Markov Decision Processes with Multiple Objectives*

Krishnendu Chatterjee[1], Rupak Majumdar[2], and Thomas A. Henzinger[1,3]

[1]UC Berkeley [2]UC Los Angeles [3]EPFL
{c_krish,tah}@eecs.berkeley.edu rupak@cs.ucla.edu

Abstract. We consider Markov decision processes (MDPs) with multiple discounted reward objectives. Such MDPs occur in design problems where one wishes to simultaneously optimize several criteria, for example, latency and power. The possible trade-offs between the different objectives are characterized by the Pareto curve. We show that every Pareto-optimal point can be achieved by a memoryless strategy; however, unlike in the single-objective case, the memoryless strategy may require randomization. Moreover, we show that the Pareto curve can be approximated in polynomial time in the size of the MDP. Additionally, we study the problem if a given value vector is realizable by any strategy, and show that it can be decided in polynomial time; but the question whether it is realizable by a deterministic memoryless strategy is NP-complete. These results provide efficient algorithms for design exploration in MDP models with multiple objectives.

1 Introduction

Markov decision processes (MDPs) are a widely studied model for dynamic and stochastic systems [2, 8]. An MDP models a dynamic system that evolves through stages. In each stage, a controller chooses one of several actions, and the system stochastically evolves to a new state based on the current state and the chosen action. In addition, one associates a cost or reward with each state and transition, and the central question is to find a strategy of choosing the actions that optimizes the rewards obtained over the run of the system, where the rewards are combined using a discounted sum. In many modeling domains, however, there is no unique objective to be optimized, but multiple, potentially dependent and conflicting objectives. For example, in designing a computer system, one is interested not only in maximizing performance but also in minimizing power. Similarly, in an inventory management system, one wishes to optimize several potentially dependent costs for maintaining each kind of product, and in AI planning, one wishes to find a plan that optimizes several distinct goals. The usual MDP model is insufficient to express these natural problems.

* This research was supported in part by the AFOSR MURI grant F49620-00-1-0327, and the NSF grants CCR-0225610, CCR-0234690, and CCR-0427202.

We study MDPs with multiple objectives, an extension of the MDP model where there are several reward functions [4, 10]. In MDPs with multiple objectives, we are interested not in a single solution that is simultaneously optimal in all objectives (which may not exist), but in a notion of "trade-offs" called the *Pareto curve*. Informally, the Pareto curve consists of the set of realizable value profiles (or dually, the strategies that realize them) which are not dominated (in every dimension) by any other value profile. Pareto optimality has been studied in co-operative game theory [6], and in multi-criterion optimization and decision making in both economics and engineering [5, 9, 11]. Finding *some* Pareto-optimal point can be reduced to optimizing a single objective: optimize a convex combination of objectives using a set of positive weights; the optimal strategy must be Pareto-optimal as well (the "weighted factor method") [4]. In design space exploration, however, we want to find not one, but *all* Pareto-optimal points in order to better understand the trade-offs in the design. Unfortunately, even with just two reward functions, the Pareto curve may have infinitely many points, and also contain irrational payoff values. Thus, previous work has focused on constructing a sampling of the Pareto curve, either by choosing a variety of weights in the weighted factor method, or by imposing a lexicographic ordering on the objectives and sequentially optimizing each objective according to the order [1, 2]. Unfortunately, this does not provide any guarantee about the quality of the solutions.

Instead, we study the *approximate* version of the problem: the ϵ-approximate Pareto curve [7] for MDPs with multiple discounted reward criteria. Informally, the ϵ-approximate Pareto curve for $\epsilon > 0$ contains a set of strategies (or dually, their payoff values) such that there is no other strategy whose value dominates the values in the Pareto curve by a factor of $1 + \epsilon$. Surprisingly, a polynomial-sized ϵ-approximate Pareto curve always exists. Moreover, we show that such an approximate Pareto curve may be computed efficiently (in polynomial time) in the size of the MDP. Our proof is based on the following characterization of Pareto-optimal points: every Pareto-optimal value profile can be realized by a memoryless (but possibly randomized) strategy. This enables the reduction of the problem to multi-objective linear programs, and we can apply the methods of [7].

We also study the *Pareto realizability* decision problem: given a profile of values, is there a Pareto-optimal strategy that dominates it? We show that the Pareto realizability problem can be solved in polynomial time. However, if we restrict the set of strategies to be pure (i.e., no randomization), then the problem becomes NP-hard. Our complexities are comparable to the single discounted reward case, where linear programming provides a polynomial-time solution [8]. However, unlike in the single-reward case, where pure and memoryless optimal strategies always exist, here, checking pure and memoryless realizability is hard.

The results of this paper provide polynomial-time algorithms for both the decision problem and the optimization problem for MDPs with multiple discounted reward objectives. Since the Pareto curve forms a useful "user interface" for desirable solutions, we believe that these results will lead to efficient design space exploration algorithms in multi-criterion design.

