Planning for Stochastic Games with Co-safe Objectives

Lei Song and **Yuan Feng**
Centre for Quantum Computation and Intelligent Systems,
University of Technology Sydney, Australia

Lijun Zhang
State Key Laboratory of Computer Science,
Institute of Software,
Chinese Academy of Sciences, China

Abstract

We consider planning problems for stochastic games with objectives specified by a branching-time logic, called probabilistic computation tree logic (PCTL). This problem has been shown to be undecidable if strategies with perfect recall, i.e., history-dependent, are considered. In this paper, we show that, if restricted to co-safe properties, a subset of PCTL properties capable to specify a wide range of properties in practice including reachability ones, the problem turns to be decidable, even when the class of general strategies is considered. We also give an algorithm for solving robust stochastic planning, where a winning strategy is tolerant to some perturbations of probabilities in the model. Our result indicates that satisfiability of co-safe PCTL is decidable as well.

1 Introduction

Markov Decision Processes (MDPs) are powerful models for systems involving both decision-making and probabilistic dynamics [Puterman, 1994]. In an MDP, states can be partitioned into *controller* states and *probabilistic* states. At a controller state, a decision has to be made to choose a successor, while at a probabilistic state, the successor will be sampled with prescribed probability distribution. One popular research area of MDPs in Artificial Intelligence (AI) is *probabilistic planning* [Mausam and Kolobov, 2012]: Given an objective, can controller states choose successors in such a way that the objective is fulfilled. Most of previous work restricted to reachability properties, for which memoryless deterministic strategies suffice, namely, given a set of goal states and a threshold, can we find a map from controller states to their successors such that the probability of reaching the goal states is greater than the threshold? See for instance [Xu and Mannor, 2011; Mausam and Kolobov, 2012; Pineda *et al.*, 2013]. Robust planning for reachability properties has also been studied [Nilim and Ghaoui, 2005; Delage and Mannor, 2007] to deal with MDPs where transitions probabilities are not known precisely.

In practice, however, we often encounter sophisticated objectives beyond reachability properties, hence making previous work not applicable. To see this, let us consider a gambling game between a gambler G with an initial credit $c > 0$ and a machine M with two buttons: an optimistic one and a pessimistic one. By pressing the optimistic button, with probability $\frac{1}{2}$, M will award G 10 credits, while with probability $\frac{1}{2}$, M will take 8 credits from G. If the pessimistic button is chosen, M will award G 2 credits or take 1 credit from G with probability $\frac{2}{3}$ and $\frac{1}{3}$, respectively. The gambler can repeat the game as many times as he wishes unless he loses all credits. In case the number of credits is less than 8, then only the pessimistic button is available. Suppose $c = 10$ initially. We would like to know (*)*"whether G has a strategy such that: 1) with probability greater than 0.2 that G will have no less than 20 credits, and 2) in the next n rounds, with probability equal to or less than 0.3 that at each game the probability of losing all credits eventually is greater than 0.5."* Such a requirement can be specified by the Probabilistic Computation Tree Logic [Hansson and Jonsson, 1994] (PCTL) as follows:

$$[\Diamond(c \geqslant 20)]_{>0.2} \wedge [\Box^{\leqslant n}[\Diamond(c = 0)]_{>0.5}]_{\leqslant 0.3}. \tag{1}$$

Intuitively, $\Diamond someting$ and $\Box something$ mean "$something$ will happen eventually" and "$something$ will always hold", respectively. The probability operator $[some\ event]_{>0.2}$ denotes that the probability of $some\ event$ happening is greater than 0.2. The formal syntax and semantics of PCTL can be found in Section 2.2. Clearly, the requirement in Eq. (1) cannot be represented as a reachability property.

In this paper we study stochastic games with branching time winning objectives given by PCTL formulas, which have been studied extensively in the model checking community [Baier *et al.*, 2004; Kucera and Strazovský, 2008; Brázdil *et al.*, 2006; Baier *et al.*, 2012]. Different from reachability properties, winning strategies of PCTL objectives may require memory and randomization [Baier *et al.*, 2004], hence makes the problem much harder to be solved. In [Baier *et al.*, 2004], it was shown that the problem of deciding whether there exists a memoryless deterministic winning strategy for a PCTL objective is NP-complete. An algorithm in EXPTIME was given in [Kucera and Strazovský, 2008] to solve the same problem except that randomized strategies were taken into account. When strategies with prefect recall are considered, the problem turns to be highly undecidable [Brázdil *et al.*, 2006], no matter whether strategies are deterministic or randomized. These results are summarized in Table 1 and 2.

The main contribution of the current paper is shown in the

Table 1: Complexity results for different classes of strategies when PCTL properties are considered

	Memoryless	Perfect recall
Deterministic	NP-complete	Undecidable
Randomized	EXPTIME	Undecidable

Table 2: Complexity results for different fragments of properties when randomized strategies with perfect recall are considered. The middle column is our result in this paper.

Reachability	Co-safe PCTL	PCTL
NP-complete	Decidable	Undecidable

middle column of Table 2. To be specific, we focus on co-safe PCTL [Katoen *et al.*, 2014], a subset of PCTL which includes a wide range of practical properties, in particular reachability ones and the one in Eq. (1). For this set of properties, memoryless deterministic strategies do not suffice, unlike the case for reachability. We show that stochastic games with co-safe PCTL objectives are decidable, even when the most general strategies, i.e. randomized strategies with perfect recall, are considered. Other contributions of this paper include:

- We show that *robust* stochastic games [Kucera and Strazovský, 2008] with co-safe PCTL objectives are also decidable, where a winning strategy is tolerant to a certain amount of perturbation.
- We prove that satisfiability of co-safe PCTL properties is decidable as well. This partially solves the PCTL satisfiability problem, which has been open for quite a long time [Brázdil *et al.*, 2008; Bertrand *et al.*, 2012].

Related Work Stochastic planning with co-safe objectives has been studied in [Lacerda *et al.*, 2014]. However, the properties under consideration are in linear-time and incomparable with co-safe properties in branching-time [Baier and Katoen, 2008]. In [Teichteil-Königsbuch, 2012], planning problems with path constraints specified by PCTL were proposed. Different from our work, [Teichteil-Königsbuch, 2012] restricted to memoryless strategies.

