number of samples $N_s \in \mathbb{N}_{\geq 1}$. In line 1, we select uniformly at random samples $\{\pi_i^s\}_{i=1}^{N_s}$ in $\mathbb{R}^n$ from Γ. Then, these samples are used to generate samples $\pi_i^{1s} \in \mathcal{BR}_{\exists}(\Pi)$ and $\pi_i^{2s} \in \mathcal{BR}_{\forall}(\Pi)$ in line 3, by projecting π_i^s onto $\mathcal{BR}_{\exists}(\Pi)$ and $\mathcal{BR}_{\forall}(\Pi)$, respectively. The output of Algorithm 1 comprises the convex hulls of these projected samples, namely $\widehat{\mathcal{BR}}_{\exists}(\Pi, N_s)$ and $\widehat{\mathcal{BR}}_{\forall}(\Pi, N_s)$. The following proposition discusses two important properties of the sets output by the algorithm.

Proposition 3. *Consider two convex polytopes Π and Γ with $\Pi \subseteq \mathcal{P}(\mathbb{X}) \subset \mathsf{int}(\Gamma)$ and an integer $N_s \in \mathbb{N}_{\geq 1}$. Under Algorithm 1, the sets $\widehat{\mathcal{BR}}_{\exists}(\Pi, N_s)$ and $\widehat{\mathcal{BR}}_{\forall}(\Pi, N_s)$ are under approximations of $\mathcal{BR}_{\exists}(\Pi)$ and $\mathcal{BR}_{\forall}(\Pi)$, respectively, for all $N_s \in \mathbb{N}_{\geq 1}$.*

Proof. See Appendix A. □

Algorithm 1 Sample-based Backward Reachable Set Computation

Input: two convex polytopes Π and Γ with $\Pi \subseteq \mathcal{P}(\mathbb{X}) \subset \text{int}(\Gamma)$, $N_s \in \mathbb{N}_{\geq 1}$

Output: approximated backward reachable sets $\widehat{\mathcal{BR}}_\exists(\Pi, N_s)$ and $\widehat{\mathcal{BR}}_\forall(\Pi, N_s)$

1: select uniformly at random a group of samples $\{\pi_i^s\}_{i=1}^{N_s}$ from Γ;
2: **for** $i = 1 : N_s$ **do**
3: compute $\pi_i^{1s} = \text{argmin}_{\pi \in \mathcal{BR}_\exists(\Pi)} \|\pi - \pi_i^s\|^2$ and $\pi_i^{2s} = \text{argmin}_{\pi \in \mathcal{BR}_\forall(\Pi)} \|\pi - \pi_i^s\|^2$
4: **end for**
5: **return** $\widehat{\mathcal{BR}}_\exists(\Pi, N_s) = \text{conv}\big(\{\pi_i^{1s}, i \in \mathbb{N}_{[1,N_s]}\}\big)$ and $\widehat{\mathcal{BR}}_\forall(\Pi, N_s) = \text{conv}\big(\{\pi_i^{2s}, i \in \mathbb{N}_{[1,N_s]}\}\big)$.

The computational complexity of Algorithm 1 is linear with the number of samples and polynomial with the number of states n and with the number of actions m. Projecting each sample π_i^s onto $\mathcal{BR}_\exists(\Pi)$ and $\mathcal{BR}_\forall(\Pi)$ is a quadratic program with $n + nm$ decision variables. There exist standard algorithms, e.g., the interior point method [3], that can solve a quadratic program in polynomial time with respect to the number of decision variables.

4.3 Approximate CTL Model Checking over Distribution Space

Leveraging the sample-based computation of backward reachable sets, we are ready to design an approximate, yet sound, CTL model checking algorithm over the distribution space.

In the following, we will show that the satisfaction set of each CTL formula in PNF can be under-approximated by the union of a group of convex polytopes, under the following assumption. We remark that this assumption and the latter Lemma 3 echo why we focus on the CTL formulae in PNF: if the set $\mathsf{Sat}(\Phi)$ is under-approximated by unions of convex polytopes, such under-approximation will in general not hold for $\mathsf{Sat}(\neg\Phi)$, unless Φ is an atomic proposition.

Assumption 1 *For each atomic proposition $a \in \mathcal{AP}_d$, the set of distributions associated with the labeling function L_d, denoted by $L_d^{-1}(a) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid a \in L_d(\pi)\}$, is a convex polytope, later considered in its H-representation.*

Let us recall how to compute the set complement of a convex polytope.

