Markov Automata with Multiple Objectives

Tim Quatmann(✉), Sebastian Junges,
and Joost-Pieter Katoen

RWTH Aachen University, Aachen, Germany
tim.quatmann@cs.rwth-aachen.de

Abstract. Markov automata combine non-determinism, probabilistic branching, and exponentially distributed delays. This compositional variant of continuous-time Markov decision processes is used in reliability engineering, performance evaluation and stochastic scheduling. Their verification so far focused on single objectives such as (timed) reachability, and expected costs. In practice, often the objectives are mutually dependent and the aim is to reveal trade-offs. We present algorithms to analyze several objectives simultaneously and approximate Pareto curves. This includes, e.g., several (timed) reachability objectives, or various expected cost objectives. We also consider combinations thereof, such as on-time-within-budget objectives—which policies guarantee reaching a goal state within a deadline with at least probability p while keeping the allowed average costs below a threshold? We adopt existing approaches for classical Markov decision processes. The main challenge is to treat policies exploiting state residence times, even for *un*timed objectives. Experimental results show the feasibility and scalability of our approach.

1 Introduction

Markov automata [1,2] extend labeled transition systems with probabilistic branching and exponentially distributed delays. They are a compositional variant of continuous-time Markov decision processes (CTMDPs), in a similar vein as Segala's probabilistic automata extend classical MDPs. Transitions of a Markov automaton (MA) lead from states to probability distributions over states, and are either labeled with actions (allowing for interaction) or real numbers (rates of exponential distributions). MAs are used in reliability engineering [3], hardware design [4], data-flow computation [5], dependability [6] and performance evaluation [7], as MAs are a natural semantic framework for modeling formalisms such as AADL, dynamic fault trees, stochastic Petri nets, stochastic activity networks, SADF etc. The verification of MAs so far focused on single objectives such as reachability, timed reachability, expected costs, and long-run averages [8–12]. These analyses cannot treat objectives that are mutually influencing each other, like quickly reaching a target is more costly. The aim of this paper is to analyze *multiple* objectives on MAs at once and to facilitate *trade-off analysis* by approximating Pareto curves.

© Springer International Publishing AG 2017
R. Majumdar and V. Kunčak (Eds.): CAV 2017, Part I, LNCS 10426, pp. 140–159, 2017.
DOI: 10.1007/978-3-319-63387-9_7

Consider the stochastic job scheduling problem of [13]: perform n jobs with exponential service times on k identical processors under a pre-emptive scheduling policy. Once a job finishes, all k processors can be assigned any of the m remaining jobs. When $n - m$ jobs are finished, this yields $\binom{m}{k}$ non-deterministic choices.

The largest-expected-service-time-first-policy is optimal to minimize the expected time to complete all jobs [13]. It is unclear how to schedule when imposing *extra* constraints, e.g., requiring a high probability to finish a batch of c jobs within a tight deadline (to accelerate their post-processing), or having a low average waiting time. These *multiple objectives* involve non-trivial *trade-offs*. Our algorithms analyze such trade-offs. Figure 1, e.g., shows the obtained result for 12 jobs and 3 processors. It approximates the set of points (p_1, p_2) for schedules achieving that (1) the expected time to complete all jobs is at most p_1 and (2) the probability to finish half of the jobs within an hour is at least p_2.

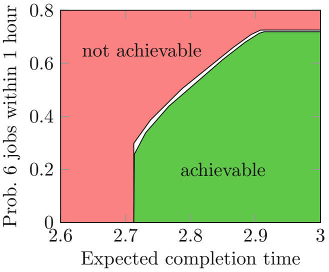
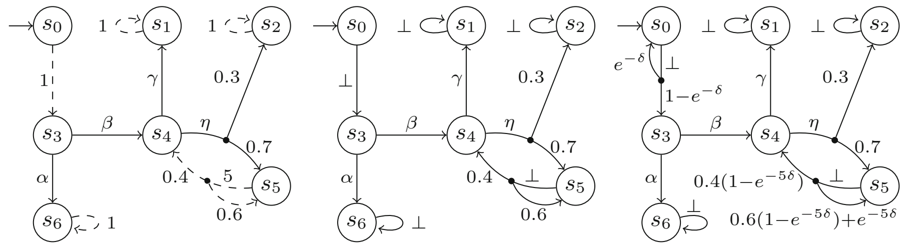


Fig. 1. Approx. Pareto curve for stochastic job scheduling.

This paper presents techniques to verify MAs with multiple objectives. We consider multiple (un)timed reachability and expected reward objectives as well as their combinations. Put shortly, we reduce all these problems to instances of multi-objective verification problems on classical MDPs. For multi-objective queries involving (combinations of) untimed reachability and expected reward objectives, corresponding algorithms on the *underlying* MDP can be used. In this case, the MDP is simply obtained by ignoring the timing information, see Fig. 2(b). The crux is in relating MA schedulers—that can exploit state sojourn times to optimize their decisions—to MDP schedulers. For multiple timed reachability objectives, *digitization* [8,9] is employed to obtain an MDP, see Fig. 2(c). The key is to mimic sojourn times by self-loops with appropriate probabilities. This provides a sound arbitrary close approximation of the timed behavior and also allows to combine timed reachability objectives with other types of objectives. The main contribution is to show that digitization is sound for *all* possible MA schedulers. This requires a new proof strategy as the existing ones are tailored to optimizing a single objective. All proofs can be found in an extended version [14]. Experiments on instances of four MA benchmarks show encouraging results. Multiple untimed reachability and expected reward objectives can be efficiently treated for models with millions of states. As for single objectives [9], timed reachability is more expensive. Our implementation is competitive to `PRISM` for multi-objective MDPs [15,16] and to `IMCA` [9] for single-objective MAs.

Related Work. Multi-objective decision making for MDPs with discounting and long-run objectives has been well investigated; for a recent survey, see [17]. Etessami *et al.* [18] consider verifying finite MDPs with multiple ω-regular objectives. Other multiple objectives include expected rewards under worst-case



(a) MA $\mathcal{M}$. (b) Underlying MDP $\mathcal{M}_\mathcal{D}$. (c) Digitization $\mathcal{M}_\delta$.

Fig. 2. MA $\mathcal{M}$ with underlying MDP $\mathcal{M}_\mathcal{D}$ and digitization $\mathcal{M}_\delta$.

reachability [19,20], quantiles and conditional probabilities [21], mean pay-offs and stability [22], long-run objectives [23,24], total average discounted rewards under PCTL [25], and stochastic shortest path objectives [26]. This has been extended to MDPs with unknown cost function [27], infinite-state MDPs [28] arising from two-player timed games in a stochastic environment, and stochastic two-player games [29]. To the best of our knowledge, this is the first work on multi-objective MDPs extended with *random timing*.

2 Preliminaries

Notations. The set of real numbers is denoted by $\mathbb{R}$, and we write $\mathbb{R}_{>0} = \{x \in \mathbb{R} \mid x > 0\}$ and $\mathbb{R}_{\geq 0} = \mathbb{R}_{>0} \cup \{0\}$. For a finite set S, $Dist(S)$ denotes the set of probability distributions over S. $\mu \in Dist(S)$ is *Dirac* if $\mu(s) = 1$ for some $s \in S$.

2.1 Models

Markov automata generalize both Markov decision processes (MDPs) and continuous time Markov chains (CTMCs). They are extended with rewards (or, equivalently, costs) to allow modelling, e.g., energy consumption.

Definition 1 (Markov automaton). *A* Markov automaton (MA) *is a tuple* $\mathcal{M} = (S, Act, \rightarrow, s_0, \{\rho_1, \dots, \rho_\ell\})$ *where* S *is a finite set of* states *with* initial state $s_0 \in S$, Act *is a finite set of* actions *with* $\perp \in Act$ *and* $Act \cap \mathbb{R}_{\geq 0} = \emptyset$,

- $\rightarrow \subseteq S \times (Act \uplus \mathbb{R}_{>0}) \times Dist(S)$ *is a set of* transitions *such that for all* $s \in S$ *there is at most one transition* $(s, \lambda, \mu) \in \rightarrow$ *with* $\lambda \in \mathbb{R}_{>0}$, *and*
- $\rho_1, \dots, \rho_\ell$ *with* $\ell \geq 0$ *are* reward functions $\rho_i \colon S \uplus (S \times Act) \rightarrow \mathbb{R}_{\geq 0}$.

In the remainder of the paper, let $\mathcal{M} = (S, Act, \rightarrow, s_0, \{\rho_1, \dots, \rho_\ell\})$ denote an MA. A transition $(s, \gamma, \mu) \in \rightarrow$, denoted by $s \xrightarrow{\gamma} \mu$, is called *probabilistic* if $\gamma \in Act$ and *Markovian* if $\gamma \in \mathbb{R}_{>0}$. In the latter case, γ is the rate of an exponential distribution, modeling a time-delayed transition. Probabilistic transitions fire instantaneously. The successor state is determined by μ, i.e., we move to s' with probability $\mu(s')$. Probabilistic (Markovian) states PS (MS) have an outgoing probabilistic (Markovian) transition, respectively: $\mathrm{PS} = \{s \in S \mid s \xrightarrow{\alpha} \mu, \alpha \in \mathit{Act}\}$ and $\mathrm{MS} = \{s \in S \mid s \xrightarrow{\lambda} \mu, \lambda \in \mathbb{R}_{>0}\}$. The *exit rate* $\mathrm{E}(s)$ of $s \in \mathrm{MS}$ is uniquely given by $s \xrightarrow{\mathrm{E}(s)} \mu$. The *transition probabilities* of $\mathcal{M}$ are given by the function $\mathbf{P} \colon S \times \mathit{Act} \times S \to [0,1]$ satisfying $\mathbf{P}(s,\alpha,s') = \mu(s')$ if either $s \xrightarrow{\alpha} \mu$ or $\big(\alpha = \perp$ and $s \xrightarrow{\mathrm{E}(s)} \mu\big)$ and $\mathbf{P}(s,\alpha,s') = 0$ in all other cases. The value $\mathbf{P}(s,\alpha,s')$ corresponds to the probability to move from s with action α to s'. The *enabled actions* at state s are given by $\mathit{Act}(s) = \{\alpha \in \mathit{Act} \mid \exists s' \in S \colon \mathbf{P}(s,\alpha,s') > 0\}$.