The rest of the paper is organized as follows. In Section 2, we give the basic definitions, and show in Section 3 the sufficiency of memoryless strategies. Section 4 gives a polynomial-time algorithm to construct the ϵ-approximate Pareto curve. Section 5 studies the decision version of the problem. Finally, in Section 6, we discuss the extension to MDPs with limit-average (not discounted) reward objectives, and mention some open problems.

2 Discounted Reward Markov Decision Processes

We denote the set of probability distributions on a set U by $\mathcal{D}(U)$.

Markov decision processes (MDPs). A *Markov decision process* (MDP) $G = (S, A, \delta)$ consists of a finite, non-empty set S of states, a finite, non-empty set A of actions, and a probabilistic transition function $\delta : S \times A \to \mathcal{D}(S)$ that, given a state $s \in S$ and an action $a \in A$, gives the probability $\delta(s,a)(t)$ of the next state t. We denote by $\mathrm{Dest}(s,a) = \mathrm{Support}(\delta(s,a))$ the set of possible successors of s when the action a is chosen. Given an MDP G, we define the set of edges by $E = \{ (s,t) \mid \exists a \in A.\ t \in \mathrm{Dest}(s,a) \}$, and write $E(s) = \{ t \mid (s,t) \in E \}$ for the set of possible successors of s.

Plays and strategies. A *play* of G is an infinite sequence $\langle s_0, s_1, \ldots \rangle$ of states such that for all $i \geq 0$, we have $(s_i, s_{i+1}) \in E$. A strategy σ is a recipe that specifies how to extend a play. Formally, a *strategy* σ is a function $\sigma\colon S^+ \to \mathcal{D}(A)$ that, given a finite and non-empty sequence of states representing the history of the play so far, chooses a probability distribution over the set A of actions. In general, a strategy depends on the history and uses randomization. A strategy that depends only on the current state is a *memoryless* or *stationary* strategy, and can be represented as a function $\sigma\colon S \to \mathcal{D}(A)$. A strategy that does not use randomization is a *deterministic* or *pure* strategy, i.e., for all histories $\langle s_0, s_1, \ldots, s_k \rangle$ there exists $a \in A$ such that $\sigma(\langle s_0, s_1, \ldots, s_k \rangle)(a) = 1$. A *pure memoryless* strategy is both pure and memoryless, and can be represented as a function $\sigma\colon S \to A$. We denote by Σ, Σ^M, Σ^P, and Σ^{PM} the sets of all strategies, all memoryless strategies, all pure strategies, and all pure memoryless strategies, respectively.

Outcomes. For a strategy σ and an initial state s, we denote by $\mathrm{Outcome}(s,\sigma)$ the set of possible plays that start from s given strategy σ, that is, $\mathrm{Outcome}(s,\sigma) = \{ \langle s_0, s_1, \ldots \rangle \mid \forall k \geq 0.\, \exists a_k \in A.\, \sigma(\langle s_0, s_1, \ldots, s_k \rangle)(a_k) > 0 \text{ and } s_{k+1} \in \mathrm{Dest}(s_k, a_k) \}$. Once the initial state and a strategy is chosen, the MDP is reduced to a stochastic process. We denote by X_i and θ_i random variables for the i-th state and the i-th action, respectively, in this stochastic process. An event is a measurable subset of $\mathrm{Outcome}(s,\sigma)$, and the probabilities of events are uniquely defined. Given a strategy σ, an initial state s, and an event Φ, we denote by $\Pr_s^\sigma(\Phi)$ the probability that a play belongs to Φ when the MDP starts in state s and the strategy σ is used. For a measurable function f

that maps plays to reals, we write $\mathbb{E}_s^\sigma[f]$ for the expected value of f when the MDP starts in state s and the strategy σ is used.

Rewards and objectives. Let $r\colon S \times A \to \mathbb{R}$ be a *reward function* that associates with every state and action a real-valued reward. For a reward function r the discounted reward objective is to maximize the discounted sum of rewards, which is defined as follows. Given a discount factor $0 \leq \beta < 1$, the *discounted reward* or *payoff value* for a strategy σ and an initial state s with respect to the reward function r is $\mathrm{Val}_{dis}^\sigma(r, s, \beta) = \sum_{t=0}^{\infty} \beta^t \cdot \mathbb{E}_s^\sigma[r(X_t, \theta_t)]$.

We consider MDPs with k different reward functions $r_1, \ldots, r_k$. Given an initial state s, a strategy σ, and a discount factor $0 \leq \beta < 1$, the discounted reward value vector, or *payoff profile*, at s for σ with respect to $\boldsymbol{r} = \langle r_1, \ldots, r_k \rangle$ is defined as $\mathrm{Val}_{dis}^\sigma(\boldsymbol{r}, s, \beta) = \langle \mathrm{Val}_{dis}^\sigma(r_1, s, \beta), \ldots, \mathrm{Val}_{dis}^\sigma(r_k, s, \beta) \rangle$.