Related work also includes research on pATL and pATL* and their model checking issues [Chen and Lu, 2007; Huang *et al.*, 2012; Huang and Luo, 2013]. However, we mention that for stochastic games pATL* and pATL still cannot express properties like the one in Eq. (1). Recall that a pATL formula in form of $\langle\langle A\rangle\rangle^{>q}(\phi)$ means that players in A can enforce a strategy such that ϕ is satisfied with probability greater than q no matter what other players do. In case there is only one player, it simply means that the player has a strategy such that ϕ will be satisfied with probability greater than q. At the first glance, it seems that probabilistic planning problems can be encoded by pATL. However, the semantics of pATL does not bind a unique strategy for players appearing in scopes of different qualifiers in form of $\langle\langle A\rangle\rangle^{>q}(\cdot)$. For instance, a natural choice for expressing the requirement in Eq. (1) is a pATL formula $\Phi_1 \wedge \Phi_2$, where $\Phi_1 = \langle\langle A_1\rangle\rangle^{>0.2}\Diamond(c \geqslant 20)$ and

$$\Phi_2 = \langle\langle A_2\rangle\rangle^{\leqslant 0.3}\Box^{\leqslant n}(\langle\langle A_3\rangle\rangle^{>0.5}\Diamond(c=0))$$

with $A_1 = A_2 = A_3 = \{G\}$. Note G appears for three times in $\Phi_1 \wedge \Phi_2$. In order to satisfy this formula, G can use different strategies when at A_1, A_2, and A_3. For instance, suppose $c = 10$ initially, by choosing the optimistic button, G can satisfy $\Diamond(c \geqslant 20)$ with probability $\frac{1}{2}$, i.e., Φ_1 is satisfied. On the other hand, by a simple calculation, if the pessimistic button is always chosen, then Φ_2 will be satisfied for any $n > 0$. Therefore, the whole formula is satisfied by choosing different strategies for G in A_1 and A_2, A_3. However, this is different from our requirement (*) given before, where we would like to find a single strategy which satisfies the two requirements at the same time.

Organization of the Paper Section 2 introduces some preliminary notations. Co-safe PCTL is formally defined in Section 3. We show the decidability of stochastic games with co-safe PCTL objectives in Section 4, where strategies are deterministic. Section 5 generalizes results in Section 4 to randomized strategies, where robust planning problems are also addressed. We conclude the paper in Section 6.

2 Preliminaries

We introduce some preliminary notations. Let $\mathbb{N}$ be the set of nature numbers and $\mathbb{N}^+$ contain only positive ones. Let $\mathbb{R}$ be the set of real numbers, while $\mathbb{Q}$ contains only rationals. For a countable set S, let $\mathcal{P}(S)$ denote its powerset. A distribution is a function $\mu : S \to [0,1]$ satisfying $\sum_{s\in S}\mu(s) = 1$. Let $Dist(S)$ denote the set of distributions over S. We shall use $s, r, t, \ldots$ and $\mu, \nu, \ldots$ to range over S and $Dist(S)$, respectively. Let S^* and S^ω denote the set of finite and infinite sequences, respectively, over the set S. The set of all (finite and infinite) sequences over S is given by $S^\infty = S^* \cup S^\omega$. Let $|\pi|$ denote the length of $\pi \in S^*$. For $i \in \mathbb{N}$, let $\pi[i]$ denote the $(i{+}1)$-th element of π provided $i < |\pi|$, and $\pi\!\downarrow = \pi[|\pi|-1]$ denote the last element of π provided $\pi \in S^*$. The concatenation of π_1 and π_2, denoted $\pi_1 \cdot \pi_2$, is the sequence obtained by appending π_2 to the end of π_1, provided π_1 is finite. A sequence π_1 is a prefix of π_2, denoted $\pi_1 \preceq \pi_2$, if there exists $\pi \in S^\infty$ (possibly empty, i.e., $\pi \in S^0$) such that $\pi_1 \cdot \pi = \pi_2$. If $\pi \notin S^0$, then π_1 is a proper prefix of π_2, denoted $\pi_1 \prec \pi_2$. The set $\Pi \subseteq S^\infty$ is *prefix-closed* iff for all $\pi_1 \in \Pi$ and $\pi_2 \in S^*$, $\pi_2 \preceq \pi_1$ implies $\pi_2 \in \Pi$.

2.1 Probabilistic Models

In this subsection we recall some probabilistic models widely used in AI community. We shall first introduce the definition of Markov chain as below:

Definition 1. *A* Markov chain *(*MC*)* $\mathsf{D} = (\mathcal{S}, \bar{s}, \mathsf{P}, AP, L)$ *is a tuple where $\mathcal{S}$ is a countable set of states; $\bar{s} \in \mathcal{S}$ is the initial state; $\mathsf{P} : \mathcal{S}\times\mathcal{S} \mapsto ([0,1]\cap\mathbb{Q})$ is a transition matrix such that $\sum_{t\in\mathcal{S}}\mathsf{P}(s,t) = 1$ for each $s \in \mathcal{S}$; AP is a finite set of atomic propositions; and $L : \mathcal{S} \to \mathcal{P}(AP)$ is a labelling function.*

A path $\pi \in \mathcal{S}^\infty$ through an MC D is a (finite or infinite) sequence of states. The cylinder set C_π of $\pi \in \mathcal{S}^*$

is defined as: $C_\pi = \{\pi' \in \mathcal{S}^\omega \mid \pi \prec \pi'\}$. The σ-algebra $\mathcal{F}$ of D is the smallest σ-algebra containing all cylinder sets C_π. By standard probability theory [Halmos, 1974; Rudin, 2006], there exists a unique probability measure $\Pr$ on $\mathcal{F}$ such that: $\Pr(C_\pi) = 1$ if $\pi = \bar{s}$, and $\Pr(C_\pi) = \Pi_{0 \leqslant i < n} \mathsf{P}(s_i, s_{i+1})$ if $\pi = s_0 \dots s_n$ with $\bar{s} = s_0$ and $n > 0$; otherwise $\Pr(C_\pi) = 0$. We are restricted to rational probabilities in this paper, which is standard and essential to guarantee our decidability results.

While being fully probabilistic, MCs cannot handle systems involving decision-making. For these cases, Markov Decision Processes (MDPs) shall be introduced.

