A generalization of the concept of Markov decision process to imprecise probabilities

David Harmanec
Medical Computing Laboratory
Department of Computer Science
School of Computing
National University of Singapore

Abstract

This paper is a first step towards generalizing the concept of Markov decision process to imprecise probabilities. A concept of generalized Markov decision process is defined and a solution procedure for it presented.

Keywords. generalized Markov decision process, sequential decision making, imprecise probabilities, interval utilities

1 Introduction

Every day we are faced with the need to make decisions that depend on our past actions and that will influence our well-being and options available to us in the future. We have to decide whether to drive to work or to use public transportation; a doctor has to decide about a treatment plan for a patient; and so on. If we want to make good decisions in situations like these, we cannot think about them individually in isolation, but rather in the context of previous and future decisions. For example, if I spend all my money today, I will loose the option to buy any food tomorrow.

The concept of Markov decision process has been very useful in helping to rigorously approach and solve this type of problem [7]. There are, however, two major obstacles in applying Markov decision process methodology in practical situations. The first problem is, that a large number of precise probability estimates is necessary to fully specify a model. This is a general problem of probabilistic modeling, but it is much more pronounced in a dynamical situation. Second problem is that the time it takes to solve a model of realistic size is, generally, quite large. It seems that if the concept of Markov decision process was generalized to allow imprecise probability and utility specifications both of these problems might be helped, at least in some situations. Imprecise probabilities framework allows one to use as little or as many probability judgements as desired or available. Of course, the conclusions drawn on the basis of less information will be less precise. To reduce computational complexity, one can try to abstract less important details away from a given model. The resulting model is smaller and, hence, easier to solve. Even if the original model was formulated in terms of precise probabilities, the abstraction, in general, imposes only constraints on probabilities in the abstract model. A generalization to imprecise probabilities overcomes this obstacle. One needs to be careful, however, to abstract in a "sensible" way as too much imprecision makes many actions/policies incomparable. In this way what is gained on model details is lost on the need to keep track of incomparable options.

This paper is a first step towards a generalization of Markov decision process theory to imprecise probabilities. It develops a formalism and interpretation of generalized Markov decision process and a solution method that is a generalization of backward induction method from the classical case.

2 Notation and terminology

I assume the reader to be familiar with the theory of imprecise probabilities (see, e.g., [9]), some knowledge of the theory of Markov decision processes (see, e.g., [7, 10]) is also useful. This section serves only as an overview of the notation and terminology I use in the rest of the paper. It is mostly derived from [9].

S denotes a finite set representing all possible and mutually exclusive states of a system of interest (world, etc.) at a particular time. In this paper I assume that S is constant over time. A *gamble* g is a mapping from

S to the set of real numbers:

$$g : S \longrightarrow \mathbb{R}$$

The gamble represents a reward (utility) a decision maker will receive at state $s \in S$ if the state s is the actual state of the system. For any subset $A \subseteq S$, I use A to denote both the subset and its characteristic function interpreted as a gamble. For any real number λ, I use λ to denote both the real number and the constant gamble assigning λ to each state. In both cases it is clear from the context which is which. The addition and multiplication by a scalar value of gambles is defined point wise (e.g., $(g+h)(s) = g(s) + h(s)$ for all states s). The set of all gambles on S is denoted $\mathcal{L}(S)$.

Let $\mathcal{K}$ denote a set of gambles on S. A *lower (upper) prevision* $\underline{P}$ $(\overline{P})$ on $\mathcal{K}$ is a mapping from $\mathcal{K}$ to the set of real numbers:

$$\underline{P} : \mathcal{K} \longrightarrow \mathbb{R}, \overline{P} : \mathcal{K} \longrightarrow \mathbb{R}$$

It represents the decision maker's lower (upper) expected reward for each gamble from $\mathcal{K}$ (and, consequently, supremum (infimum) buying (selling) price for the gamble). Assume that a lower prevision $\underline{P}$ on $\mathcal{K}$ is given, a mapping

$$\overline{P} : -\mathcal{K} \longrightarrow \mathbb{R}$$

defined as

$$\overline{P}(g) = -\underline{P}(-g)$$

for all gambles in $\mathcal{K}$, is called a *conjugate* upper prevision. A lower prevision $\underline{P}$ is *coherent* if the following three conditions hold

$$\begin{aligned} \underline{P}(g) &\geq \inf\{g(s) \mid s \in S\} \text{ for all } g \in \mathcal{K}, \\ \underline{P}(\lambda g) &= \lambda \underline{P}(g) \text{ for all } g \in \mathcal{K}, \lambda > 0, \\ \underline{P}(g+h) &\geq \underline{P}(g) + \underline{P}(h) \text{ for all } g, h \in \mathcal{K}. \end{aligned}$$