Comparison operators on vectors are interpreted in a point-wise fashion, i.e., given two real-valued vectors $\boldsymbol{v}_1 = \langle v_1^1, \ldots, v_1^k \rangle$ and $\boldsymbol{v}_2 = \langle v_2^1, \ldots, v_2^k \rangle$, and $\bowtie \in \{ <, \leq, = \}$, we write $\boldsymbol{v}_1 \bowtie \boldsymbol{v}_2$ if and only if for all $1 \leq i \leq k$, we have $v_1^i \bowtie v_2^i$. We write $\boldsymbol{v}_1 \neq \boldsymbol{v}_2$ to denote that vector $\boldsymbol{v}_1$ is not equal to $\boldsymbol{v}_2$, that is, it is not the case that $\boldsymbol{v}_1 = \boldsymbol{v}_2$.

Pareto-optimal strategies. Given an MDP G and reward functions $r_1, \ldots, r_k$, a strategy σ is a *Pareto-optimal* strategy [6] from a state s if there is no strategy $\sigma' \in \Sigma$ such that both $\mathrm{Val}_{dis}^\sigma(\boldsymbol{r}, s, \beta) \leq \mathrm{Val}_{dis}^{\sigma'}(\boldsymbol{r}, s, \beta)$ and $\mathrm{Val}_{dis}^\sigma(\boldsymbol{r}, s, \beta) \neq \mathrm{Val}_{dis}^{\sigma'}(\boldsymbol{r}, s, \beta)$; that is, there is no strategy σ' such that for all $1 \leq j \leq k$, we have $\mathrm{Val}_{dis}^\sigma(r_j, s, \beta) \leq \mathrm{Val}_{dis}^{\sigma'}(r_j, s, \beta)$, and there exists $1 \leq j \leq k$ with $\mathrm{Val}_{dis}^\sigma(r_j, s, \beta) < \mathrm{Val}_{dis}^{\sigma'}(r_j, s, \beta)$. For a Pareto-optimal strategy σ, the corresponding payoff profile $\mathrm{Val}_{dis}^\sigma(\boldsymbol{r}, s, \beta)$ is referred to as a *Pareto-optimal point*. In case $k = 1$, the class of Pareto-optimal strategies are called *optimal* strategies.

Sufficiency of strategies. Given reward functions $r_1, \ldots, r_k$, a family $\Sigma^{\mathcal{C}}$ of strategies *suffices* for Pareto optimality for discounted reward objectives if for every discount factor β, state s, and Pareto-optimal strategy $\sigma \in \Sigma$, there is a strategy $\sigma' \in \Sigma^{\mathcal{C}}$ such that $\mathrm{Val}_{dis}^\sigma(\boldsymbol{r}, s, \beta) \leq \mathrm{Val}_{dis}^{\sigma'}(\boldsymbol{r}, s, \beta)$.

Theorem 1. [2] *In MDPs with a single reward function r, the pure memoryless strategies suffice for optimality for the discounted reward objective, i.e., for all discount factors $0 \leq \beta < 1$ and states $s \in S$, there exists a pure memoryless strategy $\sigma^* \in \Sigma^{PM}$ such that for all strategies $\sigma \in \Sigma$, we have $\mathrm{Val}_{dis}^\sigma(r, s, \beta) \leq \mathrm{Val}_{dis}^{\sigma^*}(r, s, \beta)$.*

3 Memoryless Strategies Suffice for Pareto Optimality

In the sequel, we fix a discount factor β such that $0 \leq \beta < 1$. Proposition 1 shows the existence of pure memoryless Pareto-optimal strategies.

Proposition 1. *There exist pure memoryless Pareto-optimal strategies for MDPs with multiple discounted reward objectives.*

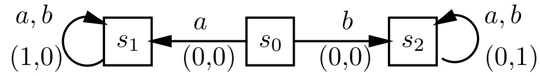

Fig. 1. MDP for Example 1

Proof. Given reward functions $r_1, \ldots, r_k$, consider a reward function $r_+ = r_1 + \cdots + r_k$, that is, for all $s \in S$, we have $r_+(s) = r_1(s) + \cdots + r_k(s)$. Let $\sigma^* \in \Sigma^{PM}$ be a pure memoryless optimal strategy for the reward function r_+ with the discounted reward objective with discount β (such a strategy exists by Theorem 1). We show that σ^* is Pareto-optimal. Assume towards contradiction that σ^* is not a Pareto-optimal strategy, then let $\sigma \in \Sigma$ be such that $\mathrm{Val}^{\sigma^*}_{dis}(\boldsymbol{r}, s, \beta) \leq \mathrm{Val}^{\sigma}_{dis}(\boldsymbol{r}, s, \beta)$, and for some j, $\mathrm{Val}^{\sigma^*}_{dis}(r_j, s, \beta) < \mathrm{Val}^{\sigma}_{dis}(r_j, s, \beta)$. Then we have $\mathrm{Val}^{\sigma^*}_{dis}(r_+, s, \beta) = \sum_{j=1}^{k} \mathrm{Val}^{\sigma^*}_{dis}(r_j, s, \beta) < \sum_{j=1}^{k} \mathrm{Val}^{\sigma}_{dis}(r_j, s, \beta) = \mathrm{Val}^{\sigma}_{dis}(r_+, s, \beta)$. This contradicts that σ^* is optimal for r_+. ∎