Example 1. Figure 2(a) shows an MA $\mathcal{M}$. We do not depict Dirac probability distributions. Markovian transitions are illustrated by dashed arrows.

We assume *action-deterministic* MAs: $|\{\mu \in \mathit{Dist}(S) \mid s \xrightarrow{\alpha} \mu\}| \leq 1$ holds for all $s \in S$ and $\alpha \in \mathit{Act}$. Terminal states $s \notin \mathrm{PS} \cup \mathrm{MS}$ are excluded by adding a Markovian self-loop. As standard for MAs [1,2], we impose the *maximal progress assumption*, i.e., probabilistic transitions take precedence over Markovian ones. Thus, we remove transitions $s \xrightarrow{\lambda} \mu$ for $s \in \mathrm{PS}$ and $\lambda \in \mathbb{R}_{>0}$ which yields $S = \mathrm{PS} \uplus \mathrm{MS}$. MAs with *Zeno behavior*, where infinitely many actions can be taken within finite time with non-zero probability, are unrealistic and considered a modeling error.

A reward function ρ_i defines *state rewards* and *action rewards*. When sojourning in a state s for t time units, the state reward $\rho_i(s) \cdot t$ is obtained. Upon taking a transition $s \xrightarrow{\gamma} \mu$, we collect action reward $\rho_i(s,\gamma)$ (if $\gamma \in \mathit{Act}$) or $\rho(s,\perp)$ (if $\gamma \in \mathbb{R}_{>0}$). For presentation purposes, in the remainder of this section, rewards are omitted. Full definitions with rewards can be found in [14].

Definition 2 (Markov decision process [30]**).** *A* Markov decision process *(MDP) is a tuple* $\mathcal{D} = (S, \mathit{Act}, \mathbf{P}, s_0, \emptyset)$ *with* S, s_0, Act *as in Definition 1 and* $\mathbf{P} \colon S \times \mathit{Act} \times S \to [0,1]$ *are the* transition probabilities *satisfying* $\sum_{s' \in S} \mathbf{P}(s,\alpha,s') \in \{0,1\}$ *for all* $s \in S$ *and* $\alpha \in \mathit{Act}$.

MDPs are MAs without Markovian states and thus without timing aspects, i.e., MDPs exhibit probabilistic branching and non-determinism. Zeno behavior is not a concern, as we do not consider timing aspects. The *underlying MDP* of an MA abstracts away from its timing:

Definition 3 (Underlying MDP). *The MDP* $\mathcal{M}_{\mathcal{D}} = (S, \mathit{Act}, \mathbf{P}, s_0, \emptyset)$ *is the* underlying MDP *of MA* $\mathcal{M} = (S, \mathit{Act}, \rightarrow, s_0, \emptyset)$ *with transition probabilities* $\mathbf{P}$.

The *digitization* $\mathcal{M}_\delta$ of $\mathcal{M}$ w.r.t. some digitization constant $\delta \in \mathbb{R}_{>0}$ is an MDP which digitizes the time [8,9]. The main difference between $\mathcal{M}_{\mathcal{D}}$ and $\mathcal{M}_\delta$ is that the latter also introduces *self-loops* which describe the probability to stay in a Markovian state for δ time units. More precisely, the outgoing transitions of states $s \in \mathrm{MS}$ in $\mathcal{M}_\delta$ represent that either (1) a Markovian transition in $\mathcal{M}$ was taken within δ time units, or (2) no transition is taken within δ time units – which is captured by taking the self-loop in $\mathcal{M}_\delta$. Counting the taken self-loops at $s \in \mathrm{MS}$ allows to approximate the sojourn time in s.

Definition 4 (Digitization of an MA). *For MA $\mathcal{M} = (S, Act, \rightarrow, s_0, \emptyset)$ with transition probabilities* $\mathbf{P}$ *and* digitization constant $\delta \in \mathbb{R}_{>0}$, *the* digitization of $\mathcal{M}$ *w.r.t.* δ *is the MDP* $\mathcal{M}_\delta = (S, Act, \mathbf{P}_\delta, s_0, \emptyset)$ *where*

$$\mathbf{P}_\delta(s, \alpha, s') = \begin{cases} \mathbf{P}(s, \bot, s') \cdot (1 - e^{-\mathrm{E}(s)\delta}) & \textit{if } s \in \mathrm{MS}, \alpha = \bot, s \neq s' \\ \mathbf{P}(s, \bot, s') \cdot (1 - e^{-\mathrm{E}(s)\delta}) + e^{-\mathrm{E}(s)\delta} & \textit{if } s \in \mathrm{MS}, \alpha = \bot, s = s' \\ \mathbf{P}(s, \alpha, s') & \textit{otherwise.} \end{cases}$$

Example 2. Figure 2 shows an MA $\mathcal{M}$ with its underlying MDP $\mathcal{M}_\mathcal{D}$ and a digitization $\mathcal{M}_\delta$ for unspecified $\delta \in \mathbb{R}_{>0}$.

Paths and Schedulers. Paths represent runs of $\mathcal{M}$ starting in the initial state. Let $t(\kappa) = 0$ and $\alpha(\kappa) = \kappa$, if $\kappa \in Act$, and $t(\kappa) = \kappa$ and $\alpha(\kappa) = \bot$, if $\kappa \in \mathbb{R}_{\geq 0}$.

Definition 5 (Infinite path). *An* infinite path *of MA* $\mathcal{M}$ *with transition probabilities* $\mathbf{P}$ *is an infinite sequence* $\pi = s_0 \xrightarrow{\kappa_0} s_1 \xrightarrow{\kappa_1} \dots$ *of states* $s_0, s_1, \dots \in S$ *and* stamps $\kappa_0, \kappa_1, \dots \in Act \uplus \mathbb{R}_{\geq 0}$ *such that (1)* $\sum_{i=0}^{\infty} t(\kappa_i) = \infty$, *and for any* $i \geq 0$ *it holds that (2)* $\mathbf{P}(s_i, \alpha(\kappa_i), s_{i+1}) > 0$, *(3)* $s_i \in \mathrm{PS}$ *implies* $\kappa_i \in Act$, *and (4)* $s_i \in \mathrm{MS}$ *implies* $\kappa_i \in \mathbb{R}_{\geq 0}$.

An infix $s_i \xrightarrow{\kappa_i} s_{i+1}$ of a path π represents that we stay at s_i for $t(\kappa_i)$ time units and then perform action $\alpha(\kappa_i)$ and move to state s_{i+1}. Condition (1) excludes Zeno paths, condition (2) ensures positive transition probabilities, and conditions (3) and (4) assert that stamps κ_i match the transition type at s_i.

A *finite path* is a finite prefix $\pi' = s_0 \xrightarrow{\kappa_0} \dots \xrightarrow{\kappa_{n-1}} s_n$ of an infinite path. The *length* of π' is $|\pi'| = n$, its *last state* is $last(\pi') = s_n$, and the *time duration* is $T(\pi') = \sum_{0 \leq i < |\pi'|} t(\kappa_i)$. We denote the sets of finite and infinite paths of $\mathcal{M}$ by $FPaths^\mathcal{M}$ and $IPaths^\mathcal{M}$, respectively. The superscript $\mathcal{M}$ is omitted if the model is clear from the context. For a finite or infinite path $\pi = s_0 \xrightarrow{\kappa_0} s_1 \xrightarrow{\kappa_1} \dots$ the *prefix* of π of length n is denoted by $pref(\pi, n)$. The ith state visited by π is given by $\pi[i] = s_i$. The *time-abstraction* $\mathrm{ta}(\pi)$ of π removes all sojourn times and is a path of the underlying MDP $\mathcal{M}_\mathcal{D}$: $\mathrm{ta}(\pi) = s_0 \xrightarrow{\alpha(\kappa_0)} s_1 \xrightarrow{\alpha(\kappa_1)} \dots$. Paths of $\mathcal{M}_\mathcal{D}$ are also referred to as the *time-abstract paths of* $\mathcal{M}$.

Definition 6 (Generic scheduler). *A* generic scheduler *for* $\mathcal{M}$ *is a measurable function* $\sigma \colon FPaths \times Act \rightarrow [0, 1]$ *such that* $\sigma(\pi, \cdot) \in Dist(Act(last(\pi)))$ *for each* $\pi \in FPaths$.

A scheduler σ for $\mathcal{M}$ resolves the non-determinism of $\mathcal{M}$: $\sigma(\pi, \alpha)$ is the probability to take transition $last(\pi) \xrightarrow{\alpha} \mu$ after observing the run π. The set of such schedulers is denoted by $\mathrm{GM}^\mathcal{M}$ (GM if $\mathcal{M}$ is clear from the context). $\sigma \in \mathrm{GM}$ is *deterministic* if the distribution $\sigma(\pi, \cdot)$ is Dirac for any π. *Time-abstract schedulers* behave independently of the time-stamps of the given path, i.e., $\sigma(\pi, \alpha) = \sigma(\pi', \alpha)$ for all actions α and paths π, π' with $\mathrm{ta}(\pi) = \mathrm{ta}(\pi')$. We write $\mathrm{TA}^\mathcal{M}$ to denote the set of time-abstract schedulers of $\mathcal{M}$. GM is the most general scheduler class for MAs. For MDPs, the most general scheduler class is TA.

2.2 Objectives

An objective $\mathbb{O}_i$ is a representation of a *quantitative* property like the probability to reach an error state, or the expected energy consumption. To express *Boolean* properties (e.g., the probability to reach an error state is below p_i), $\mathbb{O}_i$ is combined with a *threshold* $\rhd_i p_i$ where $\rhd_i \in \{<, \leq, >, \geq\}$ is a *threshold relation* and $p_i \in \mathbb{R}$ is a *threshold value*. Let $\mathcal{M}, \sigma \models \mathbb{O}_i \rhd_i p_i$ denote that the MA $\mathcal{M}$ under scheduler $\sigma \in \text{GM}$ satisfies the property $\mathbb{O}_i \rhd_i p_i$.

Reachability Objectives. $I \subseteq \mathbb{R}$ is a *time interval* if it is of the form $I = [a, b]$ or $I = [a, \infty)$, where $0 \leq a < b$. The set of paths reaching a set of goal states $G \subseteq S$ in time I is defined as

$$\Diamond^I G = \{\pi = s_0 \xrightarrow{\kappa_0} s_1 \xrightarrow{\kappa_1} \cdots \in \mathit{IPaths} \mid \exists n \geq 0\colon \pi[n] \in G \text{ and} \\ I \cap [t, t + t(\kappa_n)] \neq \emptyset \text{ for } t = T(\mathit{pref}(\pi, n))\}.$$

We write $\Diamond G$ instead of $\Diamond^{[0,\infty)} G$. A probability measure $\Pr_\sigma^\mathcal{M}$ on sets of infinite paths is defined, which generalizes both the standard probability measure on MDPs and on CTMCs. A formal definition is given in [14].

