


Omega-Regular Verification and Control for Distributional Specifications in MDPs

S. Akshay 

Dept of CSE, Indian Institute of Technology Bombay, Mumbai, India

Ouldouz Neysari 

Singapore Management University, Singapore

University of Tehran, Iran

Đorđe Žikelić 

Singapore Management University, Singapore

Abstract

A classical approach to studying Markov decision processes (MDPs) is to view them as state transformers. However, MDPs can also be viewed as distribution transformers, where an MDP under a strategy generates a sequence of probability distributions over MDP states. This view arises in several applications, even as the probabilistic model checking problem becomes much harder compared to the classical state transformer counterpart. It is known that even distributional reachability and safety problems become computationally intractable (Skolem- and positivity-hard). To address this challenge, recent works focused on sound but possibly incomplete methods for verification and control of MDPs under the distributional view. However, existing automated methods are applicable only to distributional reachability, safety and reach-avoidance specifications.

In this work, we present the first automated method for verification and control of MDPs with respect to distributional omega-regular specifications. To achieve this, we propose a novel notion of distributional certificates, which are sound and complete proof rules for proving that an MDP under a distributionally memoryless strategy satisfies some distributional omega-regular specification. We then use our distributional certificates to design the first fully automated algorithms for verification and control of MDPs with respect to distributional omega-regular specifications. Our algorithms follow a template-based synthesis approach and provide soundness and relative completeness guarantees, while running in PSPACE. Our prototype implementation demonstrates practical applicability of our algorithms to challenging examples collected from the literature.

2012 ACM Subject Classification Theory of computation → Verification by model checking

Keywords and phrases MDPs, Distributional objectives, ω -regularity, Certificates

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2025.6

Funding This work was supported in part by the Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 1 grant (Project ID:22-SIS-SMU-100), Google Research Award 2023 and the SBI Foundation Hub for Data Science & Analytics, IIT Bombay.

1 Introduction

Markov decision processes (MDPs) are a standard model for reasoning and sequential decision making in the presence of uncertainty. The verification community has long studied MDPs as state transformers, where their semantics are interpreted over cylinder sets of paths (see e.g. [8]). As a result, quantitative verification questions focus on state-based properties, such as the eventual reachability of a state with maximum probability over all MDP strategies. There is a rich body of literature on efficient algorithms for reasoning about state-based properties in MDPs, including model checking over expressive logics such as PCTL* [30].

An orthogonal class of objectives, which forms our focus in this paper, considers properties that are defined over the space of probability distributions over MDP states, rather than the



© S. Akshay, Ouldouz Neysari, and Đorđe Žikelić;
licensed under Creative Commons License CC-BY 4.0
36th International Conference on Concurrency Theory (CONCUR 2025).
Editors: Patricia Bouyer and Jaco van de Pol; Article No. 6; pp. 6:1–6:19
Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

state space of the MDP. This allows one to reason about the movement of the probability mass, for instance, one can say that always in the future the probability mass is equally divided between two bi-stable states. Such objectives, that we call *distributional objectives*, are simpler to reason about under alternative semantics which view MDPs as *distribution transformers*. In this view, starting from some initial distribution over MDP states, the MDP under a strategy induces a sequence of distributions over MDP states, generating a new distribution at each time step. One can then specify properties with respect to this sequence of distributions, such as distributional reachability or safety. This view naturally arises in several applications, including multi-agent systems [9, 5] or biochemical reaction networks [29, 28]. However, it turns out that even the simplest distributional properties such as distributional reachability and safety cannot be expressed in PCTL* [10], rendering classical probabilistic model checking algorithms inapplicable to reasoning about distributional specifications. This means that reasoning about distributional properties in MDPs requires new methods. The past decade has seen a rich line of theoretical work on analyzing Markov chains and MDPs as distribution transformers [31, 29, 11, 6, 4, 5, 2, 26]. However, existing automated methods are restricted to distributional reachability, safety, or reach-avoidance specifications.

In this paper, we present the first automated method for strategy verification and synthesis in MDPs with respect to *distributional ω -regular specifications*, significantly extending the class of distributional objectives for which automated methods are available. In doing so, we focus on the verification problem for a given MDP strategy, as well as the control problem which asks to synthesize an MDP strategy which ensures that a distributional ω -regular specification is satisfied. Our strategy verification and synthesis methods are based on the novel notion of *distributional certificates* which we introduce in this work. Distributional certificates provide a sound and complete proof rule for proving that an MDP under a distributionally memoryless strategy satisfies the distributional ω -regular specification of interest. We restrict our attention to *distributionally memoryless strategies*, which make moves based on the current distribution rather than the state, and which are known to be sufficient for reasoning about distributional reachability and safety [4, 5]. Our distributional certificates build on the ideas from program verification and certificates such as ranking function [22], invariants [21] or Büchi ranking functions [17], and bring these ideas to the setting of reasoning about distributional ω -regular specifications in MDPs.

We then present our automated algorithms for strategy verification and synthesis in MDPs with respect to distributional ω -regular specifications. In the setting of distributional objectives, it is known that safety and reachability are already hard, or more precisely, positivity and Skolem-hard [3]. The decidability of both are long-standing open problems in linear dynamical systems [32]. As a result, rather than aiming for sound and complete algorithms that would inherently be computationally expensive/infeasible, we adopt a template-based synthesis approach and instead design algorithms that can more efficiently search for distributional certificates and distributionally memoryless strategies that can be expressed in *affine arithmetic*. By fixing symbolic affine templates for the certificate and the strategy and by using existing methods for solving quantified formulas over real arithmetic, one is able to reduce the strategy verification and synthesis problems to satisfiability checking in existential theory of the reals, therefore obtaining sound algorithms that run in PSPACE. Furthermore, our algorithms also provide relative completeness guarantees for computing an affine distributional certificate and a memoryless strategy, whenever they exist.

We implement our approach and consider standard benchmarks and examples from the literature, while focusing on several distributional ω -regular specifications for our evaluation. Our results show the practicality of the approach and the potential for future applications.

Related work. In addition to the work already mentioned, we discuss a (non-exhaustive list of) few others. Our distributional certificates draw insights from classical certificates for program verification and template-based synthesis algorithms for their computation. Notable examples include ranking functions for proving termination in programs [22, 13] and invariants for proving safety in programs [21, 14]. Our distributional certificates draw insights from Büchi ranking functions of [17] for proving LTL properties in programs. However, there are several important differences. First, we leverage and lift the idea of Büchi ranking to the setting of probability distributions over MDP states. Second, our distributional certificates and algorithms for their computation also need to reason about *strategies* in MDPs. This is reflected in the following key difference. Our certificates provide soundness and completeness guarantees for all distributional ω -regular specifications and distributionally memoryless strategies, and proving this requires reasoning about reachability under a strategy (see the proof of Theorem 9). The certificate of [17], on the other hand, is complete only for specifications that can be represented via *deterministic* Büchi automata, if the specification needs to be satisfied from a set of initial states (see Corollary 2 in [17]).

In the distributional setting, the probabilistic logics defined in [10, 11, 2] are all orthogonal to the classical semantics, and the model checking techniques developed are not template-based. To the best of our knowledge, none of these works have been automated. The works [4, 5] propose certificates for distributional reachability, safety and reach-avoidance and design template-based synthesis algorithms for their computation. Our paper follows this line of work and introduces distributional certificates and template-based algorithms for *distributional ω -regular specifications*, hence significantly generalizing the class of distributional properties that we can automatically reason about.

Certificates were also used for reasoning about infinite-state probabilistic models such as probabilistic programs under the state-based view. In particular, supermartingale certificates were proposed for qualitative reachability [12, 15], quantitative reachability, safety and reach-avoidance [33, 19, 34, 18, 35], and most recently for qualitative ω -regular specifications [1]. However, these certificates are not, a priori, applicable to the distributional setting.

2 Preliminaries

In this section, we recall the basics of probabilistic systems and Markov decision processes. A *probability distribution* over a countable set X is a map $\mu : X \rightarrow [0, 1]$ such that $\sum_{x \in X} \mu(x) = 1$. The *support* of X is defined via $\text{supp}(\mu) = \{x \in X \mid \mu(x) > 0\}$. We use $\Delta(X)$ to denote the set of all probability distributions over X .

MDPs. A *Markov decision process* (MDP) is a tuple $\mathcal{M} = (S, \text{Act}, P)$. We use S to denote a finite set of *states* and Act to denote a finite set of *actions*. Slightly overloading the notation, for each state $s \in S$, we write $\text{Act}(s) \subseteq \text{Act}$ to denote the set of *available actions* at s . Finally, $P : S \times \text{Act} \rightarrow \Delta(S)$ is a *transition function*, assigning to each state s and available action $a \in \text{Act}(s)$ a probability distribution over the successor states. When $|\text{Act}(s)| = 1$ for each state s , we say that \mathcal{M} is a *Markov chain*.

An *infinite path* in an MDP is a sequence $\rho = s_1, a_1, s_2, a_2, \dots \in (S \times \text{Act})^\omega$, such that $a_i \in \text{Act}(s_i)$ and $P(s_i, a_i)(s_{i+1}) > 0$ for all $i \in \mathbb{N}$. A *finite path* ϱ in an MDP is a finite prefix of an infinite path that ends in a state. We use ρ_i and ϱ_i to denote the i -th state along an (in)finite path. We use $\text{IPaths}_{\mathcal{M}}$ and $\text{FPaths}_{\mathcal{M}}$ to denote the sets of all infinite and finite paths in the MDP \mathcal{M} , respectively.