Lemma 1. *Consider a convex polytope $\mathbb{Y}_0$ with $\mathbb{Y}_0 \subseteq \mathcal{P}(\mathbb{X})$. Suppose $\mathbb{Y}_0 = \{z \in \mathbb{R}^n \mid Ax \leq b\}$ with $A \in \mathbb{R}^{l \times n}$ and $b \in \mathbb{R}^l$. Let $\mathbb{Y}_i = \{z \in \mathcal{P}(\mathbb{X}) \mid [A]_i z \geq b_i + \epsilon_i\}$, $\forall i \in \mathbb{N}_{[1,l]}$, where $[A]_i$ and b_i denotes the i-th row of A and b, respectively, and ϵ_i is a small positive constant. Then, $\bigcup_{i=1}^l \mathbb{Y}_i \subset \mathcal{P}(\mathbb{X}) \setminus \mathbb{Y}_0$.*

The use of ϵ_i is to ensure the closure of the set $\mathbb{Y}_i$. Lemma 1 implies that, under Assumption 1, for each atomic proposition $a \in \mathcal{AP}_d$, the satisfaction set $\mathsf{Sat}(\neg a)$, i.e., the complement of $L_d^{-1}(a)$ with respect to $\mathcal{P}(\mathbb{X})$, can be under-approximated by the union of a set of convex polytopes.

The following lemma shows that the backward-reachable sets obtained from the union of a set of convex polytopes can be under-approximated by the union of the backward reachable sets obtained from the corresponding convex polytopes.

Lemma 2. *Consider a group of sets* $\{\Pi_i\}_{i=1}^M$*, where each* $\Pi_i \subseteq \mathcal{P}(\mathbb{X})$ *is a convex polytope. For any* $N_i \in \mathbb{N}_{\geq 1}$*,* $i \in \mathbb{N}_{[1,M]}$*,*

$$\cup_{i=1}^M \widehat{\mathcal{BR}}_\exists(\Pi_i, N_i) \subseteq \mathcal{BR}_\exists(\cup_{i=1}^M \Pi_i) \text{ and } \cup_{i=1}^M \widehat{\mathcal{BR}}_\forall(\Sigma_i, N_i) \subseteq \mathcal{BR}_\forall(\cup_{i=1}^M \Pi_i). \quad (7)$$

Proof. Directly follows from Proposition 3. □

The next lemma shows that the satisfaction sets obtained by applying propositional and temporal operators (and hence the sat sets of general CTL formulae in PNF) can be under-approximated by the union of convex polytopes.

Lemma 3. *Consider three CTL formulae* Φ*,* Φ_1 *and* Φ_2 *in PNF. Suppose that their satisfaction sets* $\mathsf{Sat}(\Phi)$*,* $\mathsf{Sat}(\Phi_1)$*,* $\mathsf{Sat}(\Phi_2)$ *are respectively under-approximated by unions of convex polytopes. Then, the sets* $\mathsf{Sat}(\Phi_1 \wedge \Phi_2)$*,* $\mathsf{Sat}(\Phi_1 \vee \Phi_2)$*,* $\mathsf{Sat}(\exists \bigcirc \Phi)$*,* $\mathsf{Sat}(\forall \bigcirc \Phi)$*,* $\mathsf{Sat}(\exists \Phi_1 \mathsf{U} \Phi_2)$*,* $\mathsf{Sat}(\forall \Phi_1 \mathsf{U} \Phi_2)$*,* $\mathsf{Sat}(\exists \Phi_1 \mathsf{W} \Phi_2)$*, and* $\mathsf{Sat}(\forall \Phi_1 \mathsf{W} \Phi_2)$ *can be also under-approximated by finite unions of convex polytopes.*

The following example shows satisfaction sets of the CTL formulae in Example 1.

Example 2. Let us recall the MDP and CTL formulae in Example 1. Consider the formula $\exists(\neg a \mathsf{U} b)$. Applying Algorithm 1 and Lemmata 1–3, we obtain the under-approximation of the satisfaction set $\mathsf{Sat}(\exists(\neg a \mathsf{U} b))$, which is the union of the four cyan sets shown in Fig. 4 of Appendix B. Similarly, the cyan state distribution set in Fig. 5 of Appendix B under-approximates the satisfaction set $\mathsf{Sat}(\forall(\bigcirc \neg a))$. □

Now we are ready to provide the solution to Problem 2, i.e., CTL model checking over the distribution space of MDPs.

Theorem 1. *Consider the MDP* M *and the MDP-induced transition system* MTS*. Suppose that Assumption 1 holds. Then there exists a sound and computationally tractable algorithm such that, for any CTL formula* Φ *in PNF, its satisfaction set* $\mathsf{Sat}(\Phi)$ *can be under-approximated by a finite union of convex polytopes.*

Proof. See Appendix A. □

5 Case Study

In this section, we will validate our CTL model checking algorithm through an unmanned aerial vehicle (UAV) path planning example.