Definition 7 (Reachability objective). *A* reachability objective *has the form* $\mathbb{P}(\Diamond^I G)$ *for time interval* I *and goal states* G*. The objective is* timed *if* $I \neq [0, \infty)$ *and* untimed *otherwise. For MA* $\mathcal{M}$ *and scheduler* $\sigma \in \text{GM}$*, let* $\mathcal{M}, \sigma \models \mathbb{P}(\Diamond^I G) \rhd_i p_i$ *iff* $\Pr_\sigma^\mathcal{M}(\Diamond^I G) \rhd_i p_i$*.*

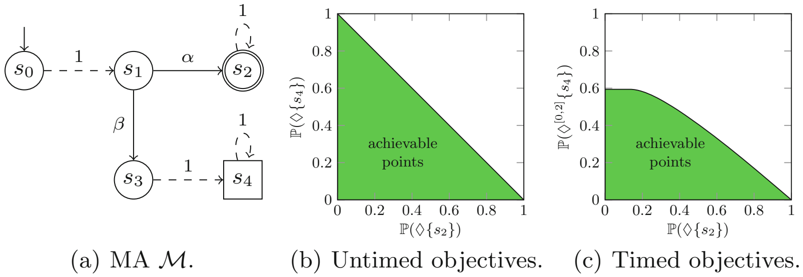
Expected Reward Objectives. Expected rewards $\text{ER}_\sigma^\mathcal{M}(\rho_j, G)$ define the expected amount of reward collected (w.r.t. ρ_j) until a goal state in $G \subseteq S$ is reached. This is a straightforward generalization of the notion on CTMCs and MDPs. A formal definition is found in [14].

Definition 8 (Expected reward objective). *An* expected reward objective *has the form* $\mathbb{E}(\#j, G)$ *where* j *is the index of reward function* ρ_j *and* $G \subseteq S$*. For MA* $\mathcal{M}$ *and scheduler* $\sigma \in \text{GM}$*, let* $\mathcal{M}, \sigma \models \mathbb{E}(\#j, G) \rhd_i p_i$ *iff* $\text{ER}_\sigma^\mathcal{M}(\rho_j, G) \rhd_i p_i$*.*

Expected *time* objectives $\mathbb{E}(T, G)$ are expected reward objectives that consider the reward function ρ_T with $\rho_T(s) = 1$ if $s \in \text{MS}$ and all other rewards are zero.

3 Multi-objective Model Checking

Standard model checking considers objectives individually. This approach is not feasible when we are interested in multiple objectives that should be fulfilled by the same scheduler, e.g., a scheduler that maximizes the expected profit might violate certain safety constraints. *Multi-objective* model checking aims to analyze multiple objectives at once and reveals possible trade-offs.



(a) MA $\mathcal{M}$. (b) Untimed objectives. (c) Timed objectives.

Fig. 3. Markov automaton and achievable points.

Definition 9 (Satisfaction of multiple objectives). *Let $\mathcal{M}$ be an MA and $\sigma \in \mathrm{GM}$. For objectives $\mathbb{O} = (\mathbb{O}_1, \dots, \mathbb{O}_d)$ with threshold relations $\rhd = (\rhd_1, \dots, \rhd_d) \in \{<, \leq, >, \geq\}^d$ and threshold values $\mathbf{p} = (p_1, \dots, p_d) \in \mathbb{R}^d$ let*

$$\mathcal{M}, \sigma \models \mathbb{O} \rhd \mathbf{p} \iff \mathcal{M}, \sigma \models \mathbb{O}_i \rhd_i p_i \textit{ for all } 1 \leq i \leq d.$$

Furthermore, let achieve$^{\mathcal{M}}(\mathbb{O} \rhd \mathbf{p}) \iff \exists \sigma \in \mathrm{GM}$ such that $\mathcal{M}, \sigma \models \mathbb{O} \rhd \mathbf{p}$.

If $\mathcal{M}, \sigma \models \mathbb{O} \rhd \mathbf{p}$, the point $\mathbf{p} \in \mathbb{R}^d$ is *achievable* in $\mathcal{M}$ with scheduler σ. The *set of achievable points* of $\mathcal{M}$ w.r.t. $\mathbb{O}$ and $\mathbf{p}$ is $\{\mathbf{p} \in \mathbb{R}^d \mid \textit{achieve}^{\mathcal{M}}(\mathbb{O} \rhd \mathbf{p})\}$. This definition is compatible with the notions on MDPs as given in [16,18].

Example 3. Figure 3(b) and (c) depict the set of achievable points of the MA $\mathcal{M}$ from Fig. 3(a) w.r.t. relations $\rhd = (\geq, \geq)$ and objectives $(\mathbb{P}(\Diamond\{s_2\}), \mathbb{P}(\Diamond\{s_4\}))$ and $(\mathbb{P}(\Diamond\{s_2\}), \mathbb{P}(\Diamond^{[0,2]}\{s_4\}))$, respectively. Using the set of achievable points, we can answer Pareto, numerical, and achievability queries as considered in [16], e.g., the Pareto front lies on the border of the set.

Schedulers. For single-objective model checking on MAs, it suffices to consider deterministic schedulers [31]. For untimed reachability and expected rewards even *time-abstract* deterministic schedulers suffice [31]. Multi-objective model checking on MDPs requires history-dependent, randomized schedulers [18]. On MAs, schedulers may also employ *timing* information to make optimal choices, even if only *untimed* objectives are considered.

Example 4. Consider the MA $\mathcal{M}$ in Fig. 3(a) with untimed objectives $\mathbb{P}(\Diamond\{s_2\}) \geq 0.5$ and $\mathbb{P}(\Diamond\{s_4\}) \geq 0.5$. A simple graph argument yields that both properties are only satisfied if action α is taken with probability exactly a half. Thus, on the underlying MDP, no deterministic scheduler satisfies both objectives. On the MA however, paths can be distinguished by their sojourn time in s_0. As the probability mass to stay in s_0 for at most $\ln(2)$ is exactly 0.5, a timed scheduler σ with $\sigma(s_0 \xrightarrow{t} s_1, \alpha) = 1$ if $t \leq \ln(2)$ and 0 otherwise does satisfy both objectives.

Theorem 1. *For some MA $\mathcal{M}$ with achieve$^{\mathcal{M}}(\mathbb{O} \rhd \mathbf{p})$, no deterministic time-abstract scheduler σ satisfies $\mathcal{M}, \sigma \models \mathbb{O} \rhd \mathbf{p}$.*

The Geometric Shape of the Achievable Points. Like for MDPs [18], the set of achievable points of any combination of aforementioned objectives is convex.

Proposition 1. *The set* $\{\mathbf{p} \in \mathbb{R}^d \mid \mathit{achieve}^{\mathcal{M}}(\mathbb{O} \rhd \mathbf{p})\}$ *is convex.*

For MDPs, the set of achievable points is a convex polytope where the vertices can be realized by deterministic schedulers that use memory bounded by the number of objectives. As there are finitely many such schedulers, the polytope is finite [18], i.e., it can be represented by a finite number of vertices. This result does not carry over to MAs. For example, the achievable points of the MA from Fig. 3(a) together with the objectives $(\mathbb{P}(\lozenge\{s_2\}), \mathbb{P}(\lozenge^{[0,2]}\{s_4\}))$ form the infinite polytope shown in Fig. 3(c). The insight here is that for any sojourn time $t \leq 2$ in s_0, the timing information is relevant for optimal schedulers: The shorter the sojourn time in s_0, the higher the probability to reach s_4 within the time bound.

Theorem 2. *For some MA* $\mathcal{M}$ *and objectives* $\mathbb{O}$*, the polytope* $\{\mathbf{p} \in \mathbb{R}^d \mid \mathit{achieve}^{\mathcal{M}}(\mathbb{O} \rhd \mathbf{p})\}$ *is not finite.*

As infinite convex polytopes cannot be represented by a finite number of vertices, any method extending the approach of [16] – which computes these vertices – can only approximate the set of achievable points.

Problem Statement. For an MA and objectives with threshold relations, construct arbitrarily tight over- and under-approximations of the achievable points.

4 Analysis of Markov Automata with Multiple Objectives

The state-of-the-art in single-objective model checking of MA is to reduce the MA to an MDP, cf. [8–10], for which efficient algorithms exist. We aim to lift this approach to multi-objective model checking. Assume MA $\mathcal{M}$ and objectives $\mathbb{O}$ with threshold relations $\rhd$. We discuss how the set of achievable points of $\mathcal{M}$ relates to the set of achievable points of an MDP. The key challenge is to deal with timing information—even for *un*timed objectives—and to consider schedulers beyond those optimizing single objectives. We obtain:

- For untimed reachability and expected reward objectives, the achievable points of $\mathcal{M}$ *equal* those of its *underlying* MDP, cf. Theorems 3 and 4.
- For timed reachability objectives, the set of achievable points of a *digitized* MDP $\mathcal{M}_\delta$ provides a *sound approximation* of the achievable points of $\mathcal{M}$, cf. Theorem 5. Corollary 1 gives the precision of the approximation.

4.1 Untimed Reachability Objectives

Although timing information is essential for *deterministic* schedulers, cf. Theorem 1, timing information does not strengthen randomized schedulers:

Theorem 3. *For MA* $\mathcal{M}$ *and untimed reachability objectives* $\mathbb{O}$ *it holds that* $\mathit{achieve}^{\mathcal{M}}(\mathbb{O} \rhd \mathbf{p}) \iff \mathit{achieve}^{\mathcal{M}_\mathcal{D}}(\mathbb{O} \rhd \mathbf{p})$.

The main idea for proving Theorem 3 is to construct for scheduler $\sigma \in \mathrm{GM}^{\mathcal{M}}$ a time-abstract scheduler $\mathrm{ta}(\sigma) \in \mathrm{TA}^{\mathcal{M}_{\mathcal{D}}}$ such that they both induce the same untimed reachability probabilities. To this end, we discuss the connection between probabilities of paths of MA $\mathcal{M}$ and paths of MDP $\mathcal{M}_{\mathcal{D}}$.