Semantics of MDPs. The semantics of MDPs are formalized in terms of strategies. A *strategy* (or *policy*) in an MDP \mathcal{M} is a function $\pi : \text{FPaths}_{\mathcal{M}} \rightarrow \Delta(\text{Act})$ which to each finite path (called a *history*) assigns a probability distribution over the action to be taken next. We require that, if a finite path $\varrho \in \text{FPaths}_{\mathcal{M}}$ ends in a state s , then $\text{supp}(\pi(\varrho)) \subseteq \text{Act}(s)$. A strategy is said to be *memoryless* if the probability distribution over actions depends only on the last state of the finite path and not on the whole history, i.e. if $\pi(\varrho) = \pi(\varrho')$ whenever ϱ and ϱ' end in the same state. For every initial state distribution $\mu_0 \in \Delta(S)$, an MDP \mathcal{M} and a strategy π together give rise to a probability space over the set of all infinite paths in the MDP [8]. We denote by $\mathbb{P}_{\mu_0}^{\pi}$ the probability measure and by $\mathbb{E}_{\mu_0}^{\pi}$ the expectation operators in this probability space, omitting the MDP \mathcal{M} from the notation when clear from the context.

MDPs as distribution transformers. MDPs are typically regarded as random generators of infinite paths, giving rise to a probability space over the set of all infinite paths in the MDP. Classical probabilistic model checking problems then explore the expected behaviour of these randomly generated infinite paths, giving rise to *path properties* [8]. However, one can also view MDPs as (*deterministic*) *transformers of distributions*.

Consider an MDP \mathcal{M} , a strategy π , and an initial state distribution $\mu_0 \in \Delta(S)$. For each $i \in \mathbb{N}$ and state s , define $\mu_i(s) = \mathbb{P}_{\mu_0}^{\pi}[\rho \in \text{IPaths}_{\mathcal{M}} \mid \rho_i = s]$, i.e. the probability that the i -th state of a randomly generated infinite path is s . We write $\mu_i = \mathcal{M}^{\pi}(\mu_0, i)$ for the induced probability distribution of the i -th state of a randomly generated infinite path. Hence, the MDP \mathcal{M} , a strategy π , and an initial state distribution $\mu_0 \in \Delta(S)$ together give rise to a sequence of probability distributions over the MDP states

$$\mu_0, \quad \mu_1 = \mathcal{M}^{\pi}(\mu_0, 1), \quad \mu_2 = \mathcal{M}^{\pi}(\mu_0, 2), \quad \mu_3 = \mathcal{M}^{\pi}(\mu_0, 3), \quad \dots$$

One can then study properties of this sequence of distributions. Some examples are *distributional reachability* and *distributional safety*, which ask if the induced sequence of distributions contains or does not contain an element of some specified set of distributions [4, 5].

ω -regular specifications. In this work we will consider ω -regular specifications, which subsume a broad class of specifications such as those belonging to linear temporal logic (LTL) or computation tree logic (CTL) [8]. An ω -regular specification over a finite set AP of atomic propositions is defined by a *non-deterministic Büchi automaton (NBA)* $N = (Q, \Sigma, \delta, q_0, F)$, where Q is a finite set of states, $\Sigma = 2^{\text{AP}}$ is a finite set of letters, $\delta : Q \times \Sigma \rightarrow 2^Q$ is a (non-deterministic) transition function, $q_0 \in Q$ is the initial state and $F \subseteq Q$ are accepting states. An infinite word of letters $\sigma_1, \sigma_2, \dots \in \Sigma^{\omega}$ is said to be *accepting*, if it gives rise to at least one accepting run in N , i.e. if there exists a run q_0, q_1, q_2, \dots such that $q_{i+1} \in \delta(q_i, \sigma_i)$ for each i and such that $q_i \in F$ for infinitely many i . Note that given an LTL formula φ , it can be converted to an equivalent NBA N^{φ} in exponential time (see e.g., [8]). In what follows, we will often write examples and benchmarks in LTL as it will be easier and often more intuitive. But for our analysis, we will convert them to their equivalent NBA and reason only about these NBA as the ω -regular specification.

Transition systems. In order to reason about the synchronous evolution of a sequence of distributions over the MDP states and a run in the NBA, we will later introduce the notion of product distributional transition systems in Section 4. Hence, we here recall the notion of transition systems, which are commonly used to model imperative numerical programs.

An (*infinite-state*) *transition system* is a tuple $\mathcal{T} = (L, V, l_{\text{init}}, \theta_{\text{init}}, \mapsto)$, where

- L is a finite set of locations,
- V is a finite set of real-valued variables,

- $l_{\text{init}} \in L$ is the initial location,
- $\theta_{\text{init}} \subseteq \mathbb{R}^{|V|}$ is the set of initial variable valuations, and
- \mapsto is a finite set of transitions of the form $\tau = (l_\tau, l'_\tau, G_\tau, U_\tau)$ with l_τ a source location, l'_τ a target location, G_τ a guard which is a boolean predicate over the variables in V , and $U_\tau : \mathbb{R}^n \rightarrow \mathbb{R}^n$ an update function.

A *state* in the transition system is a tuple $(l, x) \in L \times \mathbb{R}^{|V|}$ consisting of a location in L and a valuation of variables in V . A transition $\tau = (l_\tau, l'_\tau, G_\tau, U_\tau)$ is said to be *enabled* at a state (l, x) if $l = l_\tau$ and $x \models G_\tau$. An *infinite path* (or a *run*) in the transition system is a sequence of states $(l_0, x_0), (l_1, x_1), \dots$ with $l_0 = l_{\text{init}}$, $x_0 \in \text{Init}$, and where for each $i \in \mathbb{N}_0$ there exists a transition $\tau_i = (l_i, l_{i+1}, G_{\tau_i}, U_{\tau_i})$ enabled at (l_i, x_i) such that $x_{i+1} = U_{\tau_i}(x_i)$. A state (l, x) is said to be *reachable*, if there exists an infinite path that contains (l, x) .

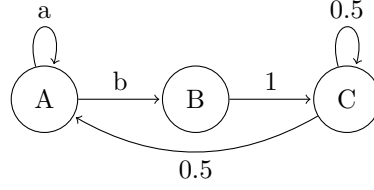
3 Problem Statement

We now formally define the problems that we consider in this work. Our goal is to design fully automated algorithms for formal verification and control in MDPs with respect to distributional ω -regular specifications. Hence, we first need to formalize the notion of distributional ω -regular specifications. In what follows, let $\mathcal{M} = (S, \text{Act}, P)$ be an MDP.

Distributional ω -regular specifications. Similarly to the classical ω -regular specification setting, we first need to specify a finite set of *atomic propositions* AP . We are interested in reasoning about a sequence of distributions induced by an MDP under a strategy. Hence, we let the set AP consist of finitely many logical formulas of the form $\exp(\mu(s_1), \dots, \mu(s_{|S|})) \geq 0$. Here, $\exp : \mathbb{R}^{|S|} \rightarrow \mathbb{R}$ is an arithmetic expression over the probabilities $\mu(s_1), \dots, \mu(s_{|S|})$ of being in each state of the MDP, where $s_1, \dots, s_{|S|}$ is an arbitrary (but throughout fixed) enumeration of MDP states. In practice, we let AP contain exactly those atomic propositions that appear in the property that we want to reason about. A *distributional ω -regular specification* φ is then defined by an NBA $N^\varphi = (Q, \Sigma, \delta, q_0, F)$ with $\Sigma = 2^{\text{AP}}$.

We now define the semantics of distributional ω -regular specifications. Consider a finite set of atomic propositions AP , a distributional ω -regular specification φ , a strategy π and an initial distribution $\mu_0 \in \Delta(S)$ in the MDP \mathcal{M} . The MDP \mathcal{M} under strategy π from the initial distribution μ_0 induces an infinite word $\sigma_0, \sigma_1, \sigma_2, \dots$ in the language 2^{AP} as follows. As defined in Section 2, an MDP \mathcal{M} under strategy π from the initial distribution μ_0 induces a sequence $\mu_0, \mu_1, \mu_2, \dots$ of distributions over MDP states. Then, for each $i \in \mathbb{N}_0$, we define the letter σ_i as the set of all atomic propositions in AP that are satisfied at the distribution μ_i , i.e. $\sigma_i = \{p \in \text{AP} \mid \mu_i \models p\}$, where we use \models to denote proposition satisfaction. We say that the MDP \mathcal{M} *satisfies* distributional ω -regular specification φ under strategy π from initial distribution $\mu_0 \in \Delta(S)$, if this infinite word $\sigma_0, \sigma_1, \sigma_2, \dots$ is accepted by the NBA N^φ .

Distributionally memoryless strategies. We restrict our attention to a class of strategies called distributionally memoryless strategies. A strategy $\pi : \text{FPaths}_{\mathcal{M}} \rightarrow \Delta(\text{Act})$ is said to be *distributionally memoryless* if the probability distribution over actions prescribed by the strategy depends only on the current distribution over the MDP states and not on the whole history. Formally, we require that for any initial distribution $\mu_0 \in \text{Init}$ and for any two finite runs $\rho = s_0, a_0, s_1, a_1, \dots, s_n$ and $\rho' = s'_0, a'_0, s'_1, a'_1, \dots, s'_n$ that induce the sequences of probability distributions $\mu_0, \mu_1, \dots, \mu_n$ and $\mu'_0, \mu'_1, \dots, \mu'_n$ with $\mu_n = \mu'_n$, we have $\pi(\rho) = \pi(\rho')$. When the strategy π is distributionally memoryless, we write $\mathcal{M}^\pi(\mu) = \mathcal{M}^\pi(\mu, 1)$ to denote an application of a single-step distribution transformer operator.