Definition 2. *A* Markov Decision Process *(*MDP*) is a tuple* $\mathsf{M} = (\mathcal{S}, (\mathcal{S}_{\bigcirc}, \mathcal{S}_{\Box}), \bar{s}, \mathcal{E}, \mathsf{P}, AP, L)$*, where* $\bar{s}$*,* AP*,* L *are as in Definition 1 and* $\mathcal{S}$ *is a finite set of states;* $(\mathcal{S}_{\bigcirc}, \mathcal{S}_{\Box})$ *is a partition of* $\mathcal{S}$ *with* $\mathcal{S}_{\bigcirc}$ *containing* probabilistic *states and* $\mathcal{S}_{\Box}$ *all* controller *states;* $\mathcal{E} \subseteq \mathcal{S}_{\Box} \times \mathcal{S}$ *is a set of transitions;* $\mathsf{P} : \mathcal{S}_{\bigcirc} \times \mathcal{S} \mapsto ([0,1] \cap \mathbb{Q})$ *is a* probability function *such that* $\sum_{t \in \mathcal{S}} \mathsf{P}(s,t) = 1$ *for each* $s \in \mathcal{S}_{\bigcirc}$.

In Definition 2, we explicitly distinguish between probabilistic states and controller states which is typical in the setting of stochastic games, see for instance [Brázdil *et al.*, 2006]. A player at controller states has full flexibility to choose its successors, while at probabilistic states, successor states have to be chosen with probabilities given by the function P, hence a player at $\mathcal{S}_{\bigcirc}$ has limited flexibility. We mention that the distinguishing $\mathcal{S}_{\Box}$ and $\mathcal{S}_{\bigcirc}$ is only to ease the presentation. Actually, our definition of MDP is of no essential difference from the standard definition, as they can be transformed into each other directly. In the sequel we write $\mathsf{P}(s, \cdot)$ to denote the distribution such that $\mathsf{P}(s, \cdot)(t) = \mathsf{P}(s,t)$ for each $t \in \mathcal{S}$.

Due to existence of controller states, the notion of *strategies* has to be introduced in order to resolve all non-deterministic choices at controller states.

Definition 3. *Let* M *be an* MDP *as in Definition 2. A strategy* σ *of* M *is a function such that for any* $\pi \in \mathcal{S}^*$ *with* $\pi{\downarrow} \in \mathcal{S}_{\Box}$*,* $\sigma(\pi) \in Dist(\mathcal{S})$*, where* $\sigma(\pi)(s) > 0$ *implies* $(\pi{\downarrow}, s) \in \mathcal{E}$ *for each* $s \in \mathcal{S}$.

Intuitively, a strategy takes a history execution as an input, represented as a finite path π, and decides for state $\pi{\downarrow}$ the probabilities with which it should choose all its successors. Sometimes, the output of a strategy given a finite path will be referred as the decisions of the strategy.

Given an MDP M and a strategy σ of M, we can always induce an MC, denoted M_σ, for which a unique probability measure can be defined. As the inputs of a strategy, i.e., all history executions, are of infinitely many, the state space of the resultant MC may be infinite even if M is finite, which explains why in Definition 1 we are not restricted to MCs of finite states.

Depending on the information available when decisions are made, strategies can be classified into different categories according to criteria in Table 1: (i) Perfect recall v.s. Memoryless: Strategies whose decisions are based on the full history are with perfect recall. In contrast, a memoryless strategy σ makes decisions only based on the current state. Hence, all memoryless strategies can be written as a function $\sigma : \mathcal{S}_{\Box} \mapsto Dist(\mathcal{S})$, and each MC induced by a memoryless strategy has the same state space as the original MDP. (ii) Randomized v.s. Deterministic: Strategies defined in Definition 3 are randomized, as decisions of strategies are distributions over successor states. In contrast, a deterministic strategy σ always chooses a successor with probability 1. The combinations of the above criteria result in four classes of strategies: Randomized with Perfect recall (RP), Randomized Memoryless (RM), Deterministic with Perfect recall (DP), and Deterministic Memoryless (DM).

2.2 Probabilistic CTL

Probabilistic Computation Tree Logic (PCTL for short, [Hansson and Jonsson, 1994]) is a branching-time logic for specifying properties of probabilistic systems. Its syntax is defined by the grammar:

$$\Phi ::= a \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \neg\Phi \mid [\varphi]_{\bowtie q}$$
$$\varphi ::= \mathsf{X}\Phi \mid \Phi_1 \mathsf{U}^{\leqslant n} \Phi_2 \mid \Phi_1 \mathsf{W}^{\leqslant n} \Phi_2$$

where $a \in AP$, $\bowtie \in \{<, >, \leqslant, \geqslant\}$ is a binary comparison operator on the reals, $q \in [0,1]$, and $n \in \mathbb{N}^+ \cup \{\infty\}$. We will simply write U and W for $\mathsf{U}^{\leqslant\infty}$ and $\mathsf{W}^{\leqslant\infty}$ in the sequel. Let $\mathsf{F} := a \wedge \neg a$ denote false and $\mathsf{T} := \neg\mathsf{F}$ denote true. As usual, $\Diamond^{\leqslant n}\Phi := \mathsf{T}\mathsf{U}^{\leqslant n}\Phi$, $\Box^{\leqslant n}\Phi := \Phi\mathsf{W}^{\leqslant n}\mathsf{F}$, and $(\Phi_1 \implies \Phi_2) := (\neg\Phi_1 \vee \Phi_2)$. We will refer to Φ and φ as state and path formulas, respectively. The satisfaction relation $s \models \Phi$ is defined in the standard manner for the Boolean connectives. For the probabilistic operator, it is defined by: $s \models [\varphi]_{\bowtie q}$ iff $\Pr\{\pi \in \mathcal{S}^\omega(s) \mid \pi \models \varphi\} \bowtie q$, where $\mathcal{S}^\omega(s)$ denotes the set of infinite paths starting from s. For an MC D, we write $\mathsf{D} \models \Phi$ iff its initial state satisfies Φ, i.e., $\bar{s} \models \Phi$. The satisfaction relation for $\pi \in \mathcal{S}^\omega$ and φ is defined by: $\pi \models \mathsf{X}\Phi$ iff $\pi[1] \models \Phi$, $\pi \models \Phi_1\mathsf{U}^{\leqslant n}\Phi_2$ iff

$$\exists 0 \leqslant j \leqslant n.\pi[j] \models \Phi_2 \wedge \forall 0 \leqslant k < j.\pi[k] \models \Phi_1, \text{ and}$$

$\pi \models \Phi_1\mathsf{W}^{\leqslant n}\Phi_2$ iff $\pi \models \Phi_1\mathsf{U}^{\leqslant n}\Phi_2 \vee \forall 0 \leqslant i \leqslant n.\pi[i] \models \Phi_1$.