Definition 10 (Induced paths of a time-abstract path). *The set of* induced paths on MA $\mathcal{M}$ *of a path* $\hat{\pi}$ *of* $\mathcal{M}_{\mathcal{D}}$ *is given by*

$$\langle \hat{\pi} \rangle = \mathrm{ta}^{-1}(\hat{\pi}) = \{\pi \in \mathit{FPaths}^{\mathcal{M}} \cup \mathit{IPaths}^{\mathcal{M}} \mid \mathrm{ta}(\pi) = \hat{\pi}\}.$$

The set $\langle \hat{\pi} \rangle$ contains all paths of $\mathcal{M}$ where replacing sojourn times by $\perp$ yields $\hat{\pi}$. For $\sigma \in \mathrm{GM}$, the probability distribution $\sigma(\pi, \cdot) \in \mathit{Dist}(\mathit{Act})$ might depend on the sojourn times of the path π. The time-abstract scheduler $\mathrm{ta}(\sigma)$ weights the distribution $\sigma(\pi, \cdot)$ with the probability masses of the paths $\pi \in \langle \hat{\pi} \rangle$.

Definition 11 (Time-abstraction of a scheduler). *The time-abstraction of* $\sigma \in \mathrm{GM}^{\mathcal{M}}$ *is defined as* $\mathrm{ta}(\sigma) \in \mathrm{TA}^{\mathcal{M}_{\mathcal{D}}}$ *such that for any* $\hat{\pi} \in \mathit{FPaths}^{\mathcal{M}_{\mathcal{D}}}$

$$\mathrm{ta}(\sigma)(\hat{\pi}, \alpha) = \int_{\pi \in \langle \hat{\pi} \rangle} \sigma(\pi, \alpha) \, \mathrm{dPr}_{\sigma}^{\mathcal{M}}(\pi \mid \langle \hat{\pi} \rangle).$$

The term $\mathrm{Pr}_{\sigma}^{\mathcal{M}}(\pi \mid \langle \hat{\pi} \rangle)$ represents the probability for a path in $\langle \hat{\pi} \rangle$ to have sojourn times as given by π. The value $\mathrm{ta}(\sigma)(\hat{\pi}, \alpha)$ coincides with the probability that σ picks action α, given that the time-abstract path $\hat{\pi}$ was observed.

Example 5. Consider the MA $\mathcal{M}$ in Fig. 2(a) and the scheduler σ choosing α at state s_3 iff the sojourn time at s_0 is at most one. Then $\mathrm{ta}(\sigma)(s_0 \xrightarrow{\perp} s_3, \alpha) = 1 - e^{-\mathrm{E}(s_0)}$, the probability that s_0 is left within one time unit. For $\bar{\pi} = s_0 \xrightarrow{\perp} s_3 \xrightarrow{\alpha} s_6$ we have

$$\mathrm{Pr}_{\sigma}^{\mathcal{M}}(\Diamond\{s_6\}) = \mathrm{Pr}_{\sigma}^{\mathcal{M}}(\langle \bar{\pi} \rangle) = 1 - e^{-\mathrm{E}(s_0)} = \mathrm{Pr}_{\mathrm{ta}(\sigma)}^{\mathcal{M}_{\mathcal{D}}}(\bar{\pi}) = \mathrm{Pr}_{\mathrm{ta}(\sigma)}^{\mathcal{M}_{\mathcal{D}}}(\Diamond\{s_6\}).$$

In the example, the considered scheduler and its time-abstraction induce the same untimed reachability probabilities. We generalize this observation.

Lemma 1. *For any* $\hat{\pi} \in \mathit{FPaths}^{\mathcal{M}_{\mathcal{D}}}$ *we have* $\mathrm{Pr}_{\sigma}^{\mathcal{M}}(\langle \hat{\pi} \rangle) = \mathrm{Pr}_{\mathrm{ta}(\sigma)}^{\mathcal{M}_{\mathcal{D}}}(\hat{\pi})$.

The result is lifted to untimed reachability probabilities.

Proposition 2. *For any* $G \subseteq S$ *it holds that* $\mathrm{Pr}_{\sigma}^{\mathcal{M}}(\Diamond G) = \mathrm{Pr}_{\mathrm{ta}(\sigma)}^{\mathcal{M}_{\mathcal{D}}}(\Diamond G)$.

As the definition of $\mathrm{ta}(\sigma)$ is independent of the considered set of goal states $G \subseteq S$, Proposition 2 can be lifted to multiple untimed reachability objectives.

Proof of Theorem 3 (sketch). By applying Proposition 2, we can show that $\mathcal{M}, \sigma \models \mathbb{O} \rhd \mathbf{p} \iff \mathcal{M}_{\mathcal{D}}, \mathrm{ta}(\sigma) \models \mathbb{O} \rhd \mathbf{p}$ for any scheduler $\sigma \in \mathrm{GM}^{\mathcal{M}}$ and untimed reachability objectives $\mathbb{O} = (\mathbb{P}(\Diamond G_1), \dots, \mathbb{P}(\Diamond G_d))$ with thresholds $\rhd \mathbf{p}$. Theorem 3 is a direct consequence of this.

4.2 Expected Reward Objectives

The results for expected reward objectives are similar to untimed reachability objectives: An analysis of the underlying MDP suffices. We show the following extension of Theorem 3 to expected reward objectives.

Theorem 4. *For MA $\mathcal{M}$ and untimed reachability and expected reward objectives $\mathbb{O}$: $\mathit{achieve}^{\mathcal{M}}(\mathbb{O} \rhd \mathbf{p}) \iff \mathit{achieve}^{\mathcal{M}_{\mathcal{D}}}(\mathbb{O} \rhd \mathbf{p})$.*

To prove this, we show that a scheduler $\sigma \in \mathrm{GM}^{\mathcal{M}}$ and its time-abstraction $\mathrm{ta}(\sigma) \in \mathrm{TA}$ induce the same expected rewards on $\mathcal{M}$ and $\mathcal{M}_{\mathcal{D}}$, respectively. Theorem 4 follows then analogously to Theorem 3.

Proposition 3. *Let ρ be some reward function of $\mathcal{M}$ and let $\rho^{\mathcal{D}}$ be its counterpart for $\mathcal{M}_{\mathcal{D}}$. For $G \subseteq S$ we have $\mathrm{ER}^{\mathcal{M}}_{\sigma}(\rho, G) = \mathrm{ER}^{\mathcal{M}_{\mathcal{D}}}_{\mathrm{ta}(\sigma)}(\rho^{\mathcal{D}}, G)$.*

Notice that $\rho^{\mathcal{D}}$ encodes the *expected* reward of $\mathcal{M}$ obtained in a state s by assuming the sojourn time to be the expected sojourn time $1/E(s)$. Although the claim is similar to Proposition 2, its proof cannot be adapted straightforwardly. In particular, the analogon to Lemma 1 does not hold: The expected reward collected along a time-abstract path $\hat{\pi} \in \mathit{FPaths}^{\mathcal{M}_{\mathcal{D}}}$ does in general not coincide for $\mathcal{M}$ and $\mathcal{M}_{\mathcal{D}}$.

Example 6. We consider standard notations for rewards as detailed in [14]. Let $\mathcal{M}$ be the MA with underlying MDP $\mathcal{M}_{\mathcal{D}}$ as shown in Fig. 2. Let $\rho(s_0) = 1$ and zero otherwise. Reconsider the scheduler σ from Example 5. Let $\hat{\pi}_\alpha = s_0 \xrightarrow{\perp} s_3 \xrightarrow{\alpha} s_6$. The probability $\mathrm{Pr}^{\mathcal{M}}_{\sigma}(\{s_0 \xrightarrow{t} s_3 \xrightarrow{\alpha} s_6 \in \langle\hat{\pi}_\alpha\rangle \mid t > 1\})$ is zero since σ chooses β on such paths. For the remaining paths in $\langle\hat{\pi}_\alpha\rangle$, action α is chosen with probability one. The expected reward in $\mathcal{M}$ along $\hat{\pi}_\alpha$ is:

$$\int_{\pi \in \langle\hat{\pi}_\alpha\rangle} \mathit{rew}^{\mathcal{M}}(\rho, \pi) \,\mathrm{d}\mathrm{Pr}^{\mathcal{M}}_{\sigma}(\pi) = \int_0^1 \rho(s_0) \cdot t \cdot \mathrm{E}(s_0) \cdot e^{-\mathrm{E}(s_0)t} \,\mathrm{d}t = 1 - 2e^{-1}.$$

The expected reward in $\mathcal{M}_{\mathcal{D}}$ along $\hat{\pi}_\alpha$ differs as

$$\mathit{rew}^{\mathcal{M}_{\mathcal{D}}}(\rho^{\mathcal{D}}, \hat{\pi}_\alpha) \cdot \mathrm{Pr}^{\mathcal{M}_{\mathcal{D}}}_{\mathrm{ta}(\sigma)}(\hat{\pi}_\alpha) = \rho^{\mathcal{D}}(s_0, \perp) \cdot \mathrm{ta}(\sigma)(s_0 \xrightarrow{\perp} s_3, \alpha) = 1 - e^{-1}.$$

The intuition is as follows: If path $s_0 \xrightarrow{t} s_3 \xrightarrow{\alpha} s_6$ of $\mathcal{M}$ under σ occurs, we have $t \leq 1$ since σ chose α. Hence, the reward collected from paths in $\langle\hat{\pi}_\alpha\rangle$ is at most $1 \cdot \rho(s_0) = 1$. There is thus a dependency between the choice of the scheduler at s_3 and the collected reward at s_0. This dependency is absent in $\mathcal{M}_{\mathcal{D}}$ as the reward at a state is independent of the subsequent performed actions.

Let $\hat{\pi}_\beta = s_0 \xrightarrow{\perp} s_3 \xrightarrow{\beta} s_4$. The expected reward along $\hat{\pi}_\beta$ is $2e^{-1}$ for $\mathcal{M}$ and e^{-1} for $\mathcal{M}_{\mathcal{D}}$. As the rewards for $\hat{\pi}_\alpha$ and $\hat{\pi}_\beta$ sum up to one in both $\mathcal{M}$ and $\mathcal{M}_{\mathcal{D}}$, the expected reward along all paths of length two coincides for $\mathcal{M}$ and $\mathcal{M}_{\mathcal{D}}$.