The above proof can be generalized to any convex combination of the multiple objectives, that is, for positive weights $w_1, \ldots, w_k$, the optimal strategy for the single objective $\sum_i w_i \cdot r_i$ is Pareto-optimal. This technique is called the *weighted factor method* [4, 10], and used commonly in engineering practice to find subsets of the Pareto set [5]. However, not all Pareto-optimal points are obtained in this fashion, as the following example shows.

Example 1. Consider the MDP from Fig. 1, with two actions a and b, and two reward functions r_1 and r_2. The transitions and the respective rewards are shown as labeled edges in the figure. Consider the discounted reward objectives for reward functions r_1 and r_2. For the pure memoryless strategies (and also the pure strategies) in this MDP, the possible value vectors are $(\frac{\beta}{1-\beta}, 0)$ and $(0, \frac{\beta}{1-\beta})$. However, consider a memoryless strategy σ_m that at state s_0 chooses action a and b each with probability $1/2$. For $\boldsymbol{r} = (r_1, r_2)$, we have $\mathrm{Val}^{\sigma_m}_{dis}(\boldsymbol{r}, s_0, \beta) = (\frac{\beta}{2\cdot(1-\beta)}, \frac{\beta}{2\cdot(1-\beta)})$. The strategy σ_m is Pareto-optimal and no pure memoryless strategy can achieve the corresponding value vector. Hence it follows that the pure strategies (and the pure memoryless strategies) do not suffice for Pareto optimality. Note that for all $0 < x < 1$, the memoryless strategy that chooses a with probability x, is a Pareto-optimal strategy, with value vector $(\frac{x\cdot\beta}{1-\beta}, \frac{(1-x)\cdot\beta}{1-\beta})$. Hence the set of Pareto-optimal value vectors may be uncountable and value vectors may have irrational values. ∎

We now show that the family of memoryless strategies suffices for Pareto optimality. We assume the state space S is enumerated as $S = \{ 1, \ldots, n \}$. For a state $t \in S$, we define the reward function r_t by $r_t(s, a) = 1$ if $s = t$, and 0 otherwise, i.e., a reward of value 1 is gained whenever state t is visited. Similarly, we define the reward function $r_{t,b}$ for a state $t \in S$ and an action $b \in A$ by $r_{t,b}(s, a) = 1$ if $s = t$ and $b = a$, and 0 otherwise, i.e., a reward of value 1 is gained whenever state t is visited and action b is chosen. Given a strategy $\sigma \in \Sigma$ and a state $s \in S$, we define the discounted frequency of the state-action pair

(t, a) as

$$\mathrm{Freq}_s^\sigma(t, a, \beta) = \mathrm{Val}_{dis}^\sigma(r_{t,a}, s, \beta) = \sum_{k=0}^{\infty} \beta^k \cdot \mathbb{E}_s^\sigma[\mathbf{1}_{(X_k=t, \theta_k=a)}],$$

and the discounted frequency of the state t as

$$\mathrm{Freq}_s^\sigma(t, \beta) = \mathrm{Val}_{dis}^\sigma(r_t, s, \beta) = \sum_{k=0}^{\infty} \beta^k \cdot \mathbb{E}_s^\sigma[\mathbf{1}_{(X_k=t)}].$$

Observe that $\sum_{a\in A} \mathrm{Freq}_s^\sigma(t, a, \beta) = \mathrm{Freq}_s^\sigma(t, \beta)$ for all $s, t \in S$ and $\sigma \in \Sigma$. For a memoryless strategy $\sigma_m \in \Sigma^M$ and a transition function δ, we denote by $\delta_{\sigma_m}(s, t) = \sum_{a\in A} \sigma_m(s)(a) \cdot \delta(s, a)(t)$ the probability of the transition from s to t given δ and σ_m.

Proposition 2. *Given a memoryless strategy $\sigma_m \in \Sigma^M$, consider a vector $\boldsymbol{z} = \langle z_1, \ldots, z_n\rangle$ of variables, where $S = \{ 1, \ldots, n \}$. The set of n equations*

$$z_i = r_s(i) + \beta \cdot \sum_{j\in S} \delta_{\sigma_m}(j, i) \cdot z_j, \quad \textit{for } i \in S,$$

has the unique solution $z_i = \mathrm{Freq}_s^{\sigma_m}(i, \beta)$ for all $1 \le i \le n$.