■ **Figure 1** An MDP which will serve as our running example. The MDP contains three states $S = \{A, B, C\}$, two actions $Act = \{a, b\}$ with $Act(A) = \{a, b\}$, $Act(B) = \{a\}$, $Act(C) = \{a\}$, and its transition function is defined via $P(A, a)(A) = 1$, $P(A, b)(B) = 1$, $P(B, a)(C) = 1$, $P(C, a)(C) = P(C, a)(A) = 0.5$. We consider a singleton initial distribution set $\text{Init} = \{(A : \frac{1}{3}, B : \frac{1}{3}, C : \frac{1}{3})\}$.

It was shown in [4, 5] that distributionally memoryless strategies are sufficient for reasoning about distributional reachability, safety and reach-avoid specifications. That is, for each of these distributional specifications, there exists a strategy in the MDP under which the specification is satisfied if and only if there exists a distributionally memoryless strategy in the MDP under which the specification is satisfied. While this result need not necessarily hold for distributional ω -regular specifications, the restriction will be needed for enabling automated verification and synthesis as they can be represented in a more compact form.

Problem statement. We are now ready to define our strategy verification and synthesis problems. Consider an MDP \mathcal{M} , a set of initial distributions $\text{Init} \subseteq \Delta(S)$, and a distributional ω -regular specification φ :

1. **Strategy verification problem.** Given a distributionally memoryless strategy π , verify that the MDP \mathcal{M} satisfies φ under π from every initial distribution $\mu_0 \in \text{Init}$.
2. **Strategy synthesis problem.** Compute a distributionally memoryless strategy π , such that the MDP \mathcal{M} satisfies φ under π from every initial distribution $\mu_0 \in \text{Init}$.

► **Example 1 (Running example).** The MDP shown in Fig. 1 was considered in [4] for studying distributional safety specifications and it will serve as our running example. We consider the strategy synthesis and verification problems with respect to the distributional ω -regular specification $\varphi = \mathbf{GF}(p(B) \geq 0.249)$. For readability, we specify φ as an LTL formula over the set of atomic propositions $\text{AP} = \{(\mu(B) \geq 0.249)\}$. This is an example of a *distributional persistence specification*, which specifies that the sequence of distributions μ_0, μ_1, \dots should contain infinitely many distributions μ_i with $\mu_i(B) \geq 0.249$. The goal of strategy synthesis is to compute a strategy under which φ is satisfied. An example of such a strategy is a (distributionally) memoryless strategy π which in state A takes action b with probability 1. The goal of strategy verification is to verify this claim.

► **Remark 2 (Problem hardness).** The problem of determining if an MDP \mathcal{M} under a distributionally memoryless strategy π satisfies a distributional ω -regular specification φ is computationally hard. It was shown to be Skolem-hard already in the very restricted setting when \mathcal{M} is a Markov chain (so the strategy π is trivial) and φ is a distributional reachability specification for an affine set of goal distributions $\{\mu \in \Delta(S) \mid \mu(s_1) = 0.25\}$ [3].

► **Remark 3 (Universal and existential satisfaction problems).** In the terminology of [5] which considered distributional reachability and safety specifications, our problem corresponds to the *universal satisfaction* setting, where the specification needs to be satisfied from *every* initial distribution $\mu_0 \in \text{Init}$. Dually, one can also consider the *existential satisfaction* setting, where the specification needs to be satisfied from *at least one* initial distribution $\mu_0 \in \text{Init}$. While we will focus on the universal satisfaction setting for ease of presentation, we also show that all our results straightforwardly extend to the existential satisfaction setting as well.

► **Remark 4** (Memoryless vs distributionally memoryless strategies). Note that distributionally memoryless strategies are *not necessarily* memoryless (in the “classical” state-based sense). This fact was already shown in [4] for distributional safety specifications, where one may require infinite memory as well as randomized strategies in order to satisfy the specification. This is in stark contrast with the state-based view, where deterministic memoryless strategies are sufficient for specifications such as reachability and safety [8].

4 Certificate for Distributional ω -regular Specifications

We now present our sound and complete certificate for proving that an MDP under a distributionally memoryless strategy satisfies some distributional ω -regular specification, which is the main theoretical contribution of this work. In Section 5, we will present our algorithms for automated synthesis and verification of strategies in MDPs with respect to distributional ω -regular specifications, where the certificate will play a central role.

In what follows, we fix an MDP $\mathcal{M} = (S, Act, P)$, a set of initial distributions Init , a distributionally memoryless strategy π , and a distributional ω -regular specification φ defined over atomic propositions AP with an NBA $N^\varphi = (Q, \Sigma, \delta, q_0, F)$ where $\Sigma = 2^{\text{AP}}$.

4.1 Product Distributional Transition System

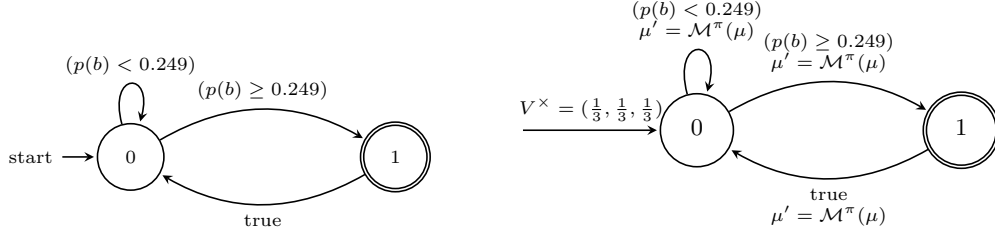
Recall from Section 2 that, for each initial distribution in Init , the MDP \mathcal{M} and the strategy π induce a sequence of distributions over the MDP states. This sequence gives rise to an infinite word in the language 2^{AP} and a run in the NBA N^φ . In what follows, we introduce product distributional transition systems (PDTs), which will allow us to synchronously reason about the distribution sequence and the NBA run.

► **Definition 5** (Product distributional transition system). *Let $\mathcal{M} = (S, Act, P)$ be an MDP, Init be a set of initial distributions, π be a distributionally memoryless strategy, and $N^\varphi = (Q, 2^{\text{AP}}, \delta, q_0, F)$ be an NBA for some distributional ω -regular specification φ defined over atomic propositions AP . A product distributional transition system (PDTs) is a transition system $\mathcal{T}^\times = (L^\times, V^\times, l_{\text{init}}^\times, \theta_{\text{init}}^\times, \mapsto^\times)$, where*

- $L^\times = Q$ is the set of states of N^φ ;
- $V^\times = \{\mu_1, \dots, \mu_{|S|}\}$ is a finite state of real-valued variables, with each variable μ_i corresponding to the probability of being in an MDP state s_i ;
- $l_{\text{init}}^\times = q_0$ is the initial state of N^φ ;
- $\theta_{\text{init}}^\times = \text{Init}$ is the set of initial distributions in \mathcal{M} ; and
- $\mapsto^\times = \{(q, q', G(\sigma), \mathcal{M}^\pi) \mid q, q' \in Q, \sigma \in 2^{\text{AP}}, q' \in \delta(q, \sigma)\}$, where $G(\sigma) = (\bigwedge_{p \in \sigma} p) \wedge (\bigwedge_{p \in \text{AP} \setminus \sigma} \neg p)$ is the predicate defined by atomic propositions contained in σ , and \mathcal{M}^π is the linear function defined by the single-step distribution transformer operator of \mathcal{M} and π .

► **Example 6.** Fig. 2 left shows the NBA for the distributional specification $\varphi = \mathbf{GF}(p(B) \geq 0.249)$ considered in Example 1. Fig. 2 right then shows the PDTs of our running example MDP in Fig. 1 and the NBA. The PDTs has the same set of locations $L^\times = \{q_0, q_1\}$ as the NBA and the set of variables $V^\times = \{\mu_1, \mu_2, \mu_3\}$ corresponding to the probabilities of being in MDP states A, B, C . The initial location is $l_{\text{init}}^\times = q_0$ and the set of initial distributions is $\text{Init} = \{(A : \frac{1}{3}, B : \frac{1}{3}, C : \frac{1}{3})\}$. Finally, the three PDTs transitions are shown in Fig. 2.

Note that PDTs indeed models a synchronous execution of a sequence of distributions over MDP states and a run in the NBA. Each infinite path $(q_0, \mu_0), (q_1, \mu_1), \dots$ in the PDTs starts from a state $(q_0, \mu_0) \in \{q_0\} \times \text{Init}$. Then, for each state (q_i, μ_i) along the infinite path,



■ **Figure 2** The figure on the left shows the NBA for distributional specification $\varphi = \text{GF}(p(b) \geq 0.249)$. The figure on the right then shows the PDTS of the MDP in Fig. 1 with the strategy that in state A takes action b with probability 1, and the NBA on the left. Each PDTS transition is labeled with its guard (top line) and its update function (bottom line). We write $\mu' = \mathcal{M}^\pi(\mu)$ as a shorthand notation for $\mu' = \{A : 0.5 \cdot \mu(C), B : \mu(A), C : \mu(B) + 0.5 \cdot \mu(C)\}$.

the next state is obtained by applying some enabled transition $(q_i, q_{i+1}, G(\sigma_i), \mathcal{M}^\pi)$. For the transition to be enabled, we must have $\sigma_i = \{p \in \text{AP} \mid \mu_i \models p\}$ be the unique letter defined by all atomic propositions satisfied in μ_i . The PDTS then moves to a state (q_{i+1}, μ_{i+1}) , where $\mu_{i+1} = \mathcal{M}^\pi(\mu_i)$ is indeed the next distribution in the sequence and $q_{i+1} \in \delta(q_i, \sigma_i)$ is indeed a successor state in the NBA.