An upper prevision $\overline{P}$ is coherent if its conjugate lower prevision is coherent. A *natural extension* of a coherent lower prevision $\underline{P}$ to a gamble $o \notin \mathcal{K}$ is defined by the formula

$$\underline{P}(o) = \sup\left\{\mu \;\middle|\; \begin{array}{c} (o-\mu) \geq \\ \geq \sum_{i=1}^{k} \lambda_i \left(g_i - \underline{P}(g_i)\right), \\ \lambda_i \geq 0, k \geq 1, g_i \in \mathcal{K} \end{array}\right\}.$$

The natural extension is the smallest coherent extension of $\underline{P}$ to $\mathcal{K} \cup \{o\}$. Basically, it expresses what the current belief implies about the belief for o.

A lower (upper) probability is a (coherent) lower (upper) prevision defined on all subsets of S. A lower probability is *2-monotone* if it holds that

$$\underline{P}(A) + \underline{P}(B) \leq \underline{P}(A \cup B) + \underline{P}(A \cap B).$$

2-monotone lower probability is coherent, but the inverse does not hold in general. If a lower probability is 2-monotone its natural extension to any gamble g on S can be computed by Choquet integral [1]:

$$\begin{aligned} \underline{P}(g) &= (c)\int_S g d\underline{P} \\ &= \int_{-\infty}^{0} \left[\underline{P}(\{s \in S \mid g(s) \geq t\}) - 1\right] dt \\ &+ \int_{0}^{\infty} \underline{P}(\{s \in S \mid g(s) \geq t\})\, dt \\ &= \lambda_1 \\ &+ \sum_{i=2}^{n} \left[(\lambda_i - \lambda_{i-1})\, \underline{P}(\{s \in S \mid g(s) \geq \lambda_i\})\right], \end{aligned}$$

where λ_i's are such that $\lambda_i = g(s_i), \lambda_i \leq \lambda_{i+1}$ for an appropriate ordering $(s_1, s_2, \ldots, s_n)$ of S.

$\mathcal{I}(\mathbb{R})$ denotes the set of all closed bounded intervals of real numbers. For any $\widehat{i} \in \mathcal{I}(\mathbb{R})$, $\widehat{i}_l$ and $\widehat{i}_u$ denote, respectively, the lower and upper bound of $\widehat{i}$, i.e., $\widehat{i} = \left[\widehat{i}_l, \widehat{i}_u\right]$. For $\widehat{i}, \widehat{j} \in \mathcal{I}(\mathbb{R})$ and $\lambda \geq 0$ we define $\widehat{i} + \widehat{j} = \left[\widehat{i}_l + \widehat{j}_l, \widehat{i}_u + \widehat{j}_u\right]$ and $\lambda \widehat{i} = \left[\lambda \widehat{i}_l, \lambda \widehat{i}_u\right]$.

3 Generalized Markov decision problem

This section introduces a subclass of the class of dynamical decision problems I am attempting to provide tools for in this work. A general outline of the problem of interest is illustrated in Figure 1. A decision

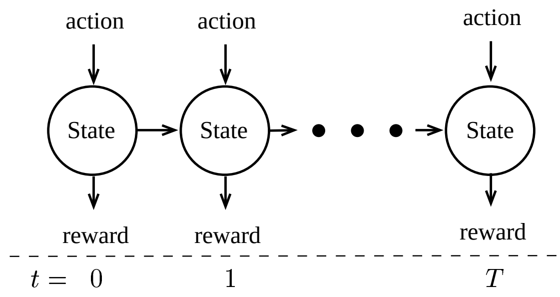

Figure 1: Dynamic Decision Problem

maker is facing a dynamical system that develops over time. The system can be in one and only one of the states from S at a particular point in time.[1] The decision maker can choose an action from a predefined set of options A at each of finitely many decision epochs.

[1] The assumption that the state space S is constant helps to simplify (already complex) notation. Both the definitions and the algorithm generalize in a straightforward manner to the case when the state space is dynamic.