This observation can be generalized to arbitrary MA and paths of arbitrary length.

Proof of Proposition 3 (sketch). For every $n \geq 0$, the expected reward collected along paths of length at most n coincides for $\mathcal{M}$ under σ and $\mathcal{M}_\mathcal{D}$ under $\mathrm{ta}(\sigma)$. The proposition follows by letting n approach infinity.

Thus, queries on MA with mixtures of untimed reachability and expected reward objectives can be analyzed on the underlying MDP $\mathcal{M}_\mathcal{D}$.

4.3 Timed Reachability Objectives

Timed reachability objectives cannot be analyzed on $\mathcal{M}_\mathcal{D}$ as it abstracts away from sojourn times. We lift the digitization approach for single-objective timed reachability [8,9] to multiple objectives. Instead of abstracting timing information, it is *digitized*. Let $\mathcal{M}_\delta$ denote the digitization of $\mathcal{M}$ for arbitrary digitization constant $\delta \in \mathbb{R}_{>0}$, see Definition 4. A time interval $I \subseteq \mathbb{R}_{\geq 0}$ of the form $[a, \infty)$ or $[a, b]$ with $\mathrm{di}_a := {}^a/_\delta \in \mathbb{N}$ and $\mathrm{di}_b := {}^b/_\delta \in \mathbb{N}$ is called *well-formed*. For the remainder, we only consider well-formed intervals, ensured by an appropriate digitization constant. An interval for time-bounds I is transformed to digitization step bounds $\mathrm{di}(I) \subseteq \mathbb{N}$. Let $a = \inf I$, we set $\mathrm{di}(I) = \{{}^t/_\delta \in \mathbb{N} \mid t \in I\} \setminus \{0 \mid a > 0\}$.

We first relate paths in $\mathcal{M}$ to paths in its digitization.

Definition 12 (Digitization of a path). *The* digitization $\mathrm{di}(\pi)$ *of path* $\pi = s_0 \xrightarrow{\kappa_0} s_1 \xrightarrow{\kappa_1} \dots$ *in* $\mathcal{M}$ *is the path in* $\mathcal{M}_\delta$ *given by*

$$\mathrm{di}(\pi) = \big(s_0 \xrightarrow{\alpha(\kappa_0)}\big)^{m_0} s_0 \xrightarrow{\alpha(\kappa_0)} \big(s_1 \xrightarrow{\alpha(\kappa_1)}\big)^{m_1} s_1 \xrightarrow{\alpha(\kappa_1)} \dots$$

where $m_i = \max\{m \in \mathbb{N} \mid m\delta \leq t(\kappa_i)\}$ *for each* $i \geq 0$.

Example 7. For the path $\pi = s_0 \xrightarrow{1.1} s_3 \xrightarrow{\beta} s_4 \xrightarrow{\eta} s_5 \xrightarrow{0.3} s_4$ of the MA $\mathcal{M}$ in Fig. 2(a) and $\delta = 0.4$, we get $\mathrm{di}(\pi) = s_0 \xrightarrow{\perp} s_0 \xrightarrow{\perp} s_0 \xrightarrow{\perp} s_3 \xrightarrow{\beta} s_4 \xrightarrow{\eta} s_5 \xrightarrow{\perp} s_4$.

The m_i in the definition above represent a digitization of the sojourn times $t(\kappa_i)$ such that $m_i\delta \leq t(\kappa_i) < (m_i{+}1)\delta$. These digitized times are incorporated into the digitization of a path by taking the self-loop at state $s_i \in \mathrm{MS}$ m_i times. We also refer to the paths of $\mathcal{M}_\delta$ as *digital paths (of* $\mathcal{M}$*)*. The number $|\bar{\pi}|_{\mathrm{ds}}$ of *digitization steps* of a digital path $\bar{\pi}$ is the number of transitions emerging from Markovian states, i.e., $|\bar{\pi}|_{\mathrm{ds}} = |\{i < |\bar{\pi}| \mid \bar{\pi}[i] \in \mathrm{MS}\}|$. One digitization step represents the elapse of at most δ time units—either by staying at some $s \in \mathrm{MS}$ for δ time or by leaving s within δ time. The number $|\mathrm{di}(\pi)|_{\mathrm{ds}}$ multiplied with δ yields an estimate for the duration $T(\pi)$. A digital path $\bar{\pi}$ can be interpreted as representation of the set of paths of $\mathcal{M}$ whose digitization is $\bar{\pi}$.

Definition 13 (Induced paths of a digital path). *The set of* induced paths *of a (finite or infinite) digital path* $\bar{\pi}$ *of* $\mathcal{M}_\delta$ *is*

$$[\bar{\pi}] = \mathrm{di}^{-1}(\bar{\pi}) = \{\pi \in \mathit{FPaths}^{\mathcal{M}} \cup \mathit{IPaths}^{\mathcal{M}} \mid \mathrm{di}(\pi) = \bar{\pi}\}.$$

For sets of digital paths Π we define the *induced paths* $[\Pi] = \bigcup_{\bar{\pi} \in \Pi} [\bar{\pi}]$. To relate timed reachability probabilities for $\mathcal{M}$ under scheduler $\sigma \in \mathrm{GM}^{\mathcal{M}}$ with ds-bounded reachability probabilities for $\mathcal{M}_\delta$, relating σ to a scheduler for $\mathcal{M}_\delta$ is necessary.

Definition 14 (Digitization of a scheduler). *The* digitization *of* $\sigma \in \mathrm{GM}^{\mathcal{M}}$ *is given by* $\mathrm{di}(\sigma) \in \mathrm{TA}^{\mathcal{M}_\delta}$ *such that for any* $\bar{\pi} \in \mathit{FPaths}^{\mathcal{M}_\delta}$ *with* $\mathit{last}(\bar{\pi}) \in \mathrm{PS}$

$$\mathrm{di}(\sigma)(\bar{\pi}, \alpha) = \int_{\pi \in [\bar{\pi}]} \sigma(\pi, \alpha) \, \mathrm{d}\mathrm{Pr}_{\sigma}^{\mathcal{M}}(\pi \mid [\bar{\pi}]).$$

The digitization $\mathrm{di}(\sigma)$ is similar to the time-abstraction $\mathrm{ta}(\sigma)$ as both schedulers get a path with restricted timing information as input and mimic the choice of σ. However, while $\mathrm{ta}(\sigma)$ receives no information regarding sojourn times, $\mathrm{di}(\sigma)$ receives the digital estimate. Intuitively, $\mathrm{di}(\sigma)(\bar{\pi}, \alpha)$ considers $\sigma(\pi, \alpha)$ for each $\pi \in [\bar{\pi}]$, weighted with the probability that the sojourn times of a path in $[\bar{\pi}]$ are as given by π. The restriction $\mathit{last}(\bar{\pi}) \in \mathrm{PS}$ asserts that $\bar{\pi}$ does not end with a self-loop on a Markovian state, implying $[\bar{\pi}] \neq \emptyset$.

Example 8. Let MA $\mathcal{M}$ in Fig. 2(a) and $\delta = 0.4$. Again, $\sigma \in \mathrm{GM}^{\mathcal{M}}$ chooses α at state s_3 iff the sojourn time at s_0 is at most one. Consider the digital paths $\bar{\pi}_m = (s_0 \xrightarrow{\perp})^m s_0 \xrightarrow{\perp} s_3$. For $\pi \in [\bar{\pi}_1] = \{s_0 \xrightarrow{t} s_3 \mid 0.4 \leq t < 0.8\}$ we have $\sigma(\pi, \alpha) = 1$. It follows $\mathrm{di}(\sigma)(\pi_1, \alpha) = 1$. For $\pi \in [\bar{\pi}_2] = \{s_0 \xrightarrow{t} s_3 \mid 0.8 \leq t < 1.2\}$ it is unclear whether σ chooses α or β. Hence, $\mathrm{di}(\sigma)$ randomly guesses:

$$\mathrm{di}(\sigma)(\bar{\pi}_2, \alpha) = \int_{\pi \in [\bar{\pi}_2]} \sigma(\pi, \alpha) \, \mathrm{d}\mathrm{Pr}_{\sigma}^{\mathcal{M}}(\pi \mid [\bar{\pi}_2]) = \frac{\int_{0.8}^{1.0} \mathrm{E}(s_0) e^{-\mathrm{E}(s_0) t} \, \mathrm{d}t}{\int_{0.8}^{1.2} \mathrm{E}(s_0) e^{-\mathrm{E}(s_0) t} \, \mathrm{d}t} \approx 0.55.$$

On $\mathcal{M}_\delta$ we consider ds-bounded reachability instead of timed reachability.

Definition 15 (ds-bounded reachability). *The set of infinite digital paths that reach* $G \subseteq S$ *within the interval* $J \subseteq \mathbb{N}$ *of consecutive natural numbers is*

$$\Diamond_{\mathrm{ds}}^{J} G = \{\bar{\pi} \in \mathit{IPaths}^{\mathcal{M}_\delta} \mid \exists n \geq 0 \colon \bar{\pi}[n] \in G \text{ and } |\mathit{pref}(\bar{\pi}, n)|_{\mathrm{ds}} \in J\}.$$

The timed reachability probabilities for $\mathcal{M}$ are estimated by ds-bounded reachability probabilities for $\mathcal{M}_\delta$. The induced ds-bounded reachability probability for $\mathcal{M}$ (under σ) coincides with ds-bounded reachability probability on $\mathcal{M}_\delta$ (under $\mathrm{di}(\sigma)$).

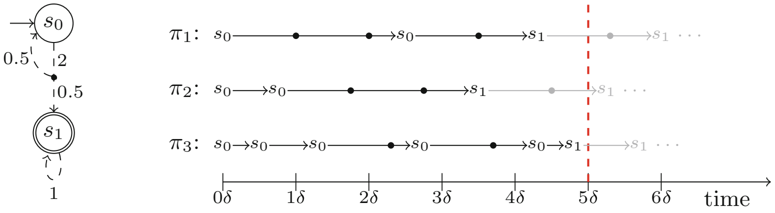
Proposition 4. *Let* $\mathcal{M}$ *be an MA with* $G \subseteq S$*,* $\sigma \in \mathrm{GM}$*, and digitization* $\mathcal{M}_\delta$*. Further, let* $J \subseteq \mathbb{N}$ *be a set of consecutive natural numbers. It holds that*

$$\mathrm{Pr}_{\sigma}^{\mathcal{M}}([\Diamond_{\mathrm{ds}}^{J} G]) = \mathrm{Pr}_{\mathrm{di}(\sigma)}^{\mathcal{M}_\delta}(\Diamond_{\mathrm{ds}}^{J} G).$$

Thus, induced ds-bounded reachability on MAs can be computed on their digitization. Next, we relate ds-bounded and timed reachability on MAs, i.e., we quantify the maximum difference between time-bounded and ds-bounded reachability probabilities.