An infinite path $(q_0, \mu_0), (q_1, \mu_1), \dots$ in the PDTS is *accepting* if $(q_i, \mu_i) \in F \times \Delta(S)$ for infinitely many $i \in \mathbb{N}_0$, i.e. if it visits states with locations belonging to the accepting set of the NBA infinitely often. The following proposition formalizes the relationship between the satisfaction of a distributional ω -regular specification in the MDP and the existence of an accepting infinite path in the PDTS. We state the proposition for a single initial distribution $\mu_0 \in \text{Init}$, so that it is applicable both in the universal and the existential satisfaction problem settings (see Remark 3).

► **Proposition 7.** *An MDP \mathcal{M} with an initial distribution $\mu_0 \in \Delta(S)$ satisfies a distributional ω -regular specification φ under a distributionally memoryless strategy π if and only if there exists an accepting infinite path in the PDTS $\mathcal{T}^\times = (L^\times, V^\times, l_{\text{init}}^\times, \theta_{\text{init}}^\times, \mapsto^\times)$.*

Proof. Suppose that MDP \mathcal{M} with initial distribution μ_0 satisfies distributional ω -regular specification φ under distributionally memoryless strategy π . Let μ_0, μ_1, \dots be the sequence of distributions induced by the MDP \mathcal{M} under strategy π , and let $\sigma_0, \sigma_1, \dots$ be the induced infinite word from the initial distribution μ_0 . By the definition of satisfiability of distributional ω -regular specifications in Section 3, the infinite word $\sigma_0, \sigma_1, \dots$ is accepted by the NBA N^φ . Hence, there exists a run q_0, q_1, \dots in N^φ such that $q_{i+1} \in \delta(q_i, \sigma_i)$ for each $i \in \mathbb{N}$. But this also means that $(q_0, \mu_0), (q_1, \mu_1), \dots$ is an accepting path in the PDTS $\mathcal{T}^\times = (L^\times, V^\times, l_{\text{init}}^\times, \theta_{\text{init}}^\times, \mapsto^\times)$, which proves one direction of the proposition.

Conversely, suppose that there exists an accepting infinite path $(q_0, \mu_0), (q_1, \mu_1), \dots$ in the PDTS $\mathcal{T}^\times = (L^\times, V^\times, l_{\text{init}}^\times, \theta_{\text{init}}^\times, \mapsto^\times)$. Then, by the definition of transition update functions in the PDTS, we know that μ_0, μ_1, \dots is the sequence of distributions induced by the MDP \mathcal{M} under strategy π . Moreover, by the definition of transition guards in the PDTS, we know that $q_{i+1} \in \delta(q_i, \sigma_i)$ for each $i \in \mathbb{N}$ with σ_i being the unique letter defined by atomic propositions in AP satisfied in μ_i . Hence, q_0, q_1, \dots is an infinite run in the NBA N^φ induced by the infinite word $\sigma_0, \sigma_1, \dots$. But from the fact that $(q_0, \mu_0), (q_1, \mu_1), \dots$ is an accepting run in the PDTS, it follows that $q_i \in F$ for infinitely many $i \in \mathbb{N}$ and so the infinite word $\sigma_0, \sigma_1, \dots$ is accepted by the NBA N^φ . By the definition of satisfiability of distributional ω -regular specifications in Section 3, this implies that MDP \mathcal{M} with initial distribution μ_0 satisfies specification φ under strategy π , which concludes our proof. ◀

4.2 Distributional Certificates

We are now ready to define our notion of distributional certificates. A *distributional certificate* is a pair (\mathcal{C}, I) that consists of two components – a *distributional Büchi ranking function* \mathcal{C} and a *distributional invariant* I . The distributional Büchi ranking function $\mathcal{C} : Q \times \Delta(S) \rightarrow \mathbb{R}$ is a function that to each state of the PDTS assigns a real value, which is required to satisfy two conditions. First, the *Initial condition* requires the value of \mathcal{C} to be non-negative at all initial states of the PDTS. Second, the *Büchi ranking condition* requires that, for every reachable state in the PDTS at which the value of \mathcal{C} is non-negative, there exists at least one successor state at which non-negativity is preserved. Furthermore, the value of \mathcal{C} decreases by at least 1 if the state is not contained in the accepting set of the PDTS. We prove in Theorem 9 below that these two conditions are necessary and sufficient to ensure that, for every initial state (q_0, μ_0) in the PDTS, there exists an accepting infinite path in the PDTS.

Note that the Büchi ranking condition needs to be satisfied only at reachable states of the PDTS. However, the problem of determining the exact set of reachable states is not feasible. Hence, with later automation in mind, we append our certificate with a distributional invariant $I \subseteq Q \times \Delta(S)$, which is a set that over-approximates the set of reachable states in the PDTS. This is ensured by extending the Initial condition to require that all initial states of the PDTS are contained in the invariant I , and by extending the Büchi ranking condition to require that the successor state described above is also contained in the invariant I .

The following definition formalizes this intuition. In what follows, for each letter $\sigma \in 2^{\text{AP}}$ and distribution $\mu \in \Delta(S)$, we write $\mu \models G(\sigma)$ as a shorthand for $\mu \models (\bigwedge_{p \in \sigma} p) \wedge (\bigwedge_{p \in \text{AP} \setminus \sigma} \neg p)$.

► **Definition 8** (Distributional certificate). *A distributional certificate for an MDP \mathcal{M} with a set of initial distributions Init , a distributionally memoryless strategy π , and a distributional ω -regular specification φ with NBA N^φ , is a tuple (\mathcal{C}, I) consisting of a function $\mathcal{C} : Q \times \Delta(S) \rightarrow \mathbb{R}$ and a set $I \subseteq Q \times \Delta(S)$, such that the following conditions hold:*

- **Initial condition.** *For all $\mu_0 \in \text{Init}$, we have $\mathcal{C}(q_0, \mu_0) \geq 0$ and $(q_0, \mu_0) \in I$.*
- **Büchi ranking condition.** *We have the following:*
 - **Non-negativity at accepting states.** *For all NBA states $q \in F$ and letters $\sigma \in 2^{\text{AP}}$,*

$$\begin{aligned} \forall \mu \in \mathbb{R}^{|\mathcal{S}|}. \quad & \bigvee_{q' \in \delta(q, \sigma)} \mu \in \Delta(S) \wedge \mu \models G(\sigma) \wedge \mathcal{C}(q, \mu) \geq 0 \wedge (q, \mu) \in I \\ & \implies \mathcal{C}(q', \mathcal{M}^\pi(\mu)) \geq 0 \wedge (q', \mathcal{M}^\pi(\mu)) \in I. \end{aligned} \quad (1)$$

- **Strict decrease and non-negativity at non-accepting states.** *For all NBA states $q \notin F$ and letters $\sigma \in 2^{\text{AP}}$,*

$$\begin{aligned} \forall \mu \in \mathbb{R}^{|\mathcal{S}|}. \quad & \bigvee_{q' \in \delta(q, \sigma)} \mu \in \Delta(S) \wedge \mu \models G(\sigma) \wedge \mathcal{C}(q, \mu) \geq 0 \wedge (q, \mu) \in I \\ & \implies \mathcal{C}(q, \mu) - 1 \geq \mathcal{C}(q', \mathcal{M}^\pi(\mu)) \geq 0 \wedge (q', \mathcal{M}^\pi(\mu)) \in I. \end{aligned} \quad (2)$$

The following theorem establishes that distributional certificates provide a sound and complete proof rule for proving that an MDP under a distributionally memoryless strategy satisfies a distributional ω -regular specification.

► **Theorem 9** (Soundness and completeness). *An MDP \mathcal{M} with a set of initial distributions Init under a distributionally memoryless strategy π satisfies a distributional ω -regular specification φ if and only if there exists a distributional certificate for \mathcal{M} , Init , π and φ .*

Proof.

Soundness. Suppose that there exists a distributional certificate (\mathcal{C}, I) for \mathcal{M} , Init , π and φ . To show that φ is satisfied, by Proposition 7 it suffices to show that the PDTS \mathcal{T}^\times admits an accepting infinite path for every initial state in $\{q_0\} \times \text{Init}$.

Fix an initial state $(q_0, \mu_0) \in \{q_0\} \times \text{Init}$. By the Initial condition in Definition 8, we know that $\mathcal{C}(q_0, \mu_0) \geq 0$ and $(q_0, \mu_0) \in I$. Hence, by the Büchi ranking condition in Definition 8, we can repeatedly select successor states in order to obtain an infinite path $(q_0, \mu_0), (q_1, \mu_1), \dots$ in \mathcal{T}^\times such that, for each $i \in \mathbb{N}_0$, we have

- $\mathcal{C}(q_i, \mu_i) \geq 0$ and $(q_i, \mu_i) \in I$, and
 - whenever $q_i \notin F$ is not an accepting state in N^φ , we have $\mathcal{C}(q_i, \mu_i) - 1 \geq \mathcal{C}(q_{i+1}, \mu_{i+1})$.
- We claim that $(q_0, \mu_0), (q_1, \mu_1), \dots$ is an accepting infinite path in \mathcal{T}^\times . To prove this, note that for every (q_i, μ_i) with $q_i \notin F$, the value of \mathcal{C} needs to keep decreasing by at least 1 in each subsequent step while also remaining non-negative. Hence, in at most $\lceil \mathcal{C}(q_i, \mu_i) \rceil$ steps, the path must again reach an accepting state. Thus, the infinite path $(q_0, \mu_0), (q_1, \mu_1), \dots$ reaches accepting states in $F \times \Delta(S)$ infinitely many times and is an accepting infinite path. Since the initial state $(q_0, \mu_0) \in \{q_0\} \times \text{Init}$ was arbitrary, this concludes the proof.