The until U and weak until W modalities are dual:

$$\begin{aligned} [\Phi_1\mathsf{U}\Phi_2]_{\geqslant q} &\equiv [(\Phi_1 \wedge \neg\Phi_2)\mathsf{W}(\neg\Phi_1 \wedge \neg\Phi_2)]_{\leqslant 1-q}, \\ [\Phi_1\mathsf{W}\Phi_2]_{\geqslant q} &\equiv [(\Phi_1 \wedge \neg\Phi_2)\mathsf{U}(\neg\Phi_1 \wedge \neg\Phi_2)]_{\leqslant 1-q}. \end{aligned} \quad (2)$$

Eq. (2) also holds if we: i) replace $\geqslant$ and $\leqslant$ with $>$ and $<$, and/or ii) replace U and W with bounded until $\mathsf{U}^{\leqslant n}$ and bounded weak until $\mathsf{W}^{\leqslant n}$, respectively. By applying duality laws, every PCTL formula can be transformed into an equivalent PCTL formula in *positive normal form*. A formula is in positive normal form, if negation only occurs adjacent to atomic propositions. In the sequel, we assume PCTL formulas to be in positive normal form.

2.3 Probabilistic Trees

In this subsection we introduce the notion of probabilistic trees [Katoen *et al.*, 2014], which are closely related to MCs, as each MC can be equivalently represented as a probabilistic tree. Let $A, B, \dots$ range over $\mathcal{P}(AP)$, where $\{a\}$ is abbreviated by a. Let ϵ be the empty sequence.

Definition 4. *A* probabilistic tree *(PT) over a set of states* S *is a tuple* $T = (W, L, P)$ *where* $\epsilon \notin W$*, and* $(W \cup \{\epsilon\}) \subseteq S^*$ *is* *prefix-closed and there exists one (and only one) $\pi \in W$ such that $|\pi| = 1$; $L : W \mapsto \mathcal{P}(AP)$ is a node labelling function; $P : W \mapsto Dist(W)$ is an edge labelling function, which is a partial function satisfying $P(\pi)(\pi') > 0$ iff $\pi' = \pi \cdot s \in W$ for some $s \in S$.*

The node π with $|\pi| = 1$ is referred to as the *root*[1], while all nodes π such that $\forall \pi' \in W. \pi \not\prec \pi'$ are referred to as the *leaves*. A PT $T = (W, L, P)$ is *total* iff for each $\pi_1 \in W$ there exists $\pi_2 \in W$ such that $\pi_1 \prec \pi_2$. T is *finite-depth* if there exists $n \in \mathbb{N}$ such that $|\pi| \leqslant n$ for each $\pi \in W$. The least such n is called the *depth* of T. Let $\mathbb{T}^\omega$ and $\mathbb{T}^*$ denote the sets of all total PTs and finite-depth PTs respectively.

Below defines a prefix relation on PTs [Katoen *et al.*, 2014], where $\upharpoonright$ denotes restriction.

Definition 5. *Let $T_i = (W_i, L_i, P_i)$ for i=1, 2 with $T_1 \in \mathbb{T}^*$ and $T_2 \in \mathbb{T}^\omega$. T_1 is a* prefix *of T_2, denoted $T_1 \preceq T_2$, iff $W_1 \subseteq W_2$ and $L_2 \upharpoonright W_1 = L_1$ and $P_2 \upharpoonright (W_1 \times W_1) = P_1$. Let $Pre_{fin}(T)$ denote the set of all prefixes of $T \in \mathbb{T}^\omega$.*

It is not hard to see each MC D can be unfolded into a total PT, denote $T(\mathsf{D})$. Conversely, any PT can be seen as an MC with infinite many states. Therefore, the satisfaction relation between PCTL formulas and total PTs can be defined directly. Such satisfaction relation can also be lifted to finite-depth PTs in $\mathbb{T}^*$ in a natural way: Let Φ be a PCTL formula and $T_1 \in \mathbb{T}^*$. T_1 satisfies Φ, denoted $T_1 \models \Phi$, iff $T_2 \models \Phi$ for any $T_2 \in \mathbb{T}^\omega$ such that $T_1 \in Pre_{fin}(T_2)$. Intuitively, the finite-depth PT T_1 suffices to conclude that Φ is satisfied regardless of the remaining execution. We say that Φ always has finite-depth witnesses iff for any MC D such that $\mathsf{D} \models \Phi$, there exists $T_1 \in Pre_{fin}(T(\mathsf{D}))$ such that $T_1 \models \Phi$.

3 Safety and Co-safety Properties

The distinction of safety and liveness properties is pivotal for verifying reactive systems. As Lamport introduced in 1977 [Lamport, 1977; Alford *et al.*, 1985], safety properties assert that something "bad" never happens, while liveness properties require that something "good" will happen eventually. For stochastic properties, PCTL formulas have been classified into safety and liveness in [Katoen *et al.*, 2014]. In this paper, we are mainly interested in co-safe PCTL, i.e., formulas whose negations are safe PCTL.

Definition 6. *The* safe and co-safe fragments *of* PCTL, *denoted PCTL_s and PCTL_{cs} respectively, are defined by the following grammar, where $n \in \mathbb{N}$, $\phi, \phi_1, \ldots$ and $\Phi, \Phi_1, \ldots$ range over formulas in PCTL_s and PCTL_{cs}, respectively.*

$$\phi ::= \mathsf{F} \mid \mathsf{T} \mid a \mid \neg a \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid [\mathsf{X}\phi]_{\geqslant q} \mid$$
$$[\phi_1 \mathsf{W} \phi_2]_{\geqslant q} \mid [\Phi_1 \mathsf{U} \Phi_2]_{\leqslant q} \mid [\phi_1 \mathsf{W}^{\leqslant n} \phi_2]_{\bowtie q} \mid [\Phi_1 \mathsf{U}^{\leqslant n} \Phi_2]_{\bowtie q}$$
$$\Phi ::= \mathsf{F} \mid \mathsf{T} \mid a \mid \neg a \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid [\mathsf{X}\Phi]_{> q} \mid$$
$$[\phi_1 \mathsf{W} \phi_2]_{< q} \mid [\Phi_1 \mathsf{U} \Phi_2]_{> q} \mid [\Phi_1 \mathsf{W}^{\leqslant n} \Phi_2]_{\bowtie q} \mid [\phi_1 \mathsf{U}^{\leqslant n} \phi_2]_{\bowtie q}$$