Proof. To establish the desired claim we show that for all $i \in S$, we have $\mathrm{Freq}_s^{\sigma_m}(i, \beta) = r_s(i) + \beta \cdot \sum_{j\in S} \delta_{\sigma_m}(j, i) \cdot \mathrm{Freq}_s^{\sigma_m}(j, \beta)$. The uniqueness follows from arguments similar to the uniqueness of values under memoryless strategies (see [2]).

$$\begin{aligned}
\mathrm{Freq}_s^{\sigma_m}(i, \beta) &= \textstyle\sum_{k=0}^{\infty} \beta^k \cdot \mathbb{E}_s^{\sigma_m}[\mathbf{1}_{(X_k=i)}] \\
&= \textstyle\sum_{k=0}^{\infty} \beta^k \cdot \mathbb{E}_s^{\sigma_m}[\sum_{j\in S} \mathbf{1}_{(X_{k-1}=j)}\ \delta_{\sigma_m}(j, i)] \\
&= r_s(i) + \textstyle\sum_{k=1}^{\infty} \beta^k \cdot \sum_{j\in S} \mathbb{E}_s^{\sigma_m}[\mathbf{1}_{(X_{k-1}=j)}] \cdot \delta_{\sigma_m}(j, i) \\
&= r_s(i) + \beta \cdot \textstyle\sum_{j\in S} \left(\sum_{k=1}^{\infty} \beta^{k-1} \mathbb{E}_s^{\sigma_m}[\mathbf{1}_{(X_{k-1}=j)}]\right) \cdot \delta_{\sigma_m}(j, i) \\
&= r_s(i) + \beta \cdot \textstyle\sum_{j\in S} \left(\sum_{z=0}^{\infty} \beta^{z} \cdot \mathbb{E}_s^{\sigma_m}[\mathbf{1}_{(X_{z}=j)}]\right) \cdot \delta_{\sigma_m}(j, i) \\
&= r_s(i) + \beta \cdot \textstyle\sum_{j\in S} \mathrm{Freq}_s^{\sigma_m}(j, \beta) \cdot \delta_{\sigma_m}(j, i). \ \blacksquare
\end{aligned}$$

Given a strategy $\sigma \in \Sigma$ and an initial state s, we define a memoryless strategy $\sigma_{f,s} \in \Sigma^M$ from the discounted frequency of the strategy σ as follows:

$$\sigma_{f,s}(t)(a) = \frac{\mathrm{Freq}_s^\sigma(t, a, \beta)}{\mathrm{Freq}_s^\sigma(t, \beta)}, \quad \text{for all } t \in S \text{ and } a \in A.$$

Since $\sum_{a\in A} \mathrm{Freq}_s^\sigma(t, a, \beta) = \mathrm{Freq}_s^\sigma(t, \beta)$ and $\mathrm{Freq}_s^\sigma(t, a, \beta) \ge 0$, it follows that $\sigma_{f,s}(t)$ is a probability distribution. Thus $\sigma_{f,s}$ is a memoryless strategy. From Proposition 2, and the identity $\mathrm{Freq}_s^\sigma(i, \beta) = r_s(i) + \beta \cdot \sum_{j\in S} \mathrm{Freq}_s^\sigma(j, \beta) \cdot \delta_{\sigma_{f,s}}(j)$, we obtain the following lemma.

Lemma 1. *For all strategies $\sigma \in \Sigma$ and states $i, s \in S$, we have $\mathrm{Freq}_s^\sigma(i, \beta) = \mathrm{Freq}_s^{\sigma_{f,s}}(i, \beta)$.*

Proof. We show that $\mathrm{Freq}_s^\sigma(i,\beta) = r_s(i) + \beta \cdot \sum_{j\in S} \mathrm{Freq}_s^\sigma(j,\beta)\cdot \delta_{\sigma_{f,s}}(j)$. The result then follows from Proposition 2.