The number of decision epochs is denoted T. If the system is in state s at the beginning of a decision epoch t, the state s' into which it transits at the beginning of the next decision epoch is influenced by the action a the decision maker executes at stage t. The process is called Markov because its development depends only on the current state and action and not on the rest of its history. The transitions during one period for a particular action are illustrated in Figure 2. The dashed arcs signify the fact that the decision maker may have stronger information for sets of destinations than for a single state. Depending on the

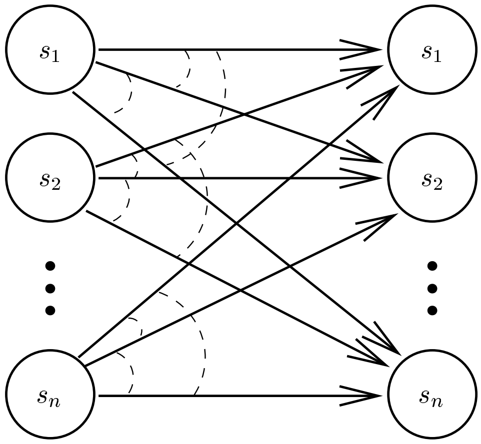

Figure 2: Time slice of a generalized Markov decision process for a particular action

action a taken, current decision epoch t, and the current state s the decision maker receives a reward from $\widehat{r}(s,a,t)$ (in general, the decision maker has only imprecise information about the actual reward received in a particular situation). At the last stage no action is taken so the reward depends only on state. The goal is to find an "optimal" policy π, i.e., mapping from state-time pairs into actions that maximizes the expected cumulative reward over all decision epochs. In the classical case the uncertainty regarding the next state given current state and action is expressed as a (conditional) probability distribution over states. In the generalized case, we express our beliefs regarding the next state as a coherent lower prevision on the set $\mathcal{L}(S)$ of all gambles on S. Formally:

Definition 1 *A tuple $\langle T, S, A, \underline{P}, \widehat{r}, \widehat{r}_T \rangle$ where T is a positive natural number, S and A are finite non-empty sets, $\underline{P}$ is a mapping from $\mathcal{L}(S) \times S \times A \times \{0,1,\ldots,T-1\}$ to the set of real numbers such that $\underline{P}(\cdot,s,a,t)$ is a coherent lower prevision on $\mathcal{L}(S)$ for any $s \in S$, $a \in A$, and any $t \in \{0,1,\ldots,T-1\}$, $\widehat{r}$ is a mapping from $S \times A \times \{0,1,\ldots,T-1\}$ to the set of bounded intervals of real numbers, and $\widehat{r}_T$ is a mapping from S to the set of bounded intervals of real numbers, is called a* generalized Markov decision problem.

In practical situation the lower previsions will not be defined for all gambles explicitly, but extended from a class of gambles (in many situations just the subsets of S) using natural extension or Choquet integral.

Example 1 *Consider the following simple generalized Markov decision problem*

$$\mathbf{M} = \langle 1, \{s_1, s_2\}, \{\alpha_1, \alpha_2\}, \underline{P}^{vac}, [0,1], [0,1] \rangle .$$

As $T = 1$ the problem has only one decision stage and, hence, is only a degenerate case of sequential decision problem. The system modeled by $\mathbf{M}$ can either be in state s_1 or s_2. The (imaginary) decision maker can choose between the act α_1 and the act α_2, and has no idea what will be the next state in either case, i.e., $\underline{P}^{vac}(\cdot,s,a,0)$ is a vacuous lower prevision for any s and a.[2] Both $\widehat{r}$ and $\widehat{r}_1$ are constant intervals $[0,1]$. This means that no matter in which state the system will be or which action is carried out the decision maker expects to receive one time reward between 0 and 1 in both states and at both time points.

Definition 2 *Let $\mathbf{M} = \langle T, S, A, \underline{P}, \widehat{r}, \widehat{r}_T \rangle$ denote a generalized Markov decision problem. A* policy *π for $\mathbf{M}$ is a mapping from $S \times \{0,1,\ldots,T-1\}$ to A.*

A policy π for $\mathbf{M}$ is a prescription how to act in all possible situations modelled by $\mathbf{M}$. If a decision maker follows π and finds herself/himself in state s at time t, then s/he executes action $\pi(s,t)$.

To evaluate a policy we need to know how much reward is the policy likely to produce over the time horizon. This is the purpose of the expected cumulative reward $\widehat{E}_{\mathbf{M},\pi}(s,t)$ defined iteratively as the lower/upper prevision of a gamble assigning s the current reward plus the expected cumulative reward in the remaining stages.

Definition 3 *Let $\mathbf{M} = \langle T, S, A, \underline{P}, \widehat{r}, \widehat{r}_T \rangle$ denote a generalized Markov decision problem and π a policy for $\mathbf{M}$. The* expected cumulative reward *of $\mathbf{M}$ under π is denoted by $\widehat{E}_{\mathbf{M},\pi}$ and defined as*

$$\left(\widehat{E}_{\mathbf{M},\pi}(s,t)\right)_l = \left(\widehat{r}(s,\pi(s,t),t)\right)_l + \underline{P}\left(\left(\widehat{E}_{\mathbf{M},\pi}(\cdot,t+1)\right)_l, s, \pi(s,t), t\right)$$

and

$$\left(\widehat{E}_{\mathbf{M},\pi}(s,t)\right)_u = \left(\widehat{r}(s,\pi(s,t),t)\right)_u + \overline{P}\left(\left(\widehat{E}_{\mathbf{M},\pi}(\cdot,t+1)\right)_u, s, \pi(s,t), t\right)$$

[2] A lower prevision $\underline{P}$ is called vacuous if it assigns $\inf g$ to any gamble g.

for $s \in S$ and $t \in \{0, 1, \ldots, T-1\}$, and

$$\widehat{E}_{\mathbf{M},\pi}(s, T) = \widehat{r}_T(s)$$

for $s \in S$.