As shown in 3, the UAV moves in a 5×5 grid world and has five possible actions {*up, down, left, right, stay*}. Due to environmental uncertainties (e.g., wind), we assume that the first four actions will drive the state to the desired next configuration with probability $1-\alpha$, and to to other neighboring states with likelihood $\frac{\alpha}{N_{neigh}}$, where N_{neigh} is the number of available/feasible neighboring states (not including the desired state). We say that a state (x_1, y_1) is a

neighboring state of a state (x_1, y_1) if $\max\{|x_1 - x_2|, |y_1 - y2|\} \leq 1$. The cyan regions are obstacles and we assume that the corresponding states are absorbing states, that is, these states are invariant under all actions. Denote by Obs the set of obstacle states. The red grid $(5, 5)$ is the target state, denoted by Target. The initial state is the blue square at $(1, 1)$. We consider the following path planning problem: to find a feasible policy such that, starting from the initial state, the UAV reaches the target set, whilst avoiding the obstacle states, with a desired probability (assumed to be given).

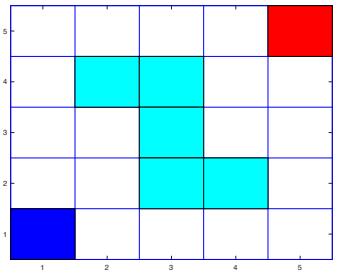

Fig. 3: Case Study - UAV Motion Planning

This problem can be studied by introducing an MDP model M. The state space X corresponds to the set comprising the 25 squares in the grid world, whereas the action space $\mathbb{U}$ is $\{$*up, down, left, right, stay*$\}$. The transition probability T is defined according to the transitions described above. Let $\mathcal{AP}_s = \{a_{\text{unsafe}}, a_{\text{target}}\}$ be the set of atomic proposition, and L_s a labeling function, such that $L_s^{-1}(a_{\text{unsafe}}) = \text{Obs}$ and $L_s^{-1}(a_{\text{target}}) = \{(5, 5)\}$, i.e. the target state. The motion planning problem is to find a policy such that the PCTL formula $\Phi_{\text{PCTL}} = \mathsf{Pr}_{\geq p}(\neg a_{\text{unsafe}} \mathsf{U} a_{\text{target}})$ is fulfilled, where $p \in (0, 1]$ is a desired (given) probability level.

Given the MDP M, we can formalise the corresponding MDP-induced transition system MTS. The distribution space $\mathcal{P}(\mathbb{X})$ is a subset of $\mathbb{R}^{25}$, and the policy set $\mathcal{U}$ is defined as in Definition 3. Let the set of atomic propositions be $\mathcal{AP}_d = \{b^{\beta}_{\text{unsafe}}, b^{p}_{\text{target}}\}$, and introduce a labeling function be L_d such that $L_d^{-1}(b^{\beta}_{\text{unsafe}}) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid \sum_{x \in \text{Obs}} \pi(x) \leq \beta\}$ and $L_d^{-1}(b^{p}_{\text{target}}) = \{\pi \in \mathcal{P}(\mathbb{X}) \mid \pi(\text{Target}) \geq p\}$, where $\beta \in [0, 1]$. That is, the sets $L_d^{-1}(b^{\beta}_{\text{unsafe}})$ and $L_d^{-1}(b^{p}_{\text{target}})$ are parameterized by β and p, respectively. Let us denote by π_0 the distribution associated with the deterministic initial state $\{(1, 1)\}$. The existence of a policy solving the motion planning problem can be asserted if π_0 satisfies the CTL formula $\Phi_{\text{CTL}} = \exists(\neg b^{\beta}_{\text{unsafe}} \mathsf{U} b^{p}_{\text{target}})$, and obtained according to Remark 2.

In the following we consider two scenarios. In the first one, the parameter α is set to be 0, that is, each action drives the current state to the specified direction with probability 1 - the model's dynamics are deterministic. The probability levels β and p are set to be 0 and 0.8, respectively. We remark that in this setting, as discussed in Table 1, if $\pi_0 = e_{(1,1)}$ fulfils the CTL formula Φ_{CTL}, then there exists a sequence of policies such that the initial state $(1, 1)$ satisfies the PCTL formula Φ_{PCTL}. Implementing the approximate CTL model checking algorithm, we find that the distribution π_0 falls within the (existential) backward-reachable set after 8 iterations (cf. Theorem 1). According to Remark 2, we can synthesize a sequence of policies such that starting from $(1, 1)$, the state reaches the target

state (5,5) while avoiding the obstacles within 8 steps, with probability no less than $p = 0.8$. Fig. 6 in Appendix B shows the evolution of the state distributions across this time horizon.