$$\begin{aligned}
\mathrm{Freq}_s^\sigma(i,\beta) &= \textstyle\sum_{k=0}^{\infty} \beta^k \cdot \mathbb{E}_s^\sigma[\mathbf{1}_{(X_k=i)}] \\
&= \textstyle\sum_{k=0}^{\infty} \beta^k \cdot \mathbb{E}_s^\sigma[\sum_{j\in S}\sum_{a\in A} \mathbf{1}_{(X_{k-1}=j,\theta_{k-1}=a)}\delta(j,a)(i)] \\
&= r_s(i) + \textstyle\sum_{k=1}^{\infty} \beta^k \cdot \sum_{j\in S}\sum_{a\in A} \mathbb{E}_s^\sigma[\mathbf{1}_{(X_{k-1}=j,\theta_{k-1}=a)}]\cdot\delta(j,a)(i) \\
&= r_s(i) + \beta\cdot \textstyle\sum_{j\in S}\sum_{a\in A}(\sum_{k=1}^{\infty}\beta^{k-1}\mathbb{E}_s^\sigma[\mathbf{1}_{(X_{k-1}=j,\theta_{k-1}=a)}])\cdot\delta(j,a)(i) \\
&= r_s(i) + \beta\cdot \textstyle\sum_{j\in S}\sum_{a\in A}\mathrm{Freq}_s^\sigma(j,a,\beta)\cdot\delta(j,a)(i) \\
&= r_s(i) + \beta\cdot \textstyle\sum_{j\in S}\sum_{a\in A}\big(\mathrm{Freq}_s^\sigma(j,\beta)\cdot\frac{\mathrm{Freq}_s^\sigma(j,a,\beta)}{\mathrm{Freq}_s^\sigma(j,\beta)}\big)\cdot\delta(j,a)(i) \\
&= r_s(i) + \beta\cdot \textstyle\sum_{j\in S}\mathrm{Freq}_s^\sigma(j,\beta)\cdot\big(\sum_{a\in A}\frac{\mathrm{Freq}_s^\sigma(j,a,\beta)}{\mathrm{Freq}_s^\sigma(j,\beta)}\big)\cdot\delta(j,a)(i) \\
&= r_s(i) + \beta\cdot \textstyle\sum_{j\in S}\mathrm{Freq}_s^\sigma(j,\beta)\cdot\sum_{a\in A}\sigma_{f,s}(j)(a)\cdot\delta(j,a)(i) \\
&= r_s(i) + \beta\cdot \textstyle\sum_{j\in S}\mathrm{Freq}_s^\sigma(j,\beta)\cdot\delta_{\sigma_{f,s}}(j,i). \blacksquare
\end{aligned}$$

Corollary 1. *For all strategies $\sigma \in \Sigma$, all states $i,s \in S$, and all actions $a \in A$, we have* $\mathrm{Freq}_s^\sigma(i,a,\beta) = \mathrm{Freq}_s^{\sigma_{f,s}}(i,a,\beta)$.

Proof. The following equalities follow from the definitions and Lemma 1:

$$\begin{aligned}
\mathrm{Freq}_s^{\sigma_{f,s}}(i,a,\beta) &= \mathrm{Freq}_s^{\sigma_{f,s}}(i,\beta)\cdot\sigma_{f,s}(i)(a) \\
&= \mathrm{Freq}_s^{\sigma}(i,\beta)\cdot\tfrac{\mathrm{Freq}_s^\sigma(i,a,\beta)}{\mathrm{Freq}_s^\sigma(i,\beta)} \\
&= \mathrm{Freq}_s^\sigma(i,a,\beta). \blacksquare
\end{aligned}$$

Theorem 2. *For all reward functions r, all strategies $\sigma \in \Sigma$, and all states $s \in S$, we have $\mathrm{Val}_{dis}^\sigma(r,s,\beta) = \mathrm{Val}_{dis}^{\sigma_{f,s}}(r,s,\beta)$.*

Proof. The result is proved as follows:

$$\begin{aligned}
\mathrm{Val}_{dis}^\sigma(r,s,\beta) &= \textstyle\sum_{k=0}^{\infty}\beta^k\cdot\mathbb{E}_s^\sigma[r(X_k,\theta_k)] \\
&= \textstyle\sum_{k=0}^{\infty}\beta^k\cdot\mathbb{E}_s^\sigma[\sum_{i\in S}\sum_{a\in A} r(i,a)\cdot\mathbf{1}_{(X_k=i,\theta_k=a)}] \\
&= \textstyle\sum_{i\in S}\sum_{a\in A}(\sum_{k=0}^{\infty}\beta^k\cdot\mathbb{E}_s^\sigma[\mathbf{1}_{(X_k=i,\theta_k=a)}])\cdot r(i,a) \\
&= \textstyle\sum_{i\in S}\sum_{a\in A}\mathrm{Freq}_s^\sigma(i,a,\beta)\cdot r(i,a) \\
&= \textstyle\sum_{i\in S}\sum_{a\in A}\mathrm{Freq}_s^{\sigma_{f,s}}(i,a,\beta)\cdot r(i,a) \quad \text{(by Corollary 1)}.
\end{aligned}$$

Similarly, it follows that $\mathrm{Val}_{dis}^{\sigma_{f,s}}(r,s,\beta) = \sum_{i\in S}\sum_{a\in A}\mathrm{Freq}_s^{\sigma_{f,s}}(i,a,\beta)\cdot r(i,a)$. This establishes the result. $\blacksquare$

Theorem 2 yields Theorem 3, and since the set of memoryless strategies is convex, it also shows that the set of Pareto-optimal points is convex.