Completeness. Conversely, suppose that \mathcal{M} with a set of initial distributions Init under distributionally memoryless strategy π satisfies distributional ω -regular specification φ . We construct an instance (\mathcal{C}, I) of a distributional certificate for \mathcal{M} , Init , π and φ as follows.

Consider an arbitrary but throughout fixed enumeration $q_1, \dots, q_{|Q|}$ of NBA states. We define an operator $\text{NEXT} : Q \times \Delta(S) \rightarrow Q \times \Delta(S)$ via

- $\text{NEXT}(q, \mu) = (q_i, \mathcal{M}^\pi(\mu))$ with i being the smallest index such that $(q, \mu), (q_i, \mathcal{M}^\pi(\mu))$ are successor states along some accepting infinite path in the PDTS, if such an accepting infinite path exists, or
- $\text{NEXT}(q, \mu) = (q, \mu)$, otherwise.

In other words, $\text{NEXT}(q, \mu)$ fixes a successor state of (q, μ) along some accepting infinite path in the PDTS if such a path exists, or halts the sequence at the state (q, μ) otherwise. Therefore, the transitive closure of the operator $\text{NEXT}(q, \mu)$ from the set of initial PDTS states $\{q_0\} \times \text{Init}$ allows us to consistently fix a *unique accepting infinite path* for each state (q, μ) that is contained along some accepting infinite path.

We can now define our distributional certificate (\mathcal{C}, I) . Let distributional invariant $I \subseteq Q \times \Delta(S)$ be the set of all states in the PDTS that are reachable from $\{q_0\} \times \text{Init}$ under the transitive closure of the NEXT operator. Moreover, for each PDTS state $(q, \mu) \in I$, let $d_{\text{accept}}(q, \mu)$ denote the number of steps when applying the NEXT operator until an accepting state in $F \times \Delta(S)$ is reached, with $d_{\text{accept}}(q, \mu) = 0$ if $(q, \mu) \in F \times \Delta(S)$ is an accepting state. Finally, let the distributional Büchi ranking function \mathcal{C} be defined via

$$\mathcal{C}(q, \mu) = \begin{cases} d_{\text{accept}}(q, \mu), & \text{if } (q, \mu) \in I, \\ -1, & \text{otherwise.} \end{cases}$$

We claim that (\mathcal{C}, I) is an instance of a distributional certificate. The Initial condition in Definition 8 is satisfied since the MDP \mathcal{M} under strategy μ satisfies φ , therefore every initial state $(q_0, \mu) \in \{q_0\} \times \text{Init}$ belongs to some accepting infinite path in the PDTS and so $(q_0, \mu) \in I$ and $d_{\text{accept}}(q_0, \mu) \geq 0$. On the other hand, by the definition of the NEXT operator and I , for each $(q, \mu) \in I$ we have that also $\text{NEXT}(q, \mu) \in I$. Moreover, $\text{NEXT}(q, \mu) = d_{\text{accept}}(q, \mu) - 1 \geq 0$ if $q \notin F$ and $\text{NEXT}(q, \mu) \geq 0$ if $q \in F$. Hence, the Büchi ranking condition in Definition 8 is also satisfied, and (\mathcal{C}, I) is a distributional certificate. ◀

► **Example 10.** Consider again the MDP in Fig. 1. As in Example 1, consider the strategy π which in state A takes action b with probability 1, and the distributional specification $\varphi = \mathbf{GF}(p(B) \geq 0.249)$. An NBA for φ and the resulting PDTS are shown in Fig. 2. The following is an example of a distributional certificate (\mathcal{C}, I) for \mathcal{M} , Init , π and φ :

$$\mathcal{C}(q, \mu) = \begin{cases} 1 + 250 \cdot \mu(A) + 750 \cdot \mu(C), & \text{if } q = q_0, \\ 1.25 - 1.25 \cdot \mu(C), & \text{if } q = q_1, \end{cases}$$

and $I = \{(q_0, \mu) \mid 1.25 + \mu(A) + 1 \cdot \mu(B) - \mu(C) \geq 0\} \cup \{(q_1, \mu) \mid \mu(A) + 0.25 \cdot \mu(B)\}$.

One can verify by inspection that the Initial condition and the Büchi ranking condition in Definition 8 are satisfied. We note that the above distributional certificate (\mathcal{C}, I) is the certificate computed by our prototype implementation in Section 6.

► **Remark 11 (Distributional certificates for the existential problem).** As discussed in Section 2 and Remark 3, our distributional certificate in Definition 4.2 and our soundness and completeness result in Theorem 9 consider the universal satisfaction setting. However, their extension to the existential setting is immediate. The only required change in the definition of distributional certificates is to require the Initial condition in Definition 4.2 to hold *for some* initial distribution $\mu_0 \in \text{Init}$. On the other hand, the soundness and completeness proof proceeds analogously as in the proof of Theorem 9, with the difference in the completeness proof being that the distributional invariant I is defined as the transitive closure of the NEXT operator with the singleton initial set $\{(q_0, \mu_0)\}$, rather than the initial set $\{q_0\} \times \text{Init}$.

5 Template-based Strategy Verification and Synthesis with Certificates

We now present our algorithms for the strategy verification and synthesis problems for distributional ω -regular specifications. The core of the verification algorithm is to synthesize an *affine* distributional certificate in the PDTS of the input MDP and the specification, which proves that the specification is satisfied. When we move from strategy verification to strategy synthesis, we also synthesize an *affine* distributionally memoryless strategy.

The restriction to affine distributional certificates and affine distributionally memoryless strategies is needed to ensure tractability. While Theorem 9 establishes soundness and completeness of our distributional certificates, in combination with Remark 2 it also implies that giving a sound and complete algorithm for synthesizing distributional certificates is Skolem-hard. Hence, in this section, we instead focus on designing sound and *relatively complete* algorithms for synthesizing an affine distributional certificate together with an affine distributionally memoryless strategy (the latter for the synthesis problem), when they exist.

Affine distributional certificates and strategies. We first formalize the notions of affine distributional certificates and distributionally memoryless strategies:

- **Affine distributional certificates.** A distributional certificate (\mathcal{C}, I) is said to be *affine* if both the distributional Büchi ranking function \mathcal{C} and the distributional invariant I can be expressed in terms of affine expressions and inequalities over the space $\Delta(S)$ of probability distributions over MDP states. We require \mathcal{C} to be of the form

$$\mathcal{C}(q, \mu) = \sum_{i=1}^{|S|} a_i^q \cdot \mu(s_i) + b^q, \quad (3)$$

where $a_1^q, \dots, a_{|S|}^q, b^q$ are some real valued coefficients for each NBA state $q \in Q$. That is, for each NBA state q , the function $\mathcal{C}(q, \cdot)$ is an affine function over the probabilities

of being in each MDP state, with $\mu(s_1), \dots, \mu(s_{|V|})$ being the variables that capture probabilities of being in each MDP state and $a_1^q, \dots, a_{|S|}^q, b^q$ being the coefficients of the affine function. Similarly, we require I to be a set defined by a conjunction of N_I affine inequalities over the probabilities of being in each MDP state, i.e. to be of the form

$$I = \left\{ (\mu, q) \in \Delta(S) \times Q \mid \bigwedge_{k=1}^{N_I} I^k(q, \mu) \geq 0 \right\}, \quad (4)$$

where each $I^k(q, \mu) = \sum_{i=1}^{|S|} c_i^{k,q} \cdot \mu(s_i) + d^{k,q} \geq 0$ and $c_1^{k,q}, \dots, c_{|S|}^{k,q}, d^{k,q}$ are some real valued coefficients for each NBA state $q \in Q$ and each $k \in \{1, \dots, N_I\}$. The number N_I is referred to as the *size of the invariant* and will be an algorithm parameter.

- **Affine distributionally memoryless strategies.** A distributionally memoryless strategy $\pi : \Delta(S) \rightarrow \Delta(Act)$ is said to be *affine*, if for each state $s \in S$, action $a \in Act$ and state distribution $\mu \in \Delta(S)$, the probability of taking action a in state s given the current distribution over states μ is of the form

$$\pi(s, a)(\mu) = \frac{\sum_{i=1}^{|S|} e_{i,s,a}^\pi \cdot \mu(s_i) + f_{s,a}^\pi}{\sum_{i=1}^{|S|} g_{i,s}^\pi \cdot \mu(s_i) + h_s^\pi}, \quad (5)$$

where $e_{1,s,a}^\pi, \dots, e_{|S|,s,a}^\pi, f_{s,a}^\pi$ and $g_{1,s}^\pi, \dots, g_{|S|,s}^\pi, h_s^\pi$ are real valued constants. The denominator is used in order to normalize the probabilities such that the sum of probabilities of all actions being taken at a state s is 1.

Algorithm input. Both our verification and synthesis algorithms take as input an MDP $\mathcal{M} = (S, Act, P)$, a set of initial distributions Init , and a distributional ω -regular specification φ defined over atomic propositions AP . We assume that the distributional ω -regular specification is provided via an NBA $N^\varphi = (Q, \Sigma, \delta, q_0, F)$ with letters $\Sigma = 2^{\text{AP}}$, which accepts the same set of infinite words over 2^{AP} as φ . Finally, the algorithms also take as input the size of the invariant N_I that needs to be synthesized. The verification algorithm in addition takes as input an affine distributionally memoryless strategy π .