We note that for bounded until and weak until, $\bowtie$ can be any operator in $\{<, >, \leqslant, \geqslant\}$. Since $\neg[\mathsf{X}\phi]_{\geqslant q} \equiv [\mathsf{X}\phi]_{< q} \equiv [\mathsf{X}\neg\phi]_{>1-q}$, together with Eq. (2), it is easy to check that the negation of each formula in PCTL_s has an equivalent representation in PCTL_{cs} and vice versa. Also note the membership of PCTL_s and PCTL_{cs} can be determined pure syntactically. For instance, F and T are both safe and co-safe; $[\Box\phi]_{\geqslant q} = [\phi \mathsf{W} \mathsf{F}]_{\geqslant q}$ is safe, provided $\phi \in \mathrm{PCTL}_s$, while $[\Diamond\Phi]_{> q} = [\mathsf{T} \mathsf{U} \Phi]_{> q}$ is co-safe, provided $\Phi \in \mathrm{PCTL}_{cs}$. In this way, PCTL_{cs} subsumes reachability properties with strict lower bounds. Note the formula in Eq. (1) also belongs to PCTL_{cs}, provided that we have atomic propositions such as $c \geqslant 20$ and $c = 0$. Below is a theorem from [Katoen *et al.*, 2014] showing that the definitions of PCTL_s and PCTL_{cs} are both sound and complete.

Theorem 1 ([Katoen *et al.*, 2014, Thm. 4 and 5])**.** *All formulas in PCTL_s (PCTL_{cs} resp.) are safe (co-safe resp.), while each safe (co-safe resp.) property expressible in* PCTL *has an equivalent formula in PCTL_s (PCTL_{cs} resp.).*

Expressiveness of co-safe PCTL We mention that many patterned properties classified in [Grunske, 2008] can be expressed in PCTL_{cs}. For instance,

- *Probabilistic Invariance*. The probability of no error occurring in the next n steps is at least 0.99: $[\Box^{\leqslant n}(no\ error)]_{\geqslant 0.99}$.
- *Probabilistic Existence*. If an error occurs, it will be solved eventually with probability greater than 0.95: $error \implies [\Diamond solved]_{>0.95}$.
- *Probabilistic Until*. With probability greater than 0.95, the alarm should be on until the problem is solved: $[alarmOn \mathsf{U} (alarmOff \wedge solved)]_{>0.95}$.
- *Probabilistic Response*. With probability greater than 0.99, during the lifetime (m) of a robot, whenever it is out of battery (A), it will be fully recharged (B) in the next n steps: $[\Box^{\leqslant m}(A \implies [\Diamond^{\leqslant n} B]_{\geqslant 1})]_{\geqslant 0.99}$.

Particularly, since PCTL_{cs} is closed under conjunction, multi-objective properties [Wakuta and Togawa, 1998; Refanidis and Vlahavas, 2003; Wiering and De Jong, 2007; Khouadjia *et al.*, 2013; Wray *et al.*, 2015] can be directly expressed in PCTL_{cs} by properly choosing atomic propositions. If given more than one objective, we can put them together in conjunction and find a strategy fulfilling all of them at the same time.

Remark 1. *According to Definition 6, $[\Diamond solved]_{>0.95}$ is co-safe, while $[\Diamond solved]_{\geqslant 0.95}$ is not. Such a subtle difference between $>$ and $\geqslant$ is of theoretic importance, as the probability of reaching "solved" may converge to 0.95 in infinite steps, but the probability of reaching "solved" is strictly less than 0.95 in any finite steps. However, as shown in [Chatterjee and Henzinger, 2008], in a finite* MDP *where all transition probabilities are rational, reachability probabilities can be computed precisely in finite steps. Therefore, in practice, the algorithms we shall introduce later can also be applied to properties with non-strict bounds, i.e., those definable by the PCTL_{cs} grammar in Definition 6 with additional operators $[\mathsf{X}\Phi]_{\geqslant q}$, $[\phi_1 \mathsf{W} \phi_2]_{\leqslant q}$, and $[\Phi_1 \mathsf{U} \Phi_2]_{\geqslant q}$.*

4 Stochastic Games with Perfect Recall

In this section we introduce stochastic games [Baier *et al.*, 2004; Kucera and Strazovský, 2008; Brázdil *et al.*, 2006]

[1] For technical reasons, ϵ is excluded from the tree.

formally. If general PCTL objectives are considered, stochastic games with perfect recall turn out to be highly undecidable [Brázdil *et al.*, 2006] even for finite models. Therefore, we mainly focus on games whose objectives are restricted to PCTL_{cs}. Even though DP $\subsetneq$ RP, these two sets of strategies are usually studied separately, as winning strategies with or without randomization make differences for PCTL objectives [Baier *et al.*, 2004]. We show in this section that stochastic games with PCTL_{cs} objectives are decidable even if strategies in DP are considered. Below defines stochastic games formally:

Definition 7. *Let* M *be an* MDP *as in Definition 2 and* $\Phi \in \mathrm{PCTL}_{cs}$*. Let XX range over the set {RP, RM, DP, DM}. The stochastic game over* M *has a winning strategy in XX for the given objective* Φ *iff there exists a XX-strategy* σ *such that* $\mathsf{M}_\sigma \models \Phi$*. A XX stochastic game answers whether or not a winning strategy in XX exists for a given objective.*

We shall fix an MDP M throughout the remaining part of this paper. As we mentioned, DP stochastic games are undecidable in general for objectives specified by PCTL. This is mainly due to the fact that some objectives only have winning strategies using infinite memories. By restricting to PCTL_{cs} objectives, however, finite memories suffice as shown in the following lemma [Katoen *et al.*, 2014].

Lemma 1. *Formulas in* PCTL_{cs} *have finite-depth witnesses.*

Lemma 1 ensures that a winning strategy only requires finite memories for objectives in PCTL_{cs}. However, it does not give us an upper bound for the size of the memory. Fortunately, such bounds can be established. For this, we first introduce some notations. In the following, we assume all probabilities are expressed as irreducible fractions.