Theorem 3. *Given an MDP with multiple reward functions $\boldsymbol{r}$, for all strategies $\sigma \in \Sigma$ and all states $s \in S$, the memoryless strategy $\sigma_{f,s} \in \Sigma^M$ satisfies $\mathrm{Val}_{dis}^\sigma(\boldsymbol{r},s,\beta) = \mathrm{Val}_{dis}^{\sigma_{f,s}}(\boldsymbol{r},s,\beta)$. Consequently, the memoryless strategies suffice for Pareto optimality for MDPs with multiple discounted reward objectives.*

4 Approximating the Pareto Curve

Pareto curve. Let M be an MDP with k reward functions $\boldsymbol{r} = \langle r_1,\ldots,r_k\rangle$. The *Pareto curve* $P_{dis}(M,s,\beta,\boldsymbol{r})$ of the MDP M at state s with respect to discounted reward objectives is the set of all k-vectors of values such that for each $\boldsymbol{v} \in P_{dis}(M,s,\beta,\boldsymbol{r})$, there is a Pareto-optimal strategy σ such that $\mathrm{Val}^{\sigma}_{dis}(\boldsymbol{r},s,\beta) = \boldsymbol{v}$. We are interested not only in the values, but also in the Pareto-optimal strategies. We often blur the distinction and refer to the Pareto curve $P_{dis}(M,s,\beta,\boldsymbol{r})$ as a set of strategies that achieve the Pareto-optimal values (if there is more than one strategy that achieves the same value vector, then $P_{dis}(M,s,\beta,\boldsymbol{r})$ contains at least one of them). For an MDP M and a real $\epsilon > 0$, an *ϵ-approximate Pareto curve*, denoted $P^{\epsilon}_{dis}(M,s,\beta,\boldsymbol{r})$, is a set of strategies in Σ such that there is no strategy $\sigma' \in \Sigma$ such that for all strategies $\sigma \in P^{\epsilon}_{dis}(M,s,\beta,\boldsymbol{r})$, we have $\mathrm{Val}^{\sigma'}_{dis}(r_i,s,\beta) \geq (1+\epsilon)\cdot \mathrm{Val}^{\sigma}_{dis}(r_i,s,\beta)$ for all $1 \leq i \leq k$. That is, an ϵ-approximate Pareto curve contains enough strategies such that every Pareto-optimal strategy is "almost" dominated by some strategy in $P^{\epsilon}_{dis}(M,s,\beta,\boldsymbol{r})$.

Multi-objective linear programming. A *multi-objective linear program L* consists of (i) a set of k objective functions $o_1,\ldots,o_k$, where $o_i(\boldsymbol{x}) = \boldsymbol{c}_i^T \cdot \boldsymbol{x}$, for a vector $\boldsymbol{c}_i$ of coefficients and a vector $\boldsymbol{x}$ of variables; and (ii) a set of linear constraints specified by $A\cdot\boldsymbol{x} \geq \boldsymbol{b}$, for a matrix A and a value vector $\boldsymbol{b}$. A valuation of $\boldsymbol{x}$ is a *solution* if it satisfies the set (ii) of linear constraints. A solution $\boldsymbol{x}$ is *Pareto-optimal* if there is no other solution $\boldsymbol{x}'$ such that both $\langle o_1(\boldsymbol{x}),\ldots,o_k(\boldsymbol{x})\rangle \leq \langle o_1(\boldsymbol{x}'),\ldots,o_k(\boldsymbol{x}')\rangle$ and $\langle o_1(\boldsymbol{x}),\ldots,o_k(\boldsymbol{x})\rangle \neq \langle o_1(\boldsymbol{x}'),\ldots,o_k(\boldsymbol{x}')\rangle$. Given a multi-objective linear program L, the *Pareto curve* for L, denoted $P(L)$, is the set of k-vectors $\boldsymbol{v}$ of values such that there is a Pareto-optimal solution $\boldsymbol{x}$ of L with $\boldsymbol{v} = \langle o_1(\boldsymbol{x}),\ldots,o_k(\boldsymbol{x})\rangle$. The definition of *$\epsilon$-approximate Pareto curves* $P^{\epsilon}(L)$ for a multi-objective linear program L and a real $\epsilon > 0$, is analogous to the definition of ϵ-approximate Pareto curves for multi-objective MDPs given above.

Theorem 4. [7] *Given a multi-objective linear program L with k objective functions, the following assertions hold:*

1. *For all $\epsilon > 0$, there exists an ϵ-approximate Pareto curve $P^{\epsilon}(L)$ whose size is is polynomial in $|L|$ and $\frac{1}{\epsilon}$, and exponential in k.*
2. *For all $\epsilon > 0$, there exists an algorithm to construct an ϵ-approximate Pareto curve $P^{\epsilon}(L)$ in time polynomial in $|L|$ and $\frac{1}{\epsilon}$, and exponential in k.*

Proof. Part 1 is a direct consequence of Theorem 1 of [7]. Part 2 follows from Theorem 3 of [7] and the fact that linear programming can be solved in polynomial time. ∎

Solving MDPs by linear programming. Given an MDP $M = (S,A,\delta)$ with state space $S = \{1,\ldots,n\}$, a reward function r, and a discount factor $0 \leq \beta < 1$, the discounted reward objective can be computed as the optimal solution of a linear program [2]. For multi-objective MDPs, we extend the standard linear programming formulation as follows. Given MDP M an discount factor β as before, an initial state s, and reward functions $r_1,\ldots,r_k$, the multi-objective linear program has the set $\{\, x(t,a) \mid t \in S \text{ and } a \in A \,\}$ of variables. Intuitvely, the variable $x(t,a)$ represents the discounted frequency of the state-action pair