Example 9. Let $\mathcal{M}$ be the MA given in Fig. 4(a). We consider the well-formed time interval $I = [0, 5\delta]$, yielding digitization step bounds $\mathrm{di}(I) = \{0, \dots, 5\}$. The digitization constant $\delta \in \mathbb{R}_{>0}$ remains unspecified in this example. Figure 4(b) illustrates paths π_1, π_2, and π_3 of $\mathcal{M}$. We depict sojourn times by arrow length. A black dot indicates that the path stays at the current state for a multiple of δ time units. All depicted paths reach $G = \{s_3\}$ within 5δ time units. However, the digitizations of π_1, π_2, and π_3 reach G within 5, 4, and 6 digitization steps, respectively. This yields

$$\pi_1, \pi_2 \in \Diamond^I G \cap [\Diamond_{\mathrm{ds}}^{\mathrm{di}(I)} G] \quad \text{and} \quad \pi_3 \in \Diamond^I G \setminus [\Diamond_{\mathrm{ds}}^{\mathrm{di}(I)} G].$$



s_0
0.5
2
0.5
s_1
1
π_1: s_0 s_0 s_1 s_1 · · ·
π_2: s_0 s_0 s_1 s_1 · · ·
π_3: s_0 s_0 s_0 s_0 s_0 s_1 s_1 · · ·
0δ
1δ
2δ
3δ
4δ
5δ
6δ
time

(a) MA $\mathcal{M}$.

(b) Sample paths of $\mathcal{M}$.

Fig. 4. MA $\mathcal{M}$ and illustration of paths of $\mathcal{M}$ (cf. Example 9).

Let $\lambda = \max\{\mathrm{E}(s) \mid s \in \mathrm{MS}\}$ be the maximum exit rate of $\mathcal{M}$. For $a \neq 0$ define

$$\varepsilon^{\downarrow}([a,b]) = \varepsilon^{\downarrow}([a,\infty)) = 1 - (1+\lambda\delta)^{\mathrm{di}_a} \cdot e^{-\lambda a}, \quad \varepsilon^{\downarrow}([0,b)) = \varepsilon^{\downarrow}([0,\infty]) = 0,$$
$$\varepsilon^{\uparrow}([a,b]) = \underbrace{1 - (1+\lambda\delta)^{\mathrm{di}_b} \cdot e^{-\lambda b}}_{=\varepsilon^{\uparrow}([0,b])} + \underbrace{1 - e^{-\lambda\delta}}_{=\varepsilon^{\uparrow}([a,\infty))}, \text{ and } \varepsilon^{\uparrow}([0,\infty)) = 0.$$

$\varepsilon^{\downarrow}(I)$ and $\varepsilon^{\uparrow}(I)$ approach 0 for small digitization constants $\delta \in \mathbb{R}_{>0}$.

Proposition 5. *For MA $\mathcal{M}$, scheduler $\sigma \in \mathrm{GM}$, goal states $G \subseteq S$, digitization constant $\delta \in \mathbb{R}_{>0}$ and time interval I*

$$\mathrm{Pr}_{\sigma}^{\mathcal{M}}(\Diamond^I G) \in \mathrm{Pr}_{\sigma}^{\mathcal{M}}([\Diamond_{\mathrm{ds}}^{I} G]) + \left[-\varepsilon^{\downarrow}(I),\, \varepsilon^{\uparrow}(I)\right]$$

Proof (Sketch). The sets $\Diamond^I G$ and $[\Diamond_{\mathrm{ds}}^{\mathrm{di}(I)} G]$ are illustrated in Fig. 5. We have

$$\mathrm{Pr}_{\sigma}(\Diamond^I G) = \mathrm{Pr}_{\sigma}([\Diamond_{\mathrm{ds}}^{\mathrm{di}(I)} G]) + \mathrm{Pr}_{\sigma}(\Diamond^I G \setminus [\Diamond_{\mathrm{ds}}^{\mathrm{di}(I)} G]) - \mathrm{Pr}_{\sigma}([\Diamond_{\mathrm{ds}}^{\mathrm{di}(I)} G] \setminus \Diamond^I G).$$

One then shows

$$\mathrm{Pr}_{\sigma}^{\mathcal{M}}(\Diamond^I G \setminus [\Diamond_{\mathrm{ds}}^{\mathrm{di}(I)} G]) \leq \varepsilon^{\uparrow}(I) \quad \text{and} \quad \mathrm{Pr}_{\sigma}^{\mathcal{M}}([\Diamond_{\mathrm{ds}}^{\mathrm{di}(I)} G] \setminus \Diamond^I G) \leq \varepsilon^{\downarrow}(I).$$

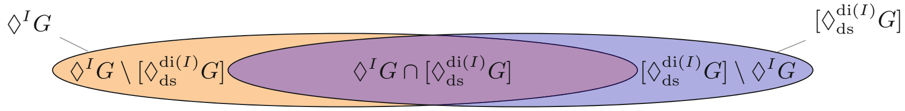

$\Diamond^I G$
$[\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G]$
$\Diamond^I G \setminus [\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G]$
$\Diamond^I G \cap [\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G]$
$[\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G] \setminus \Diamond^I G$

Fig. 5. Illustration of the sets $\Diamond^I G$ and $[\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G]$.

To this end, show for any $k \in \mathbb{N}$ that $1-(1+\lambda\delta)^k \cdot e^{-\lambda\delta k}$ is an upper bound for the probability of paths that induce more then k digitization steps within the first $k\delta$ time units. Then, this probability can be related to the probability of paths in $\Diamond^I G \setminus [\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G]$ and $[\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G] \setminus \Diamond^I G$, respectively.

From Propositions 4 and 5, we immediately have Corollary 1, which ensures that the value $\mathrm{Pr}^{\mathcal{M}}_{\sigma}(\Diamond^I G)$ can be approximated with arbitrary precision by computing $\mathrm{Pr}^{\mathcal{M}_\delta}_{\mathrm{di}(\sigma)}(\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G)$ for a sufficiently small δ.

Corollary 1. *For MA $\mathcal{M}$, scheduler $\sigma \in \mathrm{GM}$, goal states $G \subseteq S$, digitization constant $\delta \in \mathbb{R}_{>0}$ and time interval I*

$$\mathrm{Pr}^{\mathcal{M}}_{\sigma}(\Diamond^I G) \in \mathrm{Pr}^{\mathcal{M}_\delta}_{\mathrm{di}(\sigma)}(\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G) + \left[-\varepsilon^{\downarrow}(I),\, \varepsilon^{\uparrow}(I)\right]$$

This generalizes existing results [8,9] that only consider schedulers which maximize (or minimize) the corresponding probabilities. More details are given in [14].

Next, we lift Corollary 1 to multiple objectives $\mathbb{O} = (\mathbb{O}_1, \dots, \mathbb{O}_d)$. We define the satisfaction of a *timed* reachability objective $\mathbb{P}(\Diamond^I G)$ for the digitization $\mathcal{M}_\delta$ as $\mathcal{M}_\delta, \sigma \models \mathbb{P}(\Diamond^I G) \rhd_i p_i$ iff $\mathrm{Pr}^{\mathcal{M}_\delta}_{\sigma}(\Diamond^{\mathrm{di}(I)}_{\mathrm{ds}} G) \rhd_i p_i$. This allows us to consider notations like $\mathit{achieve}^{\mathcal{M}_\delta}(\mathbb{O} \rhd \mathbf{p})$, where $\mathbb{O}$ contains one or more timed reachability objectives. For a point $\mathbf{p} = (p_1, \dots, p_d) \in \mathbb{R}^d$ we consider the hyperrectangle

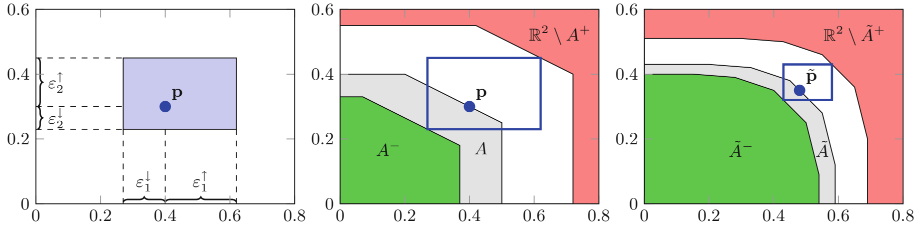
$$\varepsilon(\mathbb{O}, \mathbf{p}) = \bigtimes_{i=1}^{d} \left[p_i - \varepsilon^{\downarrow}_i,\, p_i + \varepsilon^{\uparrow}_i\right] \subseteq \mathbb{R}^d, \text{ where } \varepsilon^{\uparrow}_i = \begin{cases} \varepsilon^{\uparrow}(I) & \text{if } \mathbb{O}_i = \mathbb{P}(\Diamond^I G) \\ 0 & \text{if } \mathbb{O}_i = \mathbb{E}(\#j, G) \end{cases}$$

and $\varepsilon^{\downarrow}_i$ is defined similarly. The next example shows how the set of achievable points of $\mathcal{M}$ can be approximated using achievable points of $\mathcal{M}_\delta$.