Definition 8. *Let* m *be the number of transitions of* M*;* d *the largest denominator among all transition probabilities appearing in* M*;* θ *the least common multiple of all denominators;* p *the largest numerator of all transition probabilities; and* $\gamma = d^{4m} \cdot \theta \cdot p$*.*

It was shown in [Chatterjee and Henzinger, 2008] that maximal and minimal reachability probabilities can be computed precisely in at most $\mathcal{O}(\gamma^2)$ iterations. We show in the following lemma that such bound also holds for the size of memory used by a winning strategy, if exists, in a DP stochastic game.

Lemma 2. *Given an objective* $\Phi \in \mathrm{PCTL}_{cs}$*, whenever the DP stochastic game has a winning strategy* σ*, there exists* $T_1 \in \mathit{Pre}_{\mathit{fin}}(\mathsf{M}_\sigma)$ *with depth* $\mathcal{O}(|\Phi| \cdot \gamma^2)$ *such that* $T_1 \models \Phi$*, where* $|\Phi|$ *denotes the length of* Φ*.*

Proof. The proof is by structural induction on Φ. Let $\mathit{depth}(\Phi)$ denote the least depth of finite-depth witnesses of Φ in M. We only show proofs of several non-trivial cases:

- $\Phi = [\Phi_1 \mathsf{U} \Phi_2]_{>q}$. Then we show that $\mathit{depth}(\Phi) = \mathit{depth}(\Phi_1) + c + \mathit{depth}(\Phi_2)$, where $c \in \mathcal{O}(\gamma^2)$. Essentially, the result in [Chatterjee and Henzinger, 2008] ensures that reachability probabilities in an MC D can be computed precisely by considering a prefix of $T(\mathsf{D})$ of depth $c \in \mathcal{O}(\gamma^2)$. By induction hypothesis, a tree of depth $\mathit{depth}(\Phi_1)$ suffices to witness all states satisfying Φ_1. According to [Chatterjee and Henzinger, 2008], by extending all prefixes for c steps, we make sure all strategies which can reach states satisfying Φ_2 with probability $> q$ can be found. To ensure that all states satisfying Φ_2 will be found, further extensions up to $\mathit{depth}(\Phi_2)$ would be enough by induction hypothesis.
- $\Phi = [\phi_1 \mathsf{W} \phi_2]_{<q}$. By duality law, $\Phi \equiv [(\phi_1 \wedge \neg\phi_2)\mathsf{U}(\neg\phi_1 \wedge \neg\phi_2)]_{>1-q}$. We show that $\mathit{depth}(\Phi) = \mathit{depth}(\neg\phi_2) + c + \mathit{depth}(\neg\phi_1 \wedge \neg\phi_2)$. Note $\neg\phi_1$ and $\neg\phi_2$ are co-safe formulas. Therefore, prefixes of depth $\mathit{depth}(\neg\phi_2)$ suffice to find all states satisfying $\neg\phi_2$ by induction hypothesis. Consequently, all states satisfying $\phi_1 \wedge \neg\phi_2$ can be found, which is a subset of states satisfying $\neg\phi_2$. The remaining proof is then similar as the above case.

This completes the proof. $\square$

Due to Lemma 2, the decidability of DP stochastic games with PCTL_{cs} objectives is easy to establish.

Theorem 2. *DP stochastic games with* PCTL_{cs} *objectives are decidable and in 3-EXPTIME.*

Proof. Let M and $\Phi \in \mathrm{PCTL}_{cs}$ be given. First, we unfold M to a PT up to a certain step. Let M^i denote the resultant model at the i-th step of unfolding. In M^i, it suffices to enumerate all possible DM strategies, which are finitely many. We continue unfolding M until either a winning strategy is found, or for each state $s \in \mathcal{S}$ and path formula φ in Φ, the change of the probability of s satisfying φ is less than $\frac{1}{2\gamma}$ [Chatterjee and Henzinger, 2008], comparing to the previous valuation. As $\Phi \in \mathrm{PCTL}_{cs}$, the termination of this process is guaranteed by Lemma 2.

The above procedure will cause a triply exponential blow-up: Since we may need to unfold M up to an exponential number i of steps and hence may result in an MDP M^i doubly exponentially larger than the original model. Furthermore, the number of DM strategies of M^i is exponentially many with respect to the size of M^i. Therefore, the whole algorithm is in 3-EXPTIME. $\square$

It is not surprising that the problem is of high complexity, since even DM stochastic games are NP-complete, and DP stochastic games with linear time objectives are elementary [Baier *et al.*, 2004].

5 Stochastic Games with Perfect Recall and Randomization

In this section we focus on RP stochastic games with PCTL_{cs} objectives. Unfortunately, Theorem 2 does not hold if RP strategies are considered. This is because randomized strategies can assign arbitrary probabilities to successor states, hence parameters d, θ, and p in Definition 8 cannot be defined directly. On the other hand, *robustness* of strategies has been argued by many authors to be a natural requirement in probabilistic systems, see for instance [Nilim and Ghaoui, 2005; Kucera and Strazovský, 2008]. In the remaining of this section we will define robustness of strategies along the line of [Kucera and Strazovský, 2008] and then show that it is decidable to determine whether or not a robust winning strategy exists. Finally, we show that as special cases of robust RP stochastic games (when tolerance is equal to 0), RP stochastic games are decidable too. As we shall show, this also implies that satisfiability of PCTL_{cs} is decidable.

We follow definitions in [Kucera and Strazovský, 2008] and define strategy robustness for RP stochastic games.

Definition 9. *Let σ, σ' be two RP strategies of a stochastic game with respect to an objective $\Phi \in \mathrm{PCTL}_{cs}$. We say that*

1. *σ' is a δ-perturbation of σ if for all $\pi \in \mathcal{S}^*$ with $\pi\downarrow \in \mathcal{S}_\Box$ and $t \in \mathcal{S}$: a) $|\sigma(\pi)(t) - \sigma'(\pi)(t)| \leqslant \delta$ and b) $\sigma(\pi)(t) = 0$ iff $\sigma'(\pi)(t) = 0$, where $\delta \in (0,1) \cap \mathbb{Q}$;*
2. *σ is a δ-robust winning strategy iff for any RP strategy σ', whenever σ' is a δ-perturbation of σ, it is a winning strategy of the stochastic game too.*

Intuitively, a strategy σ' is a δ-perturbation of σ iff at each step the probability of σ' choosing a state is within $[q-\delta, q+\delta]$, where q is the probability of σ choosing the same state. Moreover, for technical reasons, an extra condition is imposed, which essentially guarantees that all states chosen by σ with positive probabilities should also be chosen by σ' with positive probabilities. This condition is necessary to preserve qualitative structures, i.e., topologies of MCs [Courcoubetis and Yannakakis, 1995] induced by strategies which are δ-perturbations of each other.