Next we need a way to compare policies. This is not as straightforward in the generalized case as in the classical one because the generalization opens a possibility of incomparability.

Definition 4 *Let $\mathbf{M} = \langle T, S, A, \underline{P}, \widehat{r}, \widehat{r}_T \rangle$ denote a generalized Markov decision problem and π_1 and π_2 two policies for $\mathbf{M}$. We say that π_1 is* preferred *over π_2 at $\langle s, t \rangle$ (for $s \in S$ and $t \in \{0, 1, \ldots, T\}$) if*

$$\left(\widehat{E}_{\mathbf{M},\pi_1}(s, t)\right)_l > \left(\widehat{E}_{\mathbf{M},\pi_2}(s, t)\right)_u.$$

A policy π for $\mathbf{M}$ is called maximal *if there is no policy π' and $s \in S$, $t \in \{0, 1, \ldots, T\}$ such that π' is preferred over π at $\langle s, t \rangle$.*

The above definition of preference and maximality are certainly open to debate. An alternative to the notion of preference as defined above worth further investigation is the Walley's definition of strict preference [9, Sections 3.7–3.9]. I did not adopt it here, because it would require to compute the value of the corresponding lower prevision for the differences of gambles corresponding to all pairs of actions, while currently I need to compute only the lower and upper prevision of gambles corresponding to individual actions. This might be a big computational burden. The advantage of the Walley's definition is that it is stronger so it might help to reduce the frequency of occurrence of indecision. I am also not entirely convinced that the Walley's definition is intuitively more appropriate for the current decision making situation as it looks at the problem in terms of "willingness to exchange one gamble for another" while the question addressed here seems to be "Which gamble is likely to produce more utility?" The modification of the algorithm for Walley's strict preference is otherwise straightforward. Other alternatives I could think of (such as max-min rule, etc.) appear to be inferior.

Requiring dominance over all $\langle s, t \rangle$ pairs or looking at the expected cumulative reward only at the first decision stage seem unsatisfactory as possible alternatives to the above definition of maximality. In the first case, there might be no such policy, in the second case, some intuitively inferior policies could be maximal.

The task now is to find all maximal policies for a given generalized Markov decision problem. It is the topic of the next section.

4 Solving generalized Markov decision problems

This section presents an algorithm for finding all the maximal policies of a given generalized Markov decision problem. It is a generalization of a dynamic programming solution method for finite-horizon Markov decision problems. The main difference from the classical case is the possibility of incomparability of two policies. The procedure is described in Algorithm 1.

The algorithm works by solving the problem backwards. First, it finds all the undominated actions in the last stage. Then it iteratively finds all maximal extensions to the last unsolved decision stage of all partial policies found so far. This is done in two steps. In the first step, adding individual actions to a particular policy is considered. In the second step the extended policies are checked whether they are dominated by an extension of another partial policy and only the undominated are kept into the next iteration. With each policy extension the corresponding expected cumulative reward is also extended. Due to the backward iteration and coupling of computation of expected cumulative reward with policy computation all the information necessary for computation is available at each step.

The description of the algorithm takes advantage of the representation of a mapping as a set of tuples from the Cartesian product of its domain and range. The correctness of the algorithm is proven in the next theorem.

Theorem 1 *Let $\mathbf{M} = \langle T, S, A, \underline{P}, \widehat{r}, \widehat{r}_T \rangle$ denote a generalized Markov decision problem. The set Π returned by Algorithm 1 is the set of pairs of all maximal policies and their corresponding expected cumulative rewards.*

Proof: (Informal) We prove by (bounded) induction on the number of steps remaining to the end of decision horizon T that at time t it holds for any $\left\langle \pi, \widehat{E} \right\rangle \in \Pi$ at the end of the corresponding iteration of the main outer loop of the Algorithm 1 that there is no policy π' for $\mathbf{M}$ such that π' is preferred to π at $\langle s', t' \rangle$ and $\widehat{E}(s', t') = \widehat{E}_{\mathbf{M},\pi}(s', t')$ for $s' \in S$ and $t' \in \{t, t+1, \ldots, T\}$. The theorem then follows.