In the second scenario, the parameter α is set to be 0.05 so that there exist possible transitions to other neighboring states under the first four actions. If we still set β and p to be 0 and 0.8, the PCTL formula Φ_{PCTL} becomes infeasible, which entails (cf. Table 1) that the distribution π_0 does not satisfy the CTL formula Φ_{CTL}. Alternatively, we relax the parameter β from 0 to 0.15. This means that the UAV is required to stay in the safe region with probability greater than 0.85 at all times. Implementing the approximate CTL model checking algorithm, we find that π_0 is in the sound approximate satisfaction set (see result in Theorem 1). Thus, similarly to the above case, we can find a feasible realization of the transient state distribution, which is shown in Fig. 7 of Appendix B.

The case study has been run on an ARM system M1 chip on MacBook Pro 2021, with 16GB RAM. A set of 400 sample has been used for each independent run of the experiments, which took an average 10 seconds for the deterministic scenario, and 200 seconds for the noisy scenario. These result shows that the computational overhead is reasonable, as this case study the sat set were computed over a space with 25 dimensions.

6 Conclusions

We have introduced a model checking framework for finite MDPs over the space of their transient distributions. Focusing on the transition system that is induced by the MDP dynamics over its space of distributions, we have employed CTL logic to specify temporal properties. This provides an alternative way to express probabilistic specifications for the MDP. We have compared the semantics of CTL formulae over distribution space with traditional PCTL specifications, and showed that these two alternatives are different, yet related. We have proposed novel reachability-based CTL model checking algorithms over distributions space, as well as more tractable sample-based procedures for computing reachable sets: it is in particular shown that, with these procedures, the satisfaction set of any CTL specification in positive normal form can be soundly under-approximated by the union of convex polytopes. The CTL model checking algorithm for existentially quantified formulae additionally results in a policy such that the CTL formula is fulfilled.

In parallel with the CTL model checking problem over MDP distribution, another worthwhile goal is the policy synthesis for the distribution-specified LTL requirements. We are also interested in developing model checking approaches based on finite-state, non-stochastic abstractions of the MDP-induced transition system, and in framing them within the theory of (bi-)simulation relations between the abstract model and the concrete MTS.

References

1. Altman, E.: Constrained Markov Decision Processes: Stochastic Modeling. Routledge (1999)
2. Baier, C., Katoen, J.P.: Principles of Model Checking. MIT press (2008)
3. Boyd, S., Boyd, S.P., Vandenberghe, L.: Convex Optimization. Cambridge university press (2004)
4. Clarke, E.M., Emerson, E.A.: Design and synthesis of synchronization skeletons using branching time temporal logic. In: Workshop on Logic of Programs. pp. 52–71. Springer (1981)
5. Clarke, E.M., Emerson, E.A., Sistla, A.P.: Automatic verification of finite-state concurrent systems using temporal logic specifications. ACM Transactions on Programming Languages and Systems **8**(2), 244–263 (1986)
6. Dehnert, C., Junges, S., Katoen, J.P., Volk, M.: A storm is coming: A modern probabilistic model checker. In: International Conference on Computer Aided Verification. pp. 592–600. Springer (2017)
7. Forejt, V., Kwiatkowska, M., Norman, G., Parker, D.: Automated verification techniques for probabilistic systems. In: International School on Formal Methods for the Design of Computer, Communication and Software Systems. pp. 53–113. Springer (2011)
8. Gao, Y., Johansson, K.H., Xie, L.: Computing probabilistic controlled invariant sets. IEEE Transactions on Automatic Control **66**(7), 3138–3151 (2020)
9. Hansson, H., Jonsson, B.: A logic for reasoning about time and reliability. Formal Aspects of Computing **6**(5), 512–535 (1994)
10. Heath, J., Kwiatkowska, M., Norman, G., Parker, D., Tymchyshyn, O.: Probabilistic model checking of complex biological pathways. Theoretical Computer Science **391**(3), 239–257 (2008)
11. Jones, A., Schwager, M., Belta, C.: Distribution temporal logic: Combining correctness with quality of estimation. In: 52nd IEEE Conference on Decision and Control. pp. 4719–4724. IEEE (2013)
12. Katoen, J.P.: The probabilistic model checking landscape. In: Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science. pp. 31–45 (2016)
13. Kwiatkowska, M., Norman, G., Parker, D.: Stochastic model checking. In: International School on Formal Methods for the Design of Computer, Communication and Software Systems. pp. 220–270. Springer (2007)
14. Kwiatkowska, M., Norman, G., Parker, D.: Prism: Probabilistic model checking for performance and reliability analysis. ACM SIGMETRICS Performance Evaluation Review **36**(4), 40–45 (2009)
15. Kwiatkowska, M., Norman, G., Parker, D.: Probabilistic model checking: Advances and applications. In: Formal System Verification, pp. 73–121. Springer (2018)
16. Kwiatkowska, M., Norman, G., Sproston, J.: Probabilistic model checking of the ieee 802.11 wireless local area network protocol. In: Joint International Workshop von Process Algebra and Probabilistic Methods, Performance Modeling and Verification. pp. 169–187. Springer (2002)
17. McMillan, K.L.: Symbolic Model Checking. Springer (1993)
18. Norman, G., Parker, D., Kwiatkowska, M., Shukla, S., Gupta, R.: Using probabilistic model checking for dynamic power management. Formal Aspects of Computing **17**(2), 160–176 (2005)
19. Rockafellar, R.T., Wets, R.J.B.: Variational Analysis. Springer (2009)