Algorithm overview. Both verification and synthesis algorithms follow a template-based synthesis approach and proceed in four steps. In the first step, the PDTS of the input MDP and the distributional specification is constructed. In the second step, the algorithms fix a symbolic template for the affine distributional certificate, i.e. symbolic variables for each real valued coefficient in affine expressions that define the certificate. The synthesis algorithm also fixes a symbolic template for the affine distributionally memoryless strategy. In the third step, the algorithms collect a system of constraints over the symbolic template variables, that together encode all defining conditions of distributional certificates in Definition 8 as well as conditions for the strategy template to define a valid distributionally memoryless strategy (the latter for the synthesis algorithm). Finally, in the fourth step, the collected system of constraints is solved by using an SMT-solver, to get a concrete valuation of the symbolic template variables which in turn gives rise to a distributional certificate and a distributionally memoryless strategy. In what follows, we detail each of these four steps.

Step 1: Constructing the PDTS. In this step, the PDTS $\mathcal{T}^\times = (L^\times, V^\times, l_{\text{init}}^\times, \theta_{\text{init}}^\times, \mapsto^\times)$ is constructed from the given MDP \mathcal{M} and the NBA N^φ , as explained in Section 4.1.

Step 2: Fixing templates. The algorithms fix a template for the affine distributional certificate (\mathcal{C}, I) , while the synthesis algorithm also fixes a template for the affine distributionally memoryless strategy π . The novelty, compared to prior work on verification and synthesis for distributional reachability and safety specifications [4, 5], lies in a more complex template design for the distributional certificate, which is now defined with respect to the PDTs:

- **Template for \mathcal{C} .** Recall that an affine distributional Büchi ranking function is of the form $\mathcal{C}(q, \mu) = \sum_{i=1}^{|S|} a_i^q \cdot \mu(s_i) + b^q$ as in eq. (3). Hence, the template for \mathcal{C} is defined by introducing a set of symbolic template variables $a_1^q, \dots, a_{|S|}^q, b^q$ for each NBA state $q \in Q$.
- **Template for I .** Similarly, the template for an affine distributional invariant I is of the form as in eq. (4). Hence, the template for I is defined by introducing a set of symbolic template variables $c_1^{k,q}, \dots, c_{|S|}^{k,q}, d^{k,q}$ for each NBA state $q \in Q$ and each $k \in \{1, \dots, N_I\}$, where N_I is the algorithm parameter that specifies the size of the invariant.
- **Template for π (synthesis algorithm).** The template for an affine distributionally memoryless strategy π is of the form as in eq. (5), hence it is defined by introducing symbolic template variables $e_{1,s,a}^\pi, \dots, e_{|S|,s,a}^\pi, f_{s,a}^\pi$ and $g_{1,s}^\pi, \dots, g_{|S|,s}^\pi, h_s^\pi$ for each state $s \in S$ and action $a \in \text{Act}$. Note that, in the special case when we are interested in synthesizing memoryless strategies instead of distributionally memoryless strategies, the strategy template becomes simpler. Instead of the template as in eq. (5), we introduce a single symbolic template variable $p_{s,a}^\pi$ for each state-action pair $s \in S$ and $a \in \text{Act}$, to encode the probability of taking action a in state s .

Step 3: Collecting constraints. In this step, the algorithms collect a system of constraints over the symbolic template variables that together encode that \mathcal{C} and \mathcal{I} indeed define a valid distributional certificate. For the synthesis algorithm, we also collect a system of constraints that encode that π defines a valid distributionally memoryless strategy. In each of the following constraints, each appearance of \mathcal{C} , \mathcal{I} and π is replaced by the symbolic template introduced in Step 2, in the form as in eq. (3), (4) and (5). Moreover, we write $\mu \in \Delta(S)$ for the conjunction of affine inequalities $\bigwedge_{i=1}^{|S|} (\mu_i \geq 0) \wedge (\mu_1 + \dots + \mu_{|S|} = 1)$.

- **Initial condition.** We define

$$\Phi_{\text{init}} \equiv \forall \mu \in \mathbb{R}^{|S|}. \mu \in \Delta(S) \wedge \mu \in \text{Init} \implies \mathcal{C}(q_0, \mu) \geq 0 \wedge \bigwedge_{k=1}^{N_I} I^k(q_0, \mu) \geq 0.$$

- **Büchi ranking condition for accepting states.** For each accepting state $q \in F$ and letter $\sigma \in 2^{\text{AP}}$ in the NBA, we define

$$\begin{aligned} \Phi_{\text{Büchi},q,\sigma} &\equiv \forall \mu \in \mathbb{R}^{|S|}. \bigvee_{q' \in \delta(q,\sigma)} \mu \in \Delta(S) \wedge \mu \models G(\sigma) \wedge \mathcal{C}(q, \mu) \geq 0 \wedge \bigwedge_{k=1}^{N_I} I^k(q, \mu) \geq 0 \\ &\implies \mathcal{C}(q', \mathcal{M}^\pi(\mu)) \geq 0 \wedge \bigwedge_{k=1}^{N_I} I^k(\mathcal{M}^\pi(q', \mu)) \geq 0. \end{aligned}$$

- **Büchi ranking condition for nonaccepting states.** For each non-accepting state $q \in Q \setminus F$ and letter $\sigma \in 2^{\text{AP}}$ in the NBA, we define

$$\begin{aligned} \Phi_{\text{Büchi},q,\sigma} &\equiv \forall \mu \in \mathbb{R}^{|S|}. \bigvee_{q' \in \delta(q,\sigma)} \mu \in \Delta(S) \wedge \mu \models G(\sigma) \wedge \mathcal{C}(q, \mu) \geq 0 \wedge \bigwedge_{k=1}^{N_I} I^k(q, \mu) \geq 0 \\ &\implies \mathcal{C}(\mu, q) - 1 \geq \mathcal{C}(\mathcal{M}^\pi(\mu), q') \geq 0 \wedge \bigwedge_{k=1}^{N_I} I^k(\mathcal{M}^\pi(q', \mu)) \geq 0. \end{aligned}$$

- **Strategy conditions (synthesis algorithm).** For the strategy template to indeed define a valid affine distributionally memoryless strategy, we require that:

$$\Phi_\pi \equiv \bigwedge_{s \in S} \left(\sum_{a \in Act} \pi(s, a)(\mu) = 1 \wedge \bigwedge_{a \in Act} (\pi(s, a)(\mu) \geq 0) \right).$$

In the above definitions, note that $\mathcal{M}^\pi(\mu)$ is the one-step successor from distribution μ when policy π is applied in the MDP, computed as: $\sum_{s \in S, a \in Act(s)} \pi(s, a)(\mu) \cdot P(s, a)$.

Step 4: Constraint solving. The strategy condition constraint Φ_π is a purely existentially quantified Boolean combination of affine inequalities over the symbolic template variables. However, constraints Φ_{init} and $\Phi_{\text{Büchi}, q, \sigma}$ are all of the form

$$\forall \mu \in \mathbb{R}^{|S|}. \bigvee_{i=1}^m \bigwedge_{j=1}^n \text{aff-expr}_{i,j}(t, \mu) \geq 0 \geq \bigwedge_{l=1}^k \text{poly-expr}_k(t, \mu) \geq 0,$$

where t is the vector of all symbolic template variables, $\text{aff-expr}_{i,j}(t, \mu)$'s are some affine functions and poly-expr_k 's are some polynomial functions over the vectors of variables t and μ . Polynomial expressions on the right hand side arise due to the quotients of affine expressions that define affine distributionally memoryless strategies, see eq. (5). Hence, multiplying both sides of the inequality by the affine expressions appearing in denominators results in polynomial expressions over variables in t and μ .

The problem of synthesizing affine distributional certificates and affine distributionally memoryless strategies then reduces to solving a system of constraints that contain quantifier alternation $\exists t. \forall \mu$. Such quantifier alternation over real-valued variables is generally hard to handle directly and can lead to inefficiency in solvers. In order to allow for a more efficient constraint solving, before passing our system of constraints to an SMT-solver, we first apply Handelman's theorem [27] to translate Φ_{init} and $\Phi_{\text{Büchi}, q, \sigma}$ into a purely existentially quantified system of polynomial constraints over the symbolic template variables in t and auxiliary variables introduced by the translation, whose satisfiability implies satisfiability of the original constraints. This translation is common in template-based program analysis, see [7] for details. This step allows for more efficient constraint solving as well as better bound on the algorithm complexity. Finally, the resulting purely existentially quantified system of polynomial constraints over real-valued variables is solved via an SMT solver.

In the special case when we are interested in synthesizing memoryless strategies rather than distributionally memoryless strategies, we may use Farkas' lemma [25] rather than Handelman's theorem. This yields a *sound and complete* translation into an equisatisfiable purely existentially satisfied system of constraints.

Soundness, relative completeness, complexity. Soundness of our algorithms follows from the soundness of all four steps above, including soundness of the transformations via Handelman's theorem and Farkas' lemma [7]. Since the Farkas' lemma transformation leads to an equisatisfiable system of constraints, it also follows that our algorithm is *relatively complete* – it is guaranteed to synthesize an affine distributional certificate and memoryless strategy whenever they exist. Finally, our algorithms provide a PSPACE complexity upper bound as they reduce the synthesis and verification problems to solving a sentence in the existential first-order theory of the reals. The following theorem summarizes these results.