First assume that $t = T$. Strictly speaking, there is no "corresponding iteration" of the main loop as this case is taken care of before the main loop. There are no policy points and the expected cumulative reward equals just the scrap value at T ($\widehat{r}_T$) by definition, so the statement holds trivially.

Assume that the statement holds at $t + 1$. We want

Algorithm 1 Solve generalized Markov decision problem

Input: $\mathbf{M} = \langle T, S, A, \underline{P}, \widehat{r}, \widehat{r}_T \rangle$ a generalized Markov decision process
Output: $\Pi = \left\{ \left\langle \pi, \widehat{E} \right\rangle \right\}$ set of pairs of all maximal policies for $\mathbf{M}$ and their corresponding expected cummulative rewards

$\widehat{E} = \emptyset$
for $s \in S$ **do**
  $\widehat{E} = \widehat{E} \cup \{\langle s, T, \widehat{r}_T(s) \rangle\}$
end for
$\Pi = \left\{ \left\langle \emptyset, \widehat{E} \right\rangle \right\}$
for $t = T-1, T-2, \ldots, 0$ **do**
  for $s \in S$ **do**
    $\Pi' = \emptyset$
    for $\left\{ \left\langle \pi, \widehat{E} \right\rangle \right\} \in \Pi$ **do**
      for $a \in A$ **do**
        $(\widehat{e}_a)_l =$
        $(\widehat{r}(s,a,t))_l + \underline{P}\left(\left(\widehat{E}(\cdot, t+1) \right)_l, s, a, t \right)$
        $(\widehat{e}_a)_u =$
        $(\widehat{r}(s,a,t))_u + \overline{P}\left(\left(\widehat{E}(\cdot, t+1) \right)_u, s, a, t \right)$
      end for
      for $a \in A$ such that $(\widehat{e}_a)_u \geq (\widehat{e}_b)_l$ for all $b \in A$ **do**
        $\Pi' =$
        $\Pi' \bigcup \left\{ \left\langle \pi \cup \{\langle s, t, a \rangle\}, \widehat{E} \cup \{\langle s, t, \widehat{e}_a \rangle\} \right\rangle \right\}$
      end for
    end for
    $\Pi'' = \emptyset$
    for $\left\{ \left\langle \pi, \widehat{E} \right\rangle \right\} \in \Pi'$ **do**
      $leave = 1$
      for $\left\{ \left\langle \pi', \widehat{E'} \right\rangle \right\} \in \Pi'$ **do**
        if $\left(\widehat{E}(s,t) \right)_u < \left(\widehat{E'}(s,t) \right)_l$ **then**
          $leave = 0$
        end if
      end for
      if $leave \equiv 1$ **then**
        $\Pi'' = \Pi'' \cup \left\{ \left\langle \pi, \widehat{E} \right\rangle \right\}$
      end if
    end for
    $\Pi = \Pi''$
  end for
end for
return Π

to prove that it holds at t. From the assumption it follows that $\widehat{E}(s', t) = \widehat{E}_{\mathbf{M},\pi}(s', t)$ for $s' \in S$ by definition. Assume by contradiction that there is a policy π' for $\mathbf{M}$ such that π' is preferred to π at $\langle s', t' \rangle$ for some $s' \in S$ and $t' \in \{t, t+1, \ldots, T\}$. By the assumption t' cannot be from $\{t+1, t+2, \ldots, T\}$ so $t = t'$. There are two possibilities:

1. $\pi(s,u) = \pi'(s,u)$ for $s \in S$ and $u \in \{t+1, t+2, \ldots, T\}$
2. $\pi(s,u) \neq \pi'(s,u)$ for some $s \in S$ and $u \in \{t+1, t+2, \ldots, T\}$

In the first case π would not be added to Π in the for-loop over all "$a \in A$ such that $(e_a)_u \geq (e_b)_l$ for all $b \in B$" because for $b = \pi(s', t)$ it holds that $(e_a)_u < (e_b)_l$. In the second case, it follows from the monotonicity of coherent lower/upper previsions that either π' is maximal or there is π'' maximal such that π'' is preferred to π at $\langle s', t' \rangle$. This means that the restriction of π'' to $\{t, t+1, \ldots, T-1\}$ must be in Π and hence π would have been eliminated in the for-loop over all "$\left\langle \pi, \widehat{E} \right\rangle \in \Pi$." So the assumption of existence of preferred policy leads to contradiction. ■