20. Tkachev, I., Abate, A.: Characterization and computation of infinite-horizon specifications over markov processes. Theoretical Computer Science **515**, 1–18 (2014)
21. Vardi, M.Y., Stockmeyer, L.: Improved upper and lower bounds for modal logics of programs. In: ACM Symposium on Theory of Computing. pp. 240–251 (1985)

Appendix A

Proof of Proposition 1. Let us first consider formulae that are quantified existentially. We have that $\exists(\Phi_1 \mathsf{W} \Phi_2) = \exists(\Phi_1 \mathsf{U} \Phi_2) \vee \exists\Box\Phi_1$. It has been shown in [2] that the computation of $\mathsf{Sat}(\exists(\Phi_1 \mathsf{U} \Phi_2))$ and $\mathsf{Sat}(\exists\Box\Phi_1)$ leverages the iterative computation of the set $\mathbb{T}_i$ and $\mathbb{P}_i$, respectively. Note that the sequence $\{\mathbb{T}_i\}_{k\in\mathbb{N}}$ is non-decreasing, whereas the sequence $\{\mathbb{P}_i\}_{k\in\mathbb{N}}$ is non-increasing. Then, it follows from the convergence of monotone set sequences [19] that $\mathsf{Sat}(\exists(\Phi_1 \mathsf{U} \Phi_2)) = \mathbb{T}_\infty$ and $\mathsf{Sat}(\exists(\Phi_1 \mathsf{W} \Phi_2)) = \mathbb{T}_\infty \cup \mathbb{P}_\infty$.
A similar reasoning applies over universally quantified formulae, by replacing the existentially quantified backward-reachable set with the universally quantified backward-reachable set. □

Proof of Proposition 2. Let us first consider the expression of $\mathcal{BR}_\exists(\Pi)$. From (1), the distribution dynamics (2) can be rewritten in the following form:

$$\pi' = \sum_{x\in\mathbb{X}} \pi(x) P^\mu(y|x) = \sum_{x\in\mathbb{X}} \pi(x) \Big(\sum_{u\in\mathbb{U}_x} T(y|x,u)\mu(u|x) \Big)$$

for some $\mu \in \mathcal{U}$. It follows that the product of $\mu(u|x)$ and $\pi(x)$ can be replaced by a matrix $Q \in \mathbb{R}^{n\times m}$ with $Q \geq 0$ and $(Q\mathbf{1})^T \in \mathcal{P}(\mathbb{X})$. Thus, the set $\mathcal{BR}_\exists(\Pi)$ defined in (3) can be rewritten as (5). If $\Pi \subseteq \mathcal{P}(\mathbb{X})$ is a convex polytope, we have that $\mathcal{BR}_\exists(\Pi)$ in (5) involves a finite number of inequalities and thus it is also a convex polytope.

For the set $\mathcal{BR}_\forall(\Pi)$, recall the fact that each policy $\mu \in \mathcal{U}$ is a distribution over the set of deterministic policies $\mathcal{U}^d$, and that the set $\mathcal{U}^d$ has finite cardinality. Thus, $\mathcal{BR}_\forall(\Pi)$ can be can be rewritten as (6) and it is also a convex polytope if Π is a convex polytope. □

Proof of Lemma 2. The under approximation relation between $\widehat{\mathcal{BR}}_\exists(\Pi, N_s)$ and $\mathcal{BR}_\exists(\Pi)$ (or $\widehat{\mathcal{BR}}_\forall(\Pi, N_s)$ and $\mathcal{BR}_\forall(\Pi)$) directly follows from that the projected samples $\pi_i^{1s} \in \mathcal{BR}_\exists(\Pi)$ and $\pi_i^{2s} \in \mathcal{BR}_\forall(\Pi)$ (see line 3 in Algorithm 1). □