► **Theorem 12.**

Soundness: *If the algorithm returns an affine distributional certificate (\mathcal{C}, I) and an affine distributionally memoryless strategy π (for the synthesis algorithm), then the MDP \mathcal{M} with initial distributions Init under strategy π satisfies specification φ .*

Relative completeness: *If there exist an affine distributional certificate (\mathcal{C}, I) and a memoryless strategy π , then there exists an invariant size $N_I \in \mathbb{N}$ such that (\mathcal{C}, I) and π are computed by the algorithm.*

Complexity: *The runtime of the algorithm is in PSPACE in the size of the encoding of the MDP, NBA N^φ , strategy π (for the verification algorithm) and invariant size $N_I \in \mathbb{N}$.*

6 Experimental Evaluation

We implemented a prototype of our method in Python 3 and experimentally evaluated it on a number of challenging verification and synthesis tasks collected from the literature on distributional specifications in MDPs. Our prototype takes as input an MDP (in the Prism [30] input format) and an LTL specification. The LTL specification is then translated into an NBA via Spot [24]. For the constraint solving step in our algorithms, we use PolyQEnt [16] which provides a tooling support for quantifier elimination via Farkas' lemma and Handelman's theorem. PolyQEnt uses Z3 [23] and MathSAT5 [20] as backend SMT solvers for the final system of purely existentially quantified constraints. We set the invariant size parameter to $N_I = 1$, which was sufficient for all our experiments. Our experiments were conducted on consumer-grade hardware (AMD Ryzen 5 5625U CPU, 8GB RAM).

Benchmarks. We evaluated our method on several examples collected from the literature:

- **GridWorld (synthesis).** Motivated by [5], these benchmarks model robot swarms in gridworld environments. Initially, all robots are placed in the top-left corner of the gridworld environment. Some of the cells are covered by walls whereas some are slippery and with certain probability may lead to moving in an undesired direction. Hence, each environment induces an MDP. As shown in [5], the evolution of a robot swarm can be analyzed by taking the distribution transformer view of MDPs and considering how the robots are distributed across the gridworld cells at each time step. In Table 1, we consider 5 gridworld benchmarks of varying sizes and consider two distributional specifications: (1) at least 90% of robots should be in some slippery target cell infinitely often, and (2) in addition, at most 50% of robots should occupy some narrow passage at any point in time.
- **PageRank (verification).** We consider a Markov chain representation of the PageRank algorithm taken from [2]. Given the context, we consider various verification tasks, which are of the form: always if the probability mass at some vertex/page is above a threshold, then eventually, it must be above a threshold in another page.
- **Pharmacokinetics (verification)** We also consider a 6 state Markov chain from [2] which is adapted from a Pharmacokinetics example in [11]. We use the two queries that were listed in [2] as the motivating examples to obtain our specifications.
- **Benchmarks from [4] (verification and synthesis).** Finally, we collect 3 pairs of verification and synthesis tasks from [4]. In the verification task a strategy is fixed, whereas in the synthesis task one also needs to compute the strategy. While [4] considered distributional safety specifications, we design more complex ω -regular specifications.

Results. Our experimental results are shown in Table 1. Our results demonstrate that our prototype is able to solve a number of challenging verification and synthesis tasks for distributional ω -regular specifications in MDPs, that were beyond the reach of all existing

■ **Table 1** For each experiment we report, from left to right, the benchmark, specification, task (verification or synthesis), the number of coefficients, the number of constraints, the number of coefficients in PolyQEnt generated file (i.e. after application of Farkas’ lemma), the number of constraints in PolyQEnt generated file, SMT-solving time, and the total runtime.

Model	Specification	Task	Coeff #	Const #	PQ Coeff #	Query #	SMT time	Total time
GW (3*3)	G F “V5>=0.9”	Synth	62	41	171	34	< 2s	6s
GW (3*3)	G F “V5>=0.9” & G “V4<=0.5”	Synth	80	45	277	49	< 5s	< 5s
GW (4*4)	G F “V11>=0.9”	Synth	121	79	286	81	< 10s	10s
GW (4*4)	G F “V11>=0.9” & G “V9<=0.5”	Synth	153	83	448	87	12s	13s
GW (5*5)	G F “V19>=0.9”	Synth	198	129	435	131	302s	303s
PageRank	F G “V2>0.2”	Verify	60	13	400	35	63s	64s
PageRank	G (“0.2<=V0” → F “0.2<=V2”)	Verify	48	13	253	22	17s	18s
PageRank	G (“0.2<=V2” → F “0.2<=V2”)	Verify	48	13	253	22	8s	8s
PageRank	G (“0.2<=V3” → F “0.2<=V2”)	Verify	48	13	253	22	44s	45s
PageRank	G (“0.2<=V4” → F “0.2<=V2”)	Verify	48	13	253	22	5s	6s
PageRank	G “V0>=0.2” “V1>=0.2” “V2>=0.2” “V3>=0.2” “V4>=0.2” → F “V2=1”	Verify	60	19	569	44	136s	137s
PageRank	F “V2=1” → G “V1<=0.2”	Verify	144	35	630	46	6s	6s
Pharmacokinetics	F “V4=1”	Verify	42	9	176	14	< 1s	< 1s
Pharmacokinetics	G (“0.13<=V3<=0.2” “0<=V3<0.13”)	Verify	56	11	365	28	102s	102s
CAV23 [4]	G F “V1>=0.249”	Verify	16	7	85	13	< 2s	< 2s
CAV23 [4]	G F “V1>=0.249”	Synth	20	14	108	16	7s	< 2s
CAV23 [4]	“V1>= 0.249” U “V2 >= 0.25”	Verify	32	13	185	22	< 5s	7s
CAV23 [4]	“V1>= 0.249” U “V2 >= 0.25”	Synth	36	20	189	25	7s	5s
CAV23 [4]	“0.334>=V1>=0.332” U “V0=0.25”	Verify	32	17	229	24	< 4s	4s
CAV23 [4]	“0.334>=V1>=0.332” U “V0=0.25”	Synth	36	20	233	28	< 1s	< 1s

methods. This is achieved at runtimes that are comparable or even lower than those reported by earlier methods for distributional reachability and safety specifications in [4, 5] on benchmarks of similar size. Hence, even though we consider a significantly more general class of distributional ω -regular specifications, our algorithms do not lead to a significant computational overhead. Moreover, for all our strategy synthesis tasks, our prototype was able to compute *memoryless strategies* that lead to distributional specification satisfaction. This demonstrates the generality of relative completeness guarantees provided by our algorithms.

We also make some observations. First, as can be seen from runtimes reported in Table 1, the final SMT-solving step is computationally the most expensive step of our algorithms. Constraint generation took at most a few seconds in all cases. Second, we observe that strategy synthesis tasks are generally computationally more expensive, which is expected given that they require synthesizing both the strategy and the distributional certificate. However, in some cases the synthesis problems can also be solved more efficiently. This is demonstrated by the last 6 experiments (CAV’23 in Table 1), where synthesis is achieved at lower runtimes due to our prototype being able to compute a simple memoryless strategy that was easier to verify, compared to the strategy considered in the verification task.

Finally, regarding the invariant size parameter, we used $N_I = 1$ because it was sufficient for all our benchmarks. We also ran our prototype tool with $N_I = 2$ on 12 of the benchmarks and 8 of them were solved within the timeout of 5 minutes. The timeouts are likely due to the larger size of the final system of constraints. Indeed, given more time, we expect our method can be effectively applied with larger template sizes as well.

7 Conclusion

In this paper, we considered distributional ω -regular specifications in MDPs and addressed the problems of strategy verification and synthesis. We developed new notions of product distributional transition systems between an MDP and an NBA. We then introduced distributional certificates, using which we provided template-based synthesis algorithms for

strategy verification and synthesis. Our experiments demonstrate the benefits and promise of our approach. As future work, we would like to go beyond MDPs and consider partially observable MDPs. Moreover, it would be interesting to lift the objectives from NBA to Rabin automata, where even the notion of distributional certificates is unclear.

References

- 1 Alessandro Abate, Mirco Giacobbe, and Diptarko Roy. Stochastic omega-regular verification and control with supermartingales. In Arie Gurfinkel and Vijay Ganesh, editors, *Computer Aided Verification - 36th International Conference, CAV 2024, Montreal, QC, Canada, July 24-27, 2024, Proceedings, Part III*, volume 14683 of *Lecture Notes in Computer Science*, pages 395–419. Springer, 2024. doi:10.1007/978-3-031-65633-0_18.
- 2 Manindra Agrawal, S. Akshay, Blaise Genest, and P. S. Thiagarajan. Approximate verification of the symbolic dynamics of Markov chains. *J. ACM*, 62(1):2:1–2:34, 2015. doi:10.1145/2629417.
- 3 S. Akshay, Timos Antonopoulos, Joël Ouaknine, and James Worrell. Reachability problems for Markov chains. *Inf. Process. Lett.*, 115(2):155–158, 2015. doi:10.1016/J.IPL.2014.08.013.
- 4 S. Akshay, Krishnendu Chatterjee, Tobias Meggendorfer, and Dorde Zikelić. MDPs as distribution transformers: Affine invariant synthesis for safety objectives. In Constantin Enea and Akash Lal, editors, *Computer Aided Verification - 35th International Conference, CAV 2023, Paris, France, July 17-22, 2023, Proceedings, Part III*, volume 13966 of *Lecture Notes in Computer Science*, pages 86–112. Springer, 2023. doi:10.1007/978-3-031-37709-9_5.
- 5 S. Akshay, Krishnendu Chatterjee, Tobias Meggendorfer, and Dorde Zikelić. Certified policy verification and synthesis for MDPs under distributional reach-avoidance properties. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*, pages 3–12. ijcai.org, 2024. URL: <https://www.ijcai.org/proceedings/2024/1>.
- 6 S. Akshay, Blaise Genest, and Nikhil Vyas. Distribution-based objectives for Markov decision processes. In Anuj Dawar and Erich Grädel, editors, *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018, Oxford, UK, July 09-12, 2018*, pages 36–45. ACM, 2018. doi:10.1145/3209108.3209185.
- 7 Ali Asadi, Krishnendu Chatterjee, Hongfei Fu, Amir Kafshdar Goharshady, and Mohammad Mahdavi. Polynomial reachability witnesses via stellensätze. In Stephen N. Freund and Eran Yahav, editors, *PLDI '21: 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation, Virtual Event, Canada, June 20-25, 2021*, pages 772–787. ACM, 2021. doi:10.1145/3453483.3454076.
- 8 Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT Press, 2008.
- 9 Roberto Baldoni, François Bonnet, Alessia Milani, and Michel Raynal. On the solvability of anonymous partial grids exploration by mobile robots. In Theodore P. Baker, Alain Bui, and Sébastien Tixeuil, editors, *Principles of Distributed Systems, 12th International Conference, OPODIS 2008, Luxor, Egypt, December 15-18, 2008. Proceedings*, volume 5401 of *Lecture Notes in Computer Science*, pages 428–445. Springer, 2008. doi:10.1007/978-3-540-92221-6_27.
- 10 Danièle Beauquier, Alexander Moshe Rabinovich, and Anatol Slissenko. A logic of probability with decidable model checking. *J. Log. Comput.*, 16(4):461–487, 2006. doi:10.1093/logcom/exl004.
- 11 Rohit Chadha, Vijay Anand Korthikanti, Mahesh Viswanathan, Gul Agha, and YoungMin Kwon. Model checking MDPs with a unique compact invariant set of distributions. In *Eighth International Conference on Quantitative Evaluation of Systems, QEST 2011, Aachen, Germany, 5-8 September, 2011*, pages 121–130. IEEE Computer Society, 2011. doi:10.1109/QEST.2011.22.
- 12 Aleksandar Chakarov and Sriram Sankaranarayanan. Probabilistic program analysis with martingales. In Natasha Sharygina and Helmut Veith, editors, *Computer Aided Verification*