In the sequel we focus on robust RP stochastic games with PCTL_{cs} objectives: Given a $\delta \in (0,1) \cap \mathbb{Q}$ and an objective $\Phi \in \mathrm{PCTL}_{cs}$, decide whether or not a δ-robust RP winning strategy exists for the stochastic game. We show that robust RP stochastic games can be solved by using the algorithm in [Kucera and Strazovský, 2008], denoted `MRGame`, as an oracle. `MRGame` solves RM stochastic games with PCTL objectives by reducing them to the decidable problem – first order theory of reals $(\mathbb{R}, +, *, \leqslant)$ [Tarski, 1951]. For similar reasons as in Section 4, `MRGame` cannot be extended to deal with RP strategies directly. Below is a theorem showing another main result of this paper:

Theorem 3. *For a given $\delta \in (0,1) \cap \mathbb{Q}$, robust RP stochastic games with PCTL_{cs} objectives are decidable.*

So far, we require $\delta > 0$. In the sequel we discuss the special case when $\delta = 0$, i.e., RP stochastic games. It is easy to see that arguments used in Theorem 3 do not apply when $\delta = 0$, as in principle a strategy can assign arbitrarily small probabilities to some states. However, we can prove a lemma showing that if there is a winning strategy for an RP stochastic game, then we can always find a winning strategy where the largest denominator of all probabilities assigned by the strategy to all states is bounded.

Lemma 3. *Let $\Phi \in \mathrm{PCTL}_{cs}$ be an objective. There is an RP winning strategy for Φ, iff there exists $D \in \mathbb{N}^+$ and a winning strategy σ such that for any $\pi \in \mathcal{S}^*$ with $\pi\downarrow \in \mathcal{S}_\Box$ and $t \in \mathcal{S}$, $\sigma(\pi)(t) = \frac{k}{D}$, where $k \in [0, D]$ is an integer.*

Proof. The sufficiency is trivial. To prove the necessity part, let σ be a winning strategy for Φ and $T_1 \in \mathit{Pre}_{fin}(T(\mathsf{M}_\sigma))$ such that $T_1 \models \Phi$. For each node π in T_1, we denote $\Pr_\pi^{T_1}(\varphi)$ the probability of π satisfying φ in T_1, where φ is a path formula in Φ. Let d_σ be the greatest denominator of all $\Pr_\pi^{T_1}(\varphi)$ including all bounds in Φ and od_{M} the greatest out-degree of states in M. For $k \in \mathbb{N}$ and $q \in [0,1]$, we denote $\lfloor q \rfloor_k = \frac{k'}{k}$, where k' is the greatest integer such that $\frac{k'}{k} \leqslant q$. Let σ' be a strategy such that $\sigma'(\pi)(t) = \lfloor \sigma(\pi)(t) \rfloor_D$ and $\sigma'(\pi)(\mathit{dummy}) = 1 - \sum_{t \in \mathcal{S}} \sigma'(\pi)(t)$, where D is any integer greater than $2 \cdot od_{\mathsf{M}} \cdot d_\sigma$ and dummy denotes a deadlock state. The reason to introduce dummy is that by letting $\sigma'(\pi)(t) = \lfloor \sigma(\pi)(t) \rfloor_D$ for each π and t, the sum of probabilities assigned by σ' to all successors may be less than 1. Therefore, we have to add an extra transition to the dummy state with the remaining probability to make sure that σ' gives rise to a distribution at each step. Let T_1' be the prefix of $\mathsf{M}_{\sigma'}$ of the same depth as T_1. It shall be easy to see that for each node π and path formula φ in Φ,

$$|\Pr_\pi^{T_1}(\varphi) - \Pr_{\pi'}^{T_1'}(\varphi)| \leqslant \frac{od_{\mathsf{M}}}{D} < \frac{1}{2 \cdot d_\sigma}.$$

As d_σ is the greatest denominator of all $\Pr_\pi(\varphi)$ including all bounds in Φ, $T_1 \models \Phi$ implies $T_1' \models \Phi$, hence σ' is also a winning strategy. $\Box$

Let $D \in \mathbb{N}^+$ be a positive integer. We call a strategy σ a D-strategy if all probabilities assigned by σ to states are in form $\frac{k}{D}$ with $k \in [0, D]$ being an integer. Given a D and i, it is easy to see that the number of D-strategies of M^i in DM is finite. Below is a theorem showing that RP stochastic games with PCTL_{cs} objectives are decidable, which can be proved in a similar way as Theorem 3.

Theorem 4. *RP stochastic games with PCTL_{cs} objectives are decidable.*

As a byproduct of Theorem 4, the satisfiability of PCTL_{cs} is decidable.

Theorem 5. *Let Φ be a formula in PCTL_{cs}. The following problem is decidable: Checking whether Φ is satisfiable, i.e., whether there exists an MC D such that $\mathsf{D} \models \Phi$.*

To the best of our knowledge, until now, the problem of deciding whether or not a given PCTL formula is satisfiable is still open and only partial results are known [Brázdil *et al.*, 2008; Bertrand *et al.*, 2012].

6 Conclusion and Future Work

We have considered stochastic games with branching-time objectives, where strategies are deterministic or randomized, and always with perfect recall. While undecidable in general, we showed that the problem can be solved if restricted to PCTL_{cs} objectives, which cover many interesting properties in practice as we have shown. Robust stochastic games were also studied. Our result indicates that satisfiability of PCTL_{cs} is decidable.

There are several directions for future work, e.g., the complexity of these problems and other fragments of PCTL which make stochastic games with perfect recall decidable.

Acknowledgments

The authors are supported by Australian Research Council under Grant DP130102764, the National Natural Science Foundation of China (Grant Nos. 61428208, 61472473 and 61361136002), AMSS-UTS Joint Research Laboratory for Quantum Computation, Chinese Academy of Sciences, and the CAS/SAFEA International Partnership Program for Creative Research Team.

References

[Alford *et al.*, 1985] M. W. Alford, J. P. Ansart, G. Hommel, L. Lamport, B. Liskov, G. P. Mullery, and F. B. Schneider. *Distributed Systems: Methods and Tools for Specification*, volume 190 of *LNCS*. Springer, 1985.