Proof of Lemma 3. Suppose $\cup_{i=1}^{M_\Phi} \Pi_i^\Phi \subseteq \mathsf{Sat}(\Phi)$, $\cup_{i=1}^{M_{\Phi_1}} \Pi_i^{\Phi_1} \subseteq \mathsf{Sat}(\Phi_1)$, $\cup_{i=1}^{M_{\Phi_2}} \Pi_i^{\Phi_2} \subseteq \mathsf{Sat}(\Phi_2)$, where Π_i^Φ, $\Pi_j^{\Phi_1}$, and $\Pi_k^{\Phi_2}$ are convex polytopes for all $i \in \mathbb{N}_{[1,M_\Phi]}$, $j \in \mathbb{N}_{[1,M_{\Phi_1}]}$, and $k \in \mathbb{N}_{[1,M_{\Phi_2}]}$.

Let us first consider $\mathsf{Sat}(\Phi_1 \wedge \Phi_2)$. Applying the distributive law of set operations, we obtain that

$$\mathsf{Sat}(\Phi_1 \wedge \Phi_2) \supseteq \left(\cup_{i=1}^{M_{\Phi_1}} \Pi_i^{\Phi_1}\right) \cap \left(\cup_{j=1}^{M_{\Phi_2}} \Pi_j^{\Phi_2}\right) = \cup_{\substack{i\in\mathbb{N}_{[1,M_{\Phi_1}]} \\ j\in\mathbb{N}_{[1,M_{\Phi_2}]}}} \left(\Pi_i^{\Phi_1} \cap \Pi_j^{\Phi_2}\right).$$

Note that the intersection of two convex polytopes is either a convex polytope or an empty set. Thus, $\mathsf{Sat}(\Phi_1 \wedge \Phi_2)$ can be under-approximated by the union of convex polytopes. For the disjunction operator, it is straightforward to see that $\mathsf{Sat}(\Phi_1 \vee \Phi_2) \supseteq \left(\cup_{i=1}^{M_{\Phi_1}} \Pi_i^{\Phi_1}\right) \cup \left(\cup_{j=1}^{M_{\Phi_2}} \Pi_j^{\Phi_2}\right)$.

Let us now consider basic formulae with temporal operators. For the next operator, it follows from Lemma 2 that $N_i \in \mathbb{N}_{\geq 1}$, $i \in \mathbb{N}_{[1,M]}$,

$$\mathsf{Sat}(\exists \bigcirc \Phi) \supseteq \mathcal{BR}_\exists(\cup_{i=1}^{M_\Phi} \Pi_i^\Phi) \supseteq \cup_{i=1}^{M_\Phi} \widehat{\mathcal{BR}}_\exists(\Pi_i^\Phi, N_i),$$
$$\mathsf{Sat}(\forall \bigcirc \Phi) \supseteq \mathcal{BR}_\forall(\cup_{i=1}^{M_\Phi} \Pi_i^\Phi) \supseteq \cup_{i=1}^{M_\Phi} \widehat{\mathcal{BR}}_\forall(\Pi_i^\Phi, N_i).$$

For the until operator, with reference to Proposition 1, let us define the set sequences for each set $\Pi_j^{\Phi_2}$, $j \in \mathbb{N}_{[1,M_{\Phi_2}]}$:

$$\hat{\mathbb{T}}_{i+1,j} = \left(\cup_{i=1}^{M_{\Phi_1}} \Pi_i^{\Phi_1}\right) \cap \widehat{\mathcal{BR}}_\exists(\hat{\mathbb{T}}_{i,j}, N_{i,j}) \text{ with } \hat{\mathbb{T}}_{0,j} = \Pi_j^{\Phi_2},$$
$$\hat{\mathbb{S}}_{i+1,j} = \left(\cup_{i=1}^{M_{\Phi_1}} \Pi_i^{\Phi_1}\right) \cap \widehat{\mathcal{BR}}_\forall(\hat{\mathbb{S}}_{i,j}, N_{i,j}) \text{ with } \hat{\mathbb{S}}_{0,j} = \Pi_j^{\Phi_2}.$$

where $N_{i,j} \in \mathbb{N}_{\geq 1}$. By the distributive law and following Lemma 2, we can recursively show that both $\hat{\mathbb{T}}_{i,j}$ and $\hat{\mathbb{S}}_{i,j}$ can be represented as the union of a finite number of convex polytopes. It follows from Proposition 1 that $\mathsf{Sat}(\exists \Phi_1 \mathsf{U} \Phi_2)$ and $\mathsf{Sat}(\forall \Phi_1 \mathsf{U} \Phi_2)$ can be, respectively, under-approximated by $\cup_{j=1}^{M_{\Phi_1}} \cup_{i=1}^{N_1} \hat{\mathbb{T}}_{i,j}$ and $\cup_{j=1}^{M_{\Phi_1}} \cup_{i=1}^{N_2} \hat{\mathbb{S}}_{i,j}$, for all $N_1, N_2 \in \mathbb{N}_{\geq 1}$, both of which are union of convex polytopes.