5 An example

Let me go through an example to illustrate the algorithm. Assume that $S = \{a, b\}$, $A = \{act1, act2\}$, $T = 2$, $\widehat{r}_2(a) = 1$, $\widehat{r}_2(b) = 0$,

$$\widehat{r}(s, act, t) = \begin{cases} 0.1 & \text{for } s = a \text{ and } t = 1, \\ 0.11 & \text{for } s = a,\ act = act1, \text{ and } t = 0, \\ 0.05 & \text{for } s = b,\ act = act1, \text{ and } t = 0, \\ 0 & \text{otherwise,} \end{cases}$$

and $\underline{P}(\cdot, s, act, t)$ is a lower prevision on $\mathcal{L}(S)$ obtained by the natural extension (Choquet integral in this case) from the following lower probabilities

$$\begin{array}{ll} \langle 0.4, 0.2 \rangle & \text{for } s = a,\ act = act1, \text{ and } t = 0, \\ \langle 0.05, 0.85 \rangle & \text{for } s = a,\ act = act1, \text{ and } t = 1, \\ \langle 0.9, 0 \rangle & \text{for } s = a,\ act = act2, \text{ and } t = 0, \\ \langle 0, 0.9 \rangle & \text{for } s = a,\ act = act2, \text{ and } t = 1, \\ \langle 0.4, 0.5 \rangle & \text{for } s = b,\ act = act1, \text{ and } t = 0, \\ \langle 0.2, 0.45 \rangle & \text{for } s = b,\ act = act1, \text{ and } t = 1, \\ \langle 0.6, 0.3 \rangle & \text{for } s = b,\ act = act2, \text{ and } t = 0, \\ \langle 0.5, 0.4 \rangle & \text{for } s = b,\ act = act2, \text{ and } t = 1, \end{array}$$

where $\langle x, y \rangle$ represents the pair of lower probability value for a and b in the corresponding situation. (This uniquely determines the whole lower probability in this case.) We want to find the solution (i.e., the

	$t=0$		$t=1$	
policy #	$s=a$	$s=b$	$s=a$	$s=b$
1	$act1$, $[0.27, 0.54]$	$act1$, $[0.225, 0.48]$	$act1$, $[0.15, 0.25]$	$act1$, $[0.2, 0.55]$
2	$act1$, $[0.33, 0.57]$	$act1$, $[0.375, 0.51]$	$act1$, $[0.15, 0.25]$	$act2$, $[0.5, 0.6]$
3	$act1$, $[0.23, 0.52]$	$act1$, $[0.2, 0.46]$	$act2$, $[0.1, 0.2]$	$act1$, $[0.2, 0.55]$
4	$act1$, $[0.29, 0.55]$	$act1$, $[0.35, 0.49]$	$act2$, $[0.1, 0.2]$	$act2$, $[0.5, 0.6]$
5	$act1$, $[0.27, 0.54]$	$act2$, $[0.165, 0.37]$	$act1$, $[0.15, 0.25]$	$act1$, $[0.2, 0.55]$
6	$act1$, $[0.33, 0.57]$	$act2$, $[0.255, 0.39]$	$act1$, $[0.15, 0.25]$	$act2$, $[0.5, 0.6]$
7	$act1$, $[0.23, 0.52]$	$act2$, $[0.13, 0.34]$	$act2$, $[0.1, 0.2]$	$act1$, $[0.2, 0.55]$
8	$act1$, $[0.29, 0.55]$	$act2$, $[0.22, 0.36]$	$act2$, $[0.1, 0.2]$	$act2$, $[0.5, 0.6]$
9	$act2$, $[0.15, 0.28]$	$act1$, $[0.225, 0.48]$	$act1$, $[0.15, 0.25]$	$act1$, $[0.2, 0.55]$
10	$act2$, $[0.15, 0.285]$	$act1$, $[0.375, 0.51]$	$act1$, $[0.15, 0.25]$	$act2$, $[0.5, 0.6]$
11	$act2$, $[0.1, 0.235]$	$act1$, $[0.2, 0.46]$	$act2$, $[0.1, 0.2]$	$act1$, $[0.2, 0.55]$
12	$act2$, $[0.1, 0.24]$	$act1$, $[0.35, 0.49]$	$act2$, $[0.1, 0.2]$	$act2$, $[0.5, 0.6]$
13	$act2$, $[0.15, 0.28]$	$act2$, $[0.165, 0.37]$	$act1$, $[0.15, 0.25]$	$act1$, $[0.2, 0.55]$
14	$act2$, $[0.15, 0.285]$	$act2$, $[0.255, 0.39]$	$act1$, $[0.15, 0.25]$	$act2$, $[0.5, 0.6]$
15	$act2$, $[0.1, 0.235]$	$act2$, $[0.13, 0.34]$	$act2$, $[0.1, 0.2]$	$act1$, $[0.2, 0.55]$
16	$act2$, $[0.1, 0.24]$	$act2$, $[0.22, 0.36]$	$act2$, $[0.1, 0.2]$	$act2$, $[0.5, 0.6]$