[Baier and Katoen, 2008] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. MIT Press, 2008.

[Baier *et al.*, 2004] Christel Baier, Marcus Größer, Martin Leucker, Benedikt Bollig, and Frank Ciesinski. Controller synthesis for probabilistic systems. In *TCS*, volume 155 of *IFIP*, pages 493–506. Springer, 2004.

[Baier *et al.*, 2012] Christel Baier, Tomás Brázdil, Marcus Größer, and Antonín Kucera. Stochastic game logic. *Acta Inf.*, 49(4):203–224, 2012.

[Bertrand *et al.*, 2012] Nathalie Bertrand, John Fearnley, and Sven Schewe. Bounded satisfiability for PCTL. In *CSL*, volume 16 of *LIPIcs*, pages 92–106. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2012.

[Brázdil *et al.*, 2006] Tomás Brázdil, Václav Brozek, Vojtech Forejt, and Antonín Kucera. Stochastic games with branching-time winning objectives. In *LICS*, pages 349–358. IEEE, 2006.

[Brázdil *et al.*, 2008] Tomás Brázdil, Vojtech Forejt, Jan Kretínský, and Antonín Kucera. The satisfiability problem for probabilistic CTL. In *LICS*, pages 391–402. IEEE Computer Society, 2008.

[Chatterjee and Henzinger, 2008] Krishnendu Chatterjee and Thomas A. Henzinger. Value iteration. In Orna Grumberg and Helmut Veith, editors, *25 Years of Model Checking*, pages 107–138. Springer, 2008.

[Chen and Lu, 2007] Taolue Chen and Jian Lu. Probabilistic alternating-time temporal logic and model checking algorithm. In *FSKD*, pages 35–39. IEEE, 2007.

[Courcoubetis and Yannakakis, 1995] Costas Courcoubetis and Mihalis Yannakakis. The complexity of probabilistic verification. *J. ACM*, 42(4):857–907, 1995.

[Delage and Mannor, 2007] Erick Delage and Shie Mannor. Percentile optimization in uncertain Markov decision processes with application to efficient exploration. In *ICML*, volume 227, pages 225–232. ACM, 2007.

[Grunske, 2008] Lars Grunske. Specification patterns for probabilistic quality properties. In *ICSE*, pages 31–40. ACM, 2008.

[Halmos, 1974] Paul Richard Halmos. *Measure theory*, volume 1950. Springer, 1974.

[Hansson and Jonsson, 1994] Hans Hansson and Bengt Jonsson. A logic for reasoning about time and reliability. *Formal Aspects of Computing*, 6:102–111, 1994.

[Huang and Luo, 2013] Xiaowei Huang and Cheng Luo. A logic of probabilistic knowledge and strategy. In *AAMAS*, pages 845–852. IFAAMAS, 2013.

[Huang *et al.*, 2012] Xiaowei Huang, Kaile Su, and Chenyi Zhang. Probabilistic alternating-time temporal logic of incomplete information and synchronous perfect recall. In *AAAI*. AAAI Press, 2012.

[Katoen *et al.*, 2014] J.-P. Katoen, Lei Song, and Lijun Zhang. Probably safe or live. In *CSL-LICS*, pages 55:1–10. ACM, 2014.

[Khouadjia *et al.*, 2013] Mostepha Redouane Khouadjia, Marc Schoenauer, Vincent Vidal, Johann Dréo, and Pierre Savéant. Pareto-based multiobjective AI planning. In *IJCAI*. IJCAI/AAAI, 2013.

[Kucera and Strazovský, 2008] Antonín Kucera and Oldrich Strazovský. On the controller synthesis for finite-state Markov decision processes. *Fundam. Inform.*, 82(1-2):141–153, 2008.

[Lacerda *et al.*, 2014] Bruno Lacerda, David Parker, and Nick Hawes. Optimal and dynamic planning for Markov decision processes with co-safe LTL specifications. In *IROS*, pages 1511–1516. IEEE, 2014.

[Lamport, 1977] L. Lamport. Proving the correctness of multiprocess programs. *IEEE TSE*, 3(2):125–143, 1977.

[Mausam and Kolobov, 2012] Mausam and Andrey Kolobov. *Planning with Markov Decision Processes: An AI Perspective*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2012.

[Nilim and Ghaoui, 2005] Arnab Nilim and Laurent El Ghaoui. Robust control of Markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

[Pineda *et al.*, 2013] Luis Enrique Pineda, Yi Lu, Shlomo Zilberstein, and Claudia V. Goldman. Fault-tolerant planning under uncertainty. In *IJCAI*. IJCAI/AAAI, 2013.

[Puterman, 1994] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1994.

[Refanidis and Vlahavas, 2003] Ioannis Refanidis and Ioannis Vlahavas. Multiobjective heuristic state-space planning. *Artificial Intelligence*, 145(1):1–32, 2003.

[Rudin, 2006] Walter Rudin. *Real and complex analysis*. Tata McGraw-Hill Education, 2006.

[Tarski, 1951] Alfred Tarski. A decision method for elementary algebra and geometry. *Bulletin of the American Mathematical Society*, 59, 1951.

[Teichteil-Königsbuch, 2012] Florent Teichteil-Königsbuch. Path-constrained Markov decision processes: bridging the gap between probabilistic model-checking and decision-theoretic planning. In *ECAI*, volume 242 of *Frontiers in Artificial Intelligence and Applications*, pages 744–749. IOS Press, 2012.

[Wakuta and Togawa, 1998] K Wakuta and K Togawa. Solution procedures for multi-objective Markov decision processes. *Optimization*, 43(1):29–46, 1998.

[Wiering and De Jong, 2007] Marco A Wiering and Edwin D De Jong. Computing optimal stationary policies for multi-objective Markov decision processes. In *ADPRL*, pages 158–165. IEEE, 2007.

[Wray *et al.*, 2015] Kyle Hollins Wray, Shlomo Zilberstein, and Abdel-Illah Mouaddib. Multi-objective MDPs with conditional lexicographic reward preferences. In *AAAI*, pages 3418–3424. AAAI Press, 2015.

[Xu and Mannor, 2011] Huan Xu and Shie Mannor. Probabilistic goal Markov decision processes. In *IJCAI*, pages 2046–2052. IJCAI/AAAI, 2011.