Finally, for the weak until operator, we need further define the following set sequences for each $\Pi_j^{\Phi_1}$, $j \in \mathbb{N}_{[1,M_{\Phi_1}]}$:

$$\hat{\mathbb{P}}_{i+1,j} = \hat{\mathbb{P}}_{i,j} \cap \widehat{\mathcal{BR}}_\exists(\hat{\mathbb{P}}_{i,j}, N_{i,j}), \text{ with } \hat{\mathbb{P}}_{0,j} = \Pi_j^{\Phi_1},$$
$$\hat{\mathbb{Q}}_{i+1,j} = \hat{\mathbb{Q}}_{i,j} \cap \widehat{\mathcal{BR}}_\forall(\hat{\mathbb{Q}}_{i,j}, N_{i,j}), \text{ with } \hat{\mathbb{Q}}_{0,j} = \Pi_j^{\Phi_1},$$

where $N_{i,j} \in \mathbb{N}_{\geq 1}$. Note that both $\hat{\mathbb{P}}_{i,j}$ and $\hat{\mathbb{Q}}_{i,j}$ are convex polytopes for all j. Let $\mathrm{Iter}_{\max} \in \mathbb{N}_{\geq 1}$ be the maximum iteration. If there exists $i \leq \mathrm{Iter}_{\max} - 1$ such that $\hat{\mathbb{P}}_{i+1,j} = \hat{\mathbb{P}}_{i,j}$, let $\hat{\mathbb{P}}_{\Box,j} = \hat{\mathbb{P}}_{i,j}$; otherwise, let $\hat{\mathbb{P}}_{\Box,j} = \emptyset$. We can define $\hat{\mathbb{Q}}_{\Box,j}$ similarly. Then, we have that $\mathsf{Sat}(\exists \Phi_1 \mathsf{W} \Phi_2)$ and $\mathsf{Sat}(\forall \Phi_1 \mathsf{W} \Phi_2)$ can be, respectively, under-approximated by $\left(\cup_{j=1}^{M_{\Phi_1}} \cup_{i=1}^{N_1} \hat{\mathbb{T}}_{i,j}\right) \cup \left(\cup_{j=1}^{M_{\Phi_1}} \hat{\mathbb{P}}_{\Box,j}\right)$ and $\left(\cup_{j=1}^{M_{\Phi_1}} \cup_{i=1}^{N_1} \hat{\mathbb{S}}_{i,j}\right) \cup \left(\cup_{j=1}^{M_{\Phi_1}} \hat{\mathbb{Q}}_{\Box,j}\right)$, for all $N_1, N_2 \in \mathbb{N}_{\geq 1}$, both of which are union of convex polytopes. This completes the proof. □

Proof of Theorem 1. Recall that the CTL model checking can be performed by a recursive procedure that calculates the satisfaction set for all subformulae

of Φ. If Φ is in PNF, it follows from Lemmata 1 and 3 that under Assumption 1, the satisfaction set of each subformula of Φ can be under-approximated by the finite union of convex polytopes. It follows from the proofs of Lemmata 1 and 3, the under-approximated set is computed by applying Algorithm 1, which is computationally tractable. Due to the under-approximation relation, we conclude that if an initial distribution π_0 belongs to this under-approximated set, we conclude that $\pi_0 \vDash \Phi$, which implies the soundness of the overall approach.