Example 10. Let $\mathbb{O} = (\mathbb{P}(\Diamond^{I_1} G_1), \mathbb{P}(\Diamond^{I_2} G_2))$ be two timed reachability objectives for an MA $\mathcal{M}$ with digitization $\mathcal{M}_\delta$ such that $\varepsilon^{\downarrow}_1 = 0.13$, $\varepsilon^{\uparrow}_1 = 0.22$, $\varepsilon^{\downarrow}_2 = 0.07$, and $\varepsilon^{\uparrow}_2 = 0.15$. The blue rectangle in Fig. 6(a) illustrates the set $\varepsilon(\mathbb{O}, \mathbf{p})$ for the point $\mathbf{p} = (0.4, 0.3)$. Assume $\mathit{achieve}^{\mathcal{M}_\delta}(\mathbb{O} \rhd \mathbf{p})$ holds for threshold relations $\rhd = \{\geq, \geq\}$, i.e., $\mathbf{p}$ is achievable for the digitization $\mathcal{M}_\delta$. From Corollary 1, we infer that $\varepsilon(\mathbb{O}, \mathbf{p})$ contains at least one point $\mathbf{p}'$ that is achievable for $\mathcal{M}$. Hence, the bottom left corner point of the rectangle is achievable for $\mathcal{M}$. This holds for any rectangle $\varepsilon(\mathbb{O}, \mathbf{q})$ with $\mathbf{q} \in A$, where A is the set of achievable points of $\mathcal{M}_\delta$ denoted by the gray area[1] in Fig. 6(b). It follows that any point in A^- (depicted

[1] In the figure, A^- partly overlaps A, i.e., the green area also belongs to A.

by the green area) is achievable for $\mathcal{M}$. On the other hand, an achievable point of $\mathcal{M}$ has to be contained in a set $\varepsilon(\mathbb{O}, \mathbf{q})$ for at least one $\mathbf{q} \in A$. The red area depicts the points $\mathbb{R}^d \setminus A^+$ for which this is not the case, i.e., points that are not achievable for $\mathcal{M}$. The digitization constant δ controls the accuracy of the resulting approximation. Figure 6(c) depicts a possible result when a smaller digitization constant $\tilde{\delta} < \delta$ is considered.



0.6
0.4
0.2
0
0
0.2
0.4
0.6
0.8
$\varepsilon_2^{\uparrow}$
$\varepsilon_2^{\downarrow}$
$\mathbf{p}$
$\varepsilon_1^{\downarrow}$
$\varepsilon_1^{\uparrow}$
$\mathbb{R}^2 \setminus A^+$
$\mathbf{p}$
A^-
A
$\mathbb{R}^2 \setminus \tilde{A}^+$
$\tilde{\mathbf{p}}$
$\tilde{A}^-$
$\tilde{A}$

(a) The set $\varepsilon(\mathbb{O}, \mathbf{p})$. (b) Coarse approximation. (c) Refined approximation.

Fig. 6. Approximation of achievable points. (Color figure online)

The observations from the example above are formalized in the following theorem. The theorem also covers unbounded reachability objectives by considering the time interval $I = [0, \infty)$. For expected reward objectives of the form $\mathbb{E}(\#j, G)$ it can be shown that $\mathrm{ER}_{\sigma}^{\mathcal{M}}(\rho_j, G) = \mathrm{ER}_{\mathrm{di}(\sigma)}^{\mathcal{M}_\delta}(\rho_j^\delta, G)$. This claim is similar to Proposition 3 and can be shown analogously. This enables multi-objective model checking of MAs with timed reachability objectives.

Theorem 5. *Let $\mathcal{M}$ be an MA with digitization $\mathcal{M}_\delta$. Furthermore, let $\mathbb{O}$ be (un)timed reachability or expected reward objectives with threshold relations $\rhd$ and $|\mathbb{O}| = d$. It holds that $A^- \subseteq \{\mathbf{p} \in \mathbb{R}^d \mid achieve^{\mathcal{M}}(\mathbb{O} \rhd \mathbf{p})\} \subseteq A^+$ with:*

$$A^- = \{\mathbf{p}' \in \mathbb{R}^d \mid \forall \mathbf{p} \in \mathbb{R}^d \colon \mathbf{p}' \in \varepsilon(\mathbb{O}, \mathbf{p}) \textit{ implies } achieve^{\mathcal{M}_\delta}(\mathbb{O} \rhd \mathbf{p})\} \textit{ and}$$
$$A^+ = \{\mathbf{p}' \in \mathbb{R}^d \mid \exists \mathbf{p} \in \mathbb{R}^d \colon \mathbf{p}' \in \varepsilon(\mathbb{O}, \mathbf{p}) \textit{ and } achieve^{\mathcal{M}_\delta}(\mathbb{O} \rhd \mathbf{p})\}.$$

5 Experimental Evaluation

Implementation. We implemented multi-objective model checking of MAs into Storm [32]. The input model is given in the PRISM language[2] and translated into a sparse representation. For MA $\mathcal{M}$, the implementation performs a multi-objective analysis on the underlying MDP $\mathcal{M}_\mathcal{D}$ or a digitization $\mathcal{M}_\delta$ and infers (an approximation of) the achievable points of $\mathcal{M}$ by exploiting the results from Sect. 4. For computing the achievable points of $\mathcal{M}_\mathcal{D}$ and $\mathcal{M}_\delta$, we apply the approach of [16]. It repeatedly checks weighted combinations of the objectives

[2] We slightly extend the PRISM language in order to describe MAs.

(by means of *value iteration* [30] – a standard technique in single-objective MDP model checking) to refine an approximation of the set of achievable points. This procedure is extended as follows. Full details can be found in [33].

- We support ds-bounded reachability objectives by combining the approach of [16] (which supports step-bounded reachability on MDPs) with techniques from single-objective MA analysis [8]. Roughly, we reduce ds-bounded reachability to untimed reachability by storing the digitized time-epoch (i.e., the current number of digitization steps) into the state space. A blow-up of the resulting model is avoided by considering each time-epoch separately.
- In contrast to [16], we allow a simultaneous analysis of minimizing and maximizing expected reward objectives. This is achieved by performing additional preprocessing steps that comprise an analysis of end components.

The source code including all material to reproduce the experiments is available at http://www.stormchecker.org/benchmarks.html.

Setup. Our implementation uses a single core (2 GHz) of a 48-core HP BL685C G7 limited to 20 GB RAM. The timeout (TO) is two hours. For a model, a set of objectives, and a precision $\eta \in \mathbb{R}_{>0}$, we measure the time to compute an η-approximation[3] of the set of achievable points. This set-up coincides with Pareto queries as discussed in [16]. The digitization constant δ is chosen heuristically such that recalculations with smaller constants $\tilde{\delta} < \delta$ are avoided. We set the precision for value-iteration to $\varepsilon = 10^{-6}$. We use classical value iteration; the use of improved algorithms [34] is left for future work.

Results for MAs. We consider four case studies: (i) a *job scheduler* [13], see Sect. 1; (ii) a *polling system* [35,36] containing a server processing jobs that arrive at two stations; (iii) a *video streaming client* buffering received packages and deciding when to start playback; and (iv) a randomized *mutual exclusion algorithm* [36], a variant of [37] with a process-dependent random delay in the critical section. Details on the benchmarks and the objectives are given in [14].

Table 1 lists results. For each instance we give the defining constants, the number of states of the MA and the used η-approximation. A multi-objective query is given by the triple (l, m, n) indicating l untimed, m expected reward, and n timed objectives. For each MA and query we depict the total run-time of our implementation (time) and the number of vertices of the obtained under-approximation (*pts*).

Queries analyzed on the underlying MDP are solved efficiently on large models with up to millions of states. For timed objectives the run-times increase drastically due to the costly analysis of digitized reachability objectives on the digitization, cf. [9]. Queries with up to four objectives can be dealt with within the time limit. Furthermore, for an approximation one order of magnitude better, the number of vertices of the result increases approximately by a factor three.

[3] An η-approximation of $A \subseteq \mathbb{R}^d$ is given by $A^-, A^+ \subseteq \mathbb{R}^d$ with $A^- \subseteq A \subseteq A^+$ and for all $\mathbf{p} \in A^+$ exists a $\mathbf{q} \in A^-$ such that the distance between $\mathbf{p}$ and $\mathbf{q}$ is at most η.

Table 1. Experimental results for multi-objective MAs.

benchmark			$(\Diamond, \text{ER}, \Diamond^I)$		$(\Diamond, \text{ER}, \Diamond^I)$		$(\Diamond, \text{ER}, \Diamond^I)$		$(\Diamond, \text{ER}, \Diamond^I)$	
N(-K)	#states	$\log_{10}(\eta)$	pts	time	pts	time	pts	time	pts	time
job scheduling			(0, 3, 0)		(0, 1, 1)		(1, 3, 0)		(1, 1, 2)	
10-2	12 554	−2	9	1.8	9	41	15	435	16	2 322
		−3	44	128	21	834	TO		TO	
12-3	116 814	−2	11	42	9	798	21	2 026	TO	
		−3	53	323	TO		TO		TO	
17-2	$4.6 \cdot 10^6$	−2	14	1 040	TO		22	4 936	TO	
		−3	58	2 692	TO		TO		TO	
polling			(0, 2, 0)		(0, 4, 0)		(0, 0, 2)		(0, 2, 2)	
3-2	1 020	−2	4	0.3	5	0.6	3	130	12	669
		−3	4	0.3	5	0.8	7	3 030	TO	
3-3	9 858	−2	5	1.3	8	23	6	2 530	TO	
		−3	6	2.0	19	3 199	TO		TO	
4-4	827 735	−2	10	963	20	4 349	TO		TO	
		−3	11	1 509	TO		TO		TO	
stream			(0, 2, 0)		(0, 1, 1)		(0, 0, 2)		(0, 2, 1)	
30	1 426	−2	20	0.9	16	90	16	55	26	268
		−3	51	8.8	46	2 686	38	1 341	TO	
250	94 376	−2	31	50	15	5 830	16	4 050	TO	
		−3	90	184	TO		TO		TO	
1000	$1.5 \cdot 10^6$	−2	41	3 765	TO		TO		TO	
		−3	TO		TO		TO		TO	
mutex			(0, 0, 3)		(0, 0, 3)					
2	13 476	−2	16	351	13	1 166				
		−3	13	2 739	TO					
3	38 453	−2	15	2 333	TO					

In addition, a lower digitization constant has then to be considered which often leads to timeouts in experiments with timed objectives.

Comparison with `PRISM` [15] *and* `IMCA` [9]. We compared the performance of our implementation with both `PRISM` and `IMCA`. Verification times are summarized in Fig. 7: On points above the diagonal, our implementation is faster. For the comparison with `PRISM` (no MAs), we considered the multi-objective MDP benchmarks from [16, 19]. Both implementations are based on [16]. For the comparison with `IMCA` (no multi-objective queries) we used the benchmarks from Table 1, with just a single objective. We observe that our implementation is competitive. Details are given in [14].