Table 1: All policies and their corresponding expected cumulative rewards

set of all maximal policies) of generalized Markov decision process $\mathbf{M} = \langle T, S, A, \underline{P}, \widehat{r}, \widehat{r}_T \rangle$. All possible policies with their corresponding expected cumulative rewards are listed in Table 1. The algorithm starts by setting the expected cumulative reward at $t = 2$ equal to $\widehat{r}_2$ (this is omitted from Table 1). Next, the algorithm enters the main loops with $t = 1$, $s = a$, and $act = act1$. The expected cumulative reward for these values is computed as

$$\begin{aligned}
(\widehat{e}_{act1})_l &= (\widehat{r}(a, act1, 1))_l + \underline{P}\left(\left(\widehat{E}(\cdot, 2)\right)_l, a, act1, 1\right) \\
&= 0.1 + \underline{P}((1, 0), a, act1, 1) \\
&= 0.1 + 0.05 \times 1 + 0.95 \times 0 \\
&= 0.15,
\end{aligned}$$

and

$$\begin{aligned}
(\widehat{e}_{act1})_u &= (\widehat{r}(a, act1, 1))_u \\
&\quad + \overline{P}\left(\left(\widehat{E}(\cdot, 2)\right)_u, a, act1, 1\right) \\
&= 0.1 + \overline{P}((1, 0), a, act1, 1) \\
&= 0.1 + 0.15 \times 1 + 0.85 \times 0 \\
&= 0.25.
\end{aligned}$$

Similarly for $act = act2$, we obtain $\widehat{e}_{act2} = [0.1, 0.2]$. As both $(\widehat{e}_{act1})_u \geq (\widehat{e}_{act2})_l$ and $(\widehat{e}_{act2})_u \geq (\widehat{e}_{act1})_l$, Π' now contains two partial policy–cumulative reward pairs: $\langle \{\langle a, 1, act1 \rangle\}, \ldots \rangle$ and $\langle \{\langle a, 1, act2 \rangle\}, \ldots \rangle$. None of these is eliminated in the for-loop over partial policies in Π' and Π is the same as Π'. That concludes the first iteration through the for-loop over states and s is changed to b. Now, $\widehat{e}_{act1} = [0.2, 0.55]$ and $\widehat{e}_{act2} = [0.5, 0.6]$ for both partial policies in Π. Again, neither act is preferred here and we end up with four partial policy–cumulative reward pairs in Π, i.e.,

$$\Pi = \left\{ \begin{array}{c}
\langle \{\langle a, 1, act1 \rangle, \langle b, 1, act1 \rangle\}, \ldots \rangle, \\
\langle \{\langle a, 1, act1 \rangle, \langle b, 1, act2 \rangle\}, \ldots \rangle, \\
\langle \{\langle a, 1, act2 \rangle, \langle b, 1, act1 \rangle\}, \ldots \rangle, \\
\langle \{\langle a, 1, act2 \rangle, \langle b, 1, act2 \rangle\}, \ldots \rangle
\end{array} \right\}.$$

This concludes the first iteration of the main for-loop over t and t is now decremented to 0, $s = a$, and $\pi = \{\langle a, 1, act1 \rangle, \langle b, 1, act1 \rangle\}$. For this partial policy we have

$$\begin{aligned}
\widehat{e}_{act1} &= [0.27, 0.54] \\
\widehat{e}_{act2} &= [0.15, 0.28]. \qquad (+)
\end{aligned}$$

Again, no act is preferred, hence, $\Pi' = \{\langle \{\langle a, 0, act1 \rangle, \langle a, 1, act1 \rangle, \langle b, 1, act1 \rangle\}, \ldots \rangle, \langle \{\langle a, 0, act2 \rangle, \langle a, 1, act1 \rangle, \langle b, 1, act1 \rangle\}, \ldots \rangle\}$. Next π is $\{\langle a, 1, act1 \rangle, \langle b, 1, act2 \rangle\}$. For this partial policy we have

$$\begin{aligned}
\widehat{e}_{act1} &= [0.33, 0.57] \qquad (*) \\
\widehat{e}_{act2} &= [0.15, 0.285].
\end{aligned}$$

As $0.285 < 0.33$ only $\pi \cup \{\langle a, 0, act1 \rangle\}$ is added to Π'. Also for the remaining two partial policies in Π only $\langle a, 0, act1 \rangle$ is undominated extension. Now Π' contains five policies. However, as can be seen from $(+)$ and $(*)$ the partial policy $\{\langle a, 0, act2 \rangle, \langle a, 1, act1 \rangle, \langle b, 1, act1 \rangle\}$ is dominated and will not be added to Π''. (This shows that this check in Algorithm 1 is really necessary.) The reader can verify that the four remaining partial policies are undominated. The final loop with $s = b$ is also left as an exercise to the reader. It is easy to see from Table 1 that the maximal policies for $\mathbf{M}$ are the policies with numbers 1, 2, 3, 4, and 6.