□

Appendix B

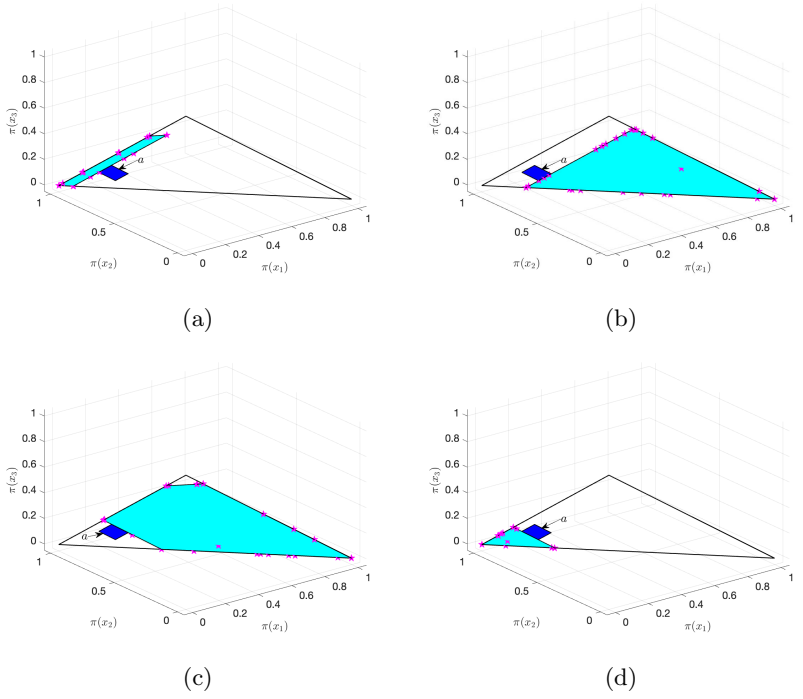

Fig. 4: Four distribution sets (cyan colour), whose union under-approximates the satisfaction set $\mathsf{Sat}(\exists(\neg a \mathsf{U} b))$, where a labels the blue polytope. The magenta stars are projections of the generated samples onto the satisfaction sets.

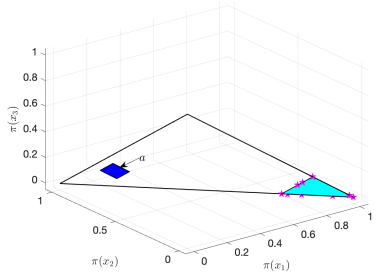

Fig. 5: The state distribution set in cyan that under-approximates the satisfaction set $\mathsf{Sat}(\forall(\bigcirc \neg a))$, where a labels the blue polytope. The magenta stars are projections of the generated samples onto the satisfaction set.

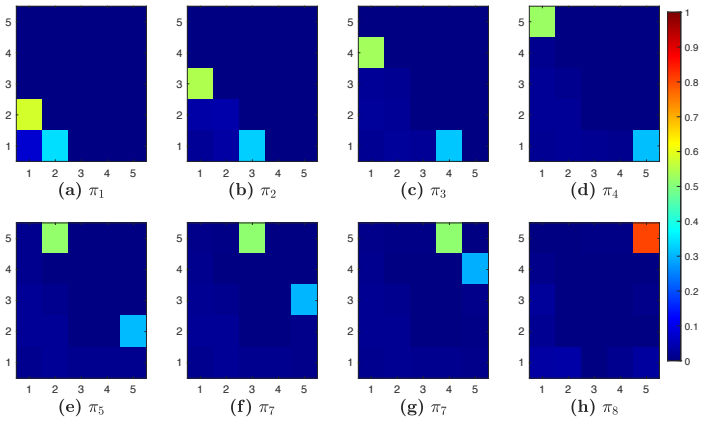

Fig. 6: MDP with deterministic transitions: evolution of state distribution that, initialised as $\pi_0 = e_{(1,1)}$ and under a feasible policy, reaches the target state $(5, 5)$ (within 8 steps) while avoiding the obstacles (as in Fig. 3) with overall probability no less than 0.8.

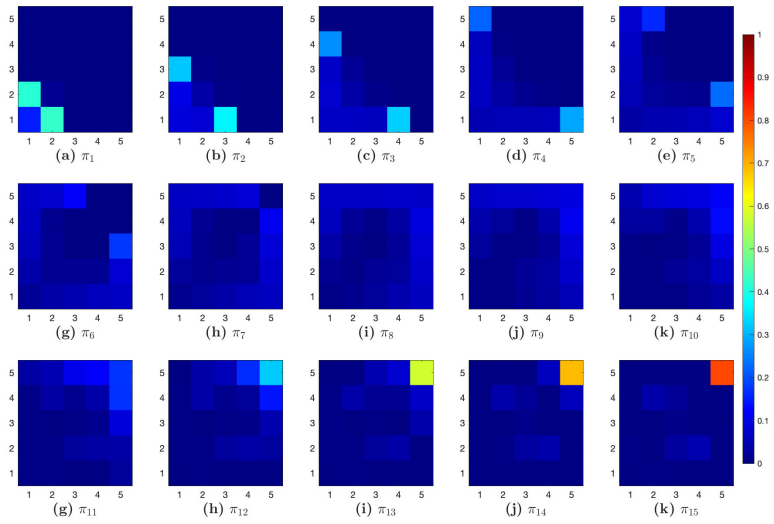

Fig. 7: MDP with noisy transitions: evolution of state distribution that, initialised as $\pi_0 = e_{(1,1)}$ and under a feasible policy, reaches the target state $(5, 5)$ with probability grater than 0.80, while possibly entering the obstacles (as in Fig. 3) with probability no greater than 0.15 at all times.