- *25th International Conference, CAV 2013, Saint Petersburg, Russia, July 13-19, 2013. Proceedings*, volume 8044 of *Lecture Notes in Computer Science*, pages 511–526. Springer, 2013. doi:10.1007/978-3-642-39799-8_34.
- 13 Krishnendu Chatterjee, Hongfei Fu, and Amir Kafshdar Goharshady. Termination analysis of probabilistic programs through positivstellensatz's. In *CAV (1)*, volume 9779 of *Lecture Notes in Computer Science*, pages 3–22. Springer, 2016. doi:10.1007/978-3-319-41528-4_1.
- 14 Krishnendu Chatterjee, Hongfei Fu, Amir Kafshdar Goharshady, and Ehsan Kafshdar Goharshady. Polynomial invariant generation for non-deterministic recursive programs. In Alastair F. Donaldson and Emina Torlak, editors, *Proceedings of the 41st ACM SIGPLAN International Conference on Programming Language Design and Implementation, PLDI 2020, London, UK, June 15-20, 2020*, pages 672–687. ACM, 2020. doi:10.1145/3385412.3385969.
- 15 Krishnendu Chatterjee, Hongfei Fu, Petr Novotný, and Rouzbeh Hasheminezhad. Algorithmic analysis of qualitative and quantitative termination problems for affine probabilistic programs. *TOPLAS*, 40(2):7:1–7:45, 2018. doi:10.1145/3174800.
- 16 Krishnendu Chatterjee, Amir Kafshdar Goharshady, Ehsan Kafshdar Goharshady, Mehrdad Karrabi, Milad Saadat, Maximilian Seeliger, and Đorđe Žikelić. Polyqent: A polynomial quantified entailment solver, 2025. arXiv:2408.03796.
- 17 Krishnendu Chatterjee, Amir Kafshdar Goharshady, Ehsan Kafshdar Goharshady, Mehrdad Karrabi, and Đorđe Žikelić. Sound and complete witnesses for template-based verification of LTL properties on polynomial programs. In André Platzer, Kristin Yvonne Rozier, Matteo Pradella, and Matteo Rossi, editors, *Formal Methods - 26th International Symposium, FM 2024, Milan, Italy, September 9-13, 2024, Proceedings, Part I*, volume 14933 of *Lecture Notes in Computer Science*, pages 600–619. Springer, 2024. doi:10.1007/978-3-031-71162-6_31.
- 18 Krishnendu Chatterjee, Amir Kafshdar Goharshady, Tobias Meggendorfer, and Đorđe Žikelić. Sound and complete certificates for quantitative termination analysis of probabilistic programs. In *CAV (1)*, volume 13371 of *Lecture Notes in Computer Science*, pages 55–78. Springer, 2022. doi:10.1007/978-3-031-13185-1_4.
- 19 Krishnendu Chatterjee, Petr Novotný, and Đorđe Žikelić. Stochastic invariants for probabilistic termination. In *POPL*, pages 145–160, 2017. doi:10.1145/3009837.3009873.
- 20 Alessandro Cimatti, Alberto Griggio, Bastiaan Schaafsma, and Roberto Sebastiani. The MathSAT5 SMT Solver. In Nir Piterman and Scott Smolka, editors, *Proceedings of TACAS*, volume 7795 of *LNCS*. Springer, 2013. doi:10.1007/978-3-642-36742-7_7.
- 21 Michael Colón, Sriram Sankaranarayanan, and Henny Sipma. Linear invariant generation using non-linear constraint solving. In Warren A. Hunt Jr. and Fabio Somenzi, editors, *Computer Aided Verification, 15th International Conference, CAV 2003, Boulder, CO, USA, July 8-12, 2003, Proceedings*, volume 2725 of *Lecture Notes in Computer Science*, pages 420–432. Springer, 2003. doi:10.1007/978-3-540-45069-6_39.
- 22 Michael Colón and Henny Sipma. Synthesis of linear ranking functions. In Tiziana Margaria and Wang Yi, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 7th International Conference, TACAS 2001 Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2001 Genova, Italy, April 2-6, 2001, Proceedings*, volume 2031 of *Lecture Notes in Computer Science*, pages 67–81. Springer, 2001. doi:10.1007/3-540-45319-9_6.
- 23 Leonardo Mendonça de Moura and Nikolaj S. Bjørner. Z3: an efficient SMT solver. In *TACAS*, volume 4963 of *Lecture Notes in Computer Science*, pages 337–340. Springer, 2008. doi:10.1007/978-3-540-78800-3_24.
- 24 Alexandre Duret-Lutz, Etienne Renault, Maximilien Colange, Florian Renkin, Alexandre Gbaguidi Aisse, Philipp Schlehuber-Caissier, Thomas Medioni, Antoine Martin, Jérôme Dubois, Clément Gillard, et al. From spot 2.0 to spot 2.10: What's new? In *International Conference on Computer Aided Verification*, pages 174–187. Springer, 2022.
- 25 Julius Farkas. Theorie der einfachen ungleichungen. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 1902(124):1–27, 1902.

- 26 Yulong Gao, Alessandro Abate, Lihua Xie, and Karl Henrik Johansson. Distributional reachability for Markov decision processes: Theory and applications. *IEEE Trans. Autom. Control.*, 69(7):4598–4613, 2024. doi:10.1109/TAC.2023.3341282.
- 27 David Handelman. Representing polynomials by positive linear functions on compact convex polyhedra. *Pacific Journal of Mathematics*, 132(1):35–62, 1988.
- 28 Thomas A. Henzinger, Maria Mateescu, and Verena Wolf. Sliding window abstraction for infinite Markov chains. In Ahmed Bouajjani and Oded Maler, editors, *Computer Aided Verification, 21st International Conference, CAV 2009, Grenoble, France, June 26 - July 2, 2009. Proceedings*, volume 5643 of *Lecture Notes in Computer Science*, pages 337–352. Springer, 2009. doi:10.1007/978-3-642-02658-4_27.
- 29 Vijay Anand Korthikanti, Mahesh Viswanathan, Gul Agha, and YoungMin Kwon. Reasoning about MDPs as transformers of probability distributions. In *QEST 2010, Seventh International Conference on the Quantitative Evaluation of Systems, Williamsburg, Virginia, USA, 15-18 September 2010*, pages 199–208. IEEE Computer Society, 2010. doi:10.1109/QEST.2010.35.
- 30 Marta Z. Kwiatkowska, Gethin Norman, and David Parker. PRISM 4.0: Verification of probabilistic real-time systems. In *CAV*, volume 6806 of *Lecture Notes in Computer Science*, pages 585–591. Springer, 2011. doi:10.1007/978-3-642-22110-1_47.
- 31 YoungMin Kwon and Gul A. Agha. Verifying the evolution of probability distributions governed by a DTMC. *IEEE Trans. Software Eng.*, 37(1):126–141, 2011. doi:10.1109/TSE.2010.80.
- 32 Joël Ouaknine and James Worrell. Decision problems for linear recurrence sequences. In Alain Finkel, Jérôme Leroux, and Igor Potapov, editors, *Reachability Problems - 6th International Workshop, RP 2012, Bordeaux, France, September 17-19, 2012. Proceedings*, volume 7550 of *Lecture Notes in Computer Science*, pages 21–28. Springer, 2012. doi:10.1007/978-3-642-33512-9_3.
- 33 Stephen Prajna, Ali Jadbabaie, and George J. Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Trans. Autom. Control.*, 52(8):1415–1428, 2007. doi:10.1109/TAC.2007.902736.
- 34 Toru Takisaka, Yuichiro Oyabu, Natsuki Urabe, and Ichiro Hasuo. Ranking and repulsing supermartingales for reachability in randomized programs. *ACM Trans. Program. Lang. Syst.*, 43(2):5:1–5:46, 2021. doi:10.1145/3450967.
- 35 Dorde Zikelić, Mathias Lechner, Thomas A. Henzinger, and Krishnendu Chatterjee. Learning control policies for stochastic systems with reach-avoid guarantees. In Brian Williams, Yiling Chen, and Jennifer Neville, editors, *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 11926–11935. AAAI Press, 2023. doi:10.1609/AAAI.V37I10.26407.