6 Discussion and related work

This paper is a first step toward a generalization of the theory of Markov decision processes to imprecise probabilities. It presents a definition of generalized Markov decision process in terms of interval value functions and lower/upper previsions together with the related notions of expected cumulative reward and maximal policy. A solution method generalizing classical backward induction is also presented. Obviously, much remains to be done. For example, results from the classical theory of Markov decision processes regarding sufficiency of Markov policies (for certain types of models) should be verified or generalized, generalizations of infinite-horizon models and semi-Markov models would be useful, etc. The generalization to imprecise probabilities also creates new problems. Due to possible incomparability of actions/policies we need to keep track of potentially many partial policies during the solution process. This may become computationally intractable. Important question for future research is how to deal with this problem, e.g., what type(s) of models allow us to avoid or minimize this difficulty, how to efficiently store the sets of undominated policies, etc.

Givan et al [4, 2] investigate what they call Bounded Parameter Markov Decision Process. A Bounded Parameter Markov Decision Process is a generalization of a class of Markov Decision Processes with infinite number of decision epochs. The generalization replaces both point transition probabilities and point rewards with closed intervals. Besides the differences in time horizon and representation of transitions, there are two major differences between their approach and approach taken in this paper. First, they interpret their model as representing all classical (precise) models consistent with the bounds of the generalized model and not as a notion of its own. This is similar to the convex sets of probabilities and lower prevision interpretations of imprecise probability models. Second, and more important, difference is the "optimality" criterion used. They consider optimistic and pessimistic "optimal" policies that would be optimal if the imprecision in the model was resolved in the best and worst possible way, respectively. This is essentially a variation of max-min and max-max rules. They do not consider possibility of indecision.

Another related work is the work of Haddawy's group, e.g., [3, 5]. Their motivation is mainly abstraction for efficient solution of planning problems. They consider more general setting for sequential decision making (they do not assume the Markov property), but restrict the uncertainty representation to lower/upper probabilities on a subclass of $\mathcal{P}(S)$ (in my notation). Also, the computation of the expected reward of a particular policy is done using only approximate methods.

The term "Generalized Markov Decision Process" was used earlier [8, 6] for a concept that is unrelated to the use of the term in this paper.

Acknowledgments

I would like to thank Peter Walley and the two anonymous referees for their helpful and stimulating comments on the manuscript of the paper.

The work on this paper has been supported by a Strategic Research Grant No. RP960351 from the National Science and Technology Board and the Ministry of Education, Singapore.

References

[1] G. Choquet. Theory of capacities. *Annales de L'Institut Fourier*, 5:131–295, 1953-54.

[2] T. Dean, R. Givan, and S. Leach. Model reduction techniques for computing approximately optimal solutions for Markov decision processes. In D. Geiger and P. P. Shenoy, editors, *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence (UAI–97)*, pages 124–131, San Francisco, CA, 1997. Morgan Kaufmann Publishers.

[3] A. Doan and P. Haddawy. Sound abstraction of probabilistic actions in the constraint mass assignment framework. In E. Horvitz and F. V. Jensen, editors, *Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence (UAI–96)*, pages 228–235, San Francisco, California, 1996. Morgan-Kaufmann.

[4] R. Givan, S. Leach, and T. Dean. Bounded parameter markov decision processes. Technical Report CS-97-05, Brown University, Providence, Rhode Island, 1997.

[5] V. Ha and P. Haddawy. Theoretical foundations for abstraction-based probabilistic planning. In E. Horvitz and F. V. Jensen, editors, *Proceedings of the Twelfth Annual Conference on Uncertainty*

in Artificial Intelligence (UAI–96), pages 291–298, San Francisco, California, 1996. Morgan-Kaufmann.

[6] M. L. Littman. *Algorithms for Sequential Decision Making*. PhD thesis, Brown University, Providence, Rhode Island, 1996.

[7] M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. Wiley series in probability and mathematical statistics. Applied probability and statistics section. John Wiley & Sons, New York, 1994.

[8] C. Szepesvári and M. L. Littman. Generalized markov decision processes: Dynamic-programming and reinforcement-learning algorithms. Technical Report CS-96-11, Brown University, Providence, Rhode Island, 1996.

[9] P. Walley. *Statistical Reasoning With Imprecise Probabilities*. Chapman and Hall, New York, 1991.

[10] D. J. White. *Markov decision processes*. John Wiley & Sons, Chichester, England, 1993.