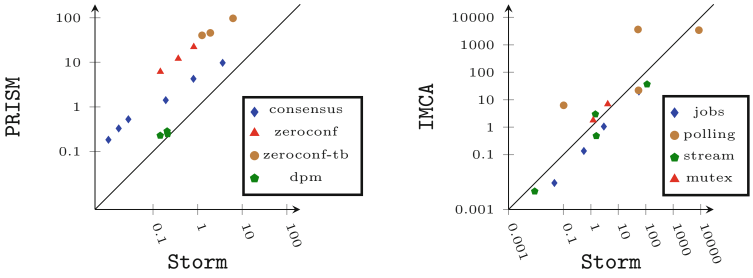


Fig. 7. Verification times (in seconds) of our implementation and other tools.

6 Conclusion

We considered multi-objective verification of Markov automata, including in particular timed reachability objectives. The next step is to apply our algorithms to the manifold applications of MA, such as generalized stochastic Petri nets to enrich the analysis possibilities of such nets.

Acknowledgement. This work was supported by the CDZ project CAP (GZ 1023).

References

1. Eisentraut, C., Hermanns, H., Zhang, L.: On probabilistic automata in continuous time. In: Proceedings of LICS, IEEE CS, pp. 342–351 (2010)
2. Deng, Y., Hennessy, M.: On the semantics of Markov automata. Inf. Comput. **222**, 139–168 (2013)
3. Boudali, H., Crouzen, P., Stoelinga, M.: A rigorous, compositional, and extensible framework for dynamic fault tree analysis. IEEE Trans. Dependable Secur. Comput. **7**(2), 128–143 (2010)
4. Coste, N., Hermanns, H., Lantreibecq, E., Serwe, W.: Towards performance prediction of compositional models in industrial GALS designs. In: Bouajjani, A., Maler, O. (eds.) CAV 2009. LNCS, vol. 5643, pp. 204–218. Springer, Heidelberg (2009). doi:10.1007/978-3-642-02658-4_18
5. Katoen, J.P., Wu, H.: Probabilistic model checking for uncertain scenario-aware data flow. ACM Trans. Embed. Comput. Sys. **22**(1), 15:1–15:27 (2016)
6. Bozzano, M., Cimatti, A., Katoen, J.P., Nguyen, V.Y., Noll, T., Roveri, M.: Safety, dependability and performance analysis of extended AADL models. Comput. J. **54**(5), 754–775 (2011)
7. Eisentraut, C., Hermanns, H., Katoen, J.-P., Zhang, L.: A semantics for every GSPN. In: Colom, J.-M., Desel, J. (eds.) PETRI NETS 2013. LNCS, vol. 7927, pp. 90–109. Springer, Heidelberg (2013). doi:10.1007/978-3-642-38697-8_6
8. Hatefi, H., Hermanns, H.: Model checking algorithms for Markov automata. ECEASST **53** (2012)
9. Guck, D., Hatefi, H., Hermanns, H., Katoen, J.P., Timmer, M.: Analysis of timed and long-run objectives for Markov automata. LMCS **10**(3) (2014)
10. Guck, D., Timmer, M., Hatefi, H., Ruijters, E., Stoelinga, M.: Modelling and analysis of Markov reward automata. In: Cassez, F., Raskin, J.-F. (eds.) ATVA 2014. LNCS, vol. 8837, pp. 168–184. Springer, Cham (2014). doi:10.1007/978-3-319-11936-6_13
11. Hatefi, H., Braitling, B., Wimmer, R., Fioriti, L.M.F., Hermanns, H., Becker, B.: Cost vs. time in stochastic games and Markov automata. In: Li, X., Liu, Z., Yi, W. (eds.) SETTA 2015. LNCS, vol. 9409, pp. 19–34. Springer, Cham (2015). doi:10.1007/978-3-319-25942-0_2
12. Butkova, Y., Wimmer, R., Hermanns, H.: Long-run rewards for Markov automata. In: Legay, A., Margaria, T. (eds.) TACAS 2017. LNCS, vol. 10206, pp. 188–203. Springer, Heidelberg (2017). doi:10.1007/978-3-662-54580-5_11
13. Bruno, J.L., Downey, P.J., Frederickson, G.N.: Sequencing tasks with exponential service times to minimize the expected flow time or makespan. J. ACM **28**(1), 100–113 (1981)

14. Quatmann, T., Junges, S., Katoen, J.P.: Markov automata with multiple objectives (2017). CoRR abs/1704.06648
15. Kwiatkowska, M., Norman, G., Parker, D.: PRISM 4.0: verification of probabilistic real-time systems. In: Gopalakrishnan, G., Qadeer, S. (eds.) CAV 2011. LNCS, vol. 6806, pp. 585–591. Springer, Heidelberg (2011). doi:10.1007/978-3-642-22110-1_47
16. Forejt, V., Kwiatkowska, M., Parker, D.: Pareto curves for probabilistic model checking. In: Chakraborty, S., Mukund, M. (eds.) ATVA 2012. LNCS, vol. 7561, pp. 317–332. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33386-6_25
17. Roijers, D.M., Vamplew, P., Whiteson, S., Dazeley, R.: A survey of multi-objective sequential decision-making. J. Artif. Intell. Res. **48**, 67–113 (2013)
18. Etessami, K., Kwiatkowska, M., Vardi, M.Y., Yannakakis, M.: Multi-objective model checking of Markov decision processes. In: Grumberg, O., Huth, M. (eds.) TACAS 2007. LNCS, vol. 4424, pp. 50–65. Springer, Heidelberg (2007). doi:10.1007/978-3-540-71209-1_6
19. Forejt, V., Kwiatkowska, M., Norman, G., Parker, D., Qu, H.: Quantitative multi-objective verification for probabilistic systems. In: Abdulla, P.A., Leino, K.R.M. (eds.) TACAS 2011. LNCS, vol. 6605, pp. 112–127. Springer, Heidelberg (2011). doi:10.1007/978-3-642-19835-9_11
20. Bruyère, V., Filiot, E., Randour, M., Raskin, J.F.: Meet your expectations with guarantees: beyond worst-case synthesis in quantitative games. In: Proceeding of STACS. LIPIcs, vol. 25, pp. 199–213. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik (2014)
21. Baier, C., Dubslaff, C., Klüppelholz, S.: Trade-off analysis meets probabilistic model checking. In: CSL-LICS, pp. 1:1–1:10. ACM (2014)
22. Brázdil, T., Chatterjee, K., Forejt, V., Kucera, A.: Trading performance for stability in Markov decision processes. J. Comput. Syst. Sci. **84**, 144–170 (2017)
23. Brázdil, T., Brozek, V., Chatterjee, K., Forejt, V., Kucera, A.: Markov decision processes with multiple long-run average objectives. LMCS **10**(1) (2014)
24. Basset, N., Kwiatkowska, M., Topcu, U., Wiltsche, C.: Strategy synthesis for stochastic games with multiple long-run objectives. In: Baier, C., Tinelli, C. (eds.) TACAS 2015. LNCS, vol. 9035, pp. 256–271. Springer, Heidelberg (2015). doi:10.1007/978-3-662-46681-0_22
25. Teichteil-Königsbuch, F.: Path-constrained Markov decision processes: bridging the gap between probabilistic model-checking and decision-theoretic planning. In: Proceedings of ECAI. Frontiers in AI and Applications, vol. 242, pp. 744–749. IOS Press (2012)
26. Randour, M., Raskin, J.-F., Sankur, O.: Variations on the stochastic shortest path problem. In: D'Souza, D., Lal, A., Larsen, K.G. (eds.) VMCAI 2015. LNCS, vol. 8931, pp. 1–18. Springer, Heidelberg (2015). doi:10.1007/978-3-662-46081-8_1
27. Junges, S., Jansen, N., Dehnert, C., Topcu, U., Katoen, J.-P.: Safety-constrained reinforcement learning for MDPs. In: Chechik, M., Raskin, J.-F. (eds.) TACAS 2016. LNCS, vol. 9636, pp. 130–146. Springer, Heidelberg (2016). doi:10.1007/978-3-662-49674-9_8
28. David, A., Jensen, P.G., Larsen, K.G., Legay, A., Lime, D., Sørensen, M.G., Taankvist, J.H.: On time with minimal expected cost!. In: Cassez, F., Raskin, J.-F. (eds.) ATVA 2014. LNCS, vol. 8837, pp. 129–145. Springer, Cham (2014). doi:10.1007/978-3-319-11936-6_10
29. Chen, T., Forejt, V., Kwiatkowska, M., Simaitis, A., Wiltsche, C.: On stochastic games with multiple objectives. In: Chatterjee, K., Sgall, J. (eds.) MFCS 2013. LNCS, vol. 8087, pp. 266–277. Springer, Heidelberg (2013). doi:10.1007/978-3-642-40313-2_25

30. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley, New York (1994)
31. Neuhäußer, M.R., Stoelinga, M., Katoen, J.-P.: Delayed nondeterminism in continuous-time Markov decision processes. In: de Alfaro, L. (ed.) FoSSaCS 2009. LNCS, vol. 5504, pp. 364–379. Springer, Heidelberg (2009). doi:10.1007/978-3-642-00596-1_26
32. Dehnert, C., Junges, S., Katoen, J.P., Volk, M.: A storm is coming: a modern probabilistic model checker. In: Majumdar, R., Kučnak, V. (eds.) CAV 2017, Part I. LNCS, vol. 10426, pp. 592–600. Springer, Cham (2017)
33. Quatmann, T.: Multi-objective model checking of Markov automata. Master's thesis, RWTH Aachen University (2016)
34. Haddad, S., Monmege, B.: Reachability in MDPs: refining convergence of value iteration. In: Ouaknine, J., Potapov, I., Worrell, J. (eds.) RP 2014. LNCS, vol. 8762, pp. 125–137. Springer, Cham (2014). doi:10.1007/978-3-319-11439-2_10
35. Srinivasan, M.M.: Nondeterministic polling systems. Manag. Sci. **37**(6), 667–681 (1991)
36. Timmer, M., Katoen, J.-P., Pol, J., Stoelinga, M.I.A.: Efficient modelling and generation of Markov automata. In: Koutny, M., Ulidowski, I. (eds.) CONCUR 2012. LNCS, vol. 7454, pp. 364–379. Springer, Heidelberg (2012). doi:10.1007/978-3-642-32940-1_26
37. Pnueli, A., Zuck, L.: Verification of multiprocess probabilistic protocols. Distrib. Comput. **1**(1), 53–72 (1986)