(t,a) when the starting state is s. The constraints of the multi-objective linear program over the variables $x(\cdot,\cdot)$ are given by:

$$\begin{aligned} \textstyle\sum_{a\in A} x(t,a) = r_s(t) + \beta \cdot \sum_{u\in S}\sum_{a_1\in A} \delta(u,a_1)(t)\cdot x(u,a_1), &\quad \text{for } t \in S; \\ x(t,a) \geq 0, &\quad \text{for } t\in S,\ a\in A. \end{aligned} \tag{1}$$

Equation (1) provides constraints on the discounted frequencies. The k objective functions are

$$\max \sum_{t\in S}\sum_{a\in A} r_i(t,a)\cdot x(t,a), \quad \text{for } i \in \{\,1,\ldots,k\,\}.$$

Consider any solution $x(t,a)$, for $t\in S$ and $a\in A$, of this linear program. Let $x(t) = \sum_{a\in A} x(t,a)$. The solution derives a memoryless strategy that chooses action a at state t with probability $\frac{x(t,a)}{x(t)}$. The linear program with the i-th objective function asks to maximize the discounted reward for the i-th reward function r_i over the set of all memoryless strategies. The optimal solution for the linear program with only the i-th objective also derives an optimal memoryless strategy for the reward function r_i. Furthermore, given a solution of the linear program, or equivalently, the memoryless strategy derived from the solution, we can compute the corresponding payoff profile in polynomial time, because the MDP reduces to a Markov chain when the strategy is fixed.

We denote by $L_{dis}(M,s,\beta,\boldsymbol{r})$ the multi-objective linear program defined above for the memoryless strategies of an MDP M, state s of M, discount factor β, and reward functions $\boldsymbol{r} = \langle r_1,\ldots,r_k\rangle$. Let $P(L_{dis}(M,s,\beta,\boldsymbol{r}))$ be the Pareto curve for this multi-objective linear program. With abuse of notation, we write $P(L_{dis}(M,s,\beta,\boldsymbol{r}))$ also for the set of memoryless strategies that are derived from the Pareto-optimal solutions of the multi-objective linear program. It follows that the Pareto curve $P(L_{dis}(M,s,\beta,\boldsymbol{r}))$ characterizes the set of memoryless Pareto-optimal points for the MDP with k discounted reward objectives. Since memoryless strategies suffice for Pareto optimality for discounted reward objectives (Theorem 3), the following lemma is immediate. Theorem 5 follows from Theorem 4 and Lemma 2.

Lemma 2. *Given an MDP M with k reward functions $\boldsymbol{r}$, a state s of M, and a discount factor $0 \leq \beta < 1$, let $L_{dis}(M,s,\beta,\boldsymbol{r})$ be the corresponding multi-objective linear program. The following assertions hold:*

1. *$P(L_{dis}(M,s,\beta,\boldsymbol{r})) = P_{dis}(M,s,\beta,\boldsymbol{r})$, that is, the Pareto curves for the linear program and the discounted reward MDP coincide.*
2. *For all $\epsilon > 0$ and all ϵ-approximate Pareto curves $P^\epsilon(L_{dis}(M,s,\beta,\boldsymbol{r}))$ of $L_{dis}(M,s,\beta,\boldsymbol{r})$, there is an ϵ-approximate Pareto curve $P^\epsilon_{dis}(M,s,\beta,\boldsymbol{r})$ such that $P^\epsilon(L_{dis}(M,s,\beta,\boldsymbol{r})) = P^\epsilon_{dis}(M,s,\beta,\boldsymbol{r})$.*

Theorem 5. *Given an MDP M with k reward functions $\boldsymbol{r}$ and a discount factor $0 \leq \beta < 1$, the following assertions hold:*

1. *For all $\epsilon > 0$, there exists an ϵ-approximate Pareto curve $P^{\epsilon}_{dis}(M, s, \beta, \boldsymbol{r})$ whose size is polynomial in $|M|$, $|\beta|$, $|\boldsymbol{r}|$, and $\frac{1}{\epsilon}$, and exponential in k.*
2. *For all $\epsilon > 0$, there exists an algorithm to construct an ϵ-approximate Pareto curve $P^{\epsilon}_{dis}(M, s, \beta, \boldsymbol{r})$ in time polynomial in $|M|$, $|\beta|$, $|\boldsymbol{r}|$, and $\frac{1}{\epsilon}$, and exponential in k.*

Theorem 5 shows that the Pareto curve can be efficiently ϵ-approximated. Recall that it follows from Example 1 that the set of Pareto-optimal points may be uncountable and the values may be irrational. Hence the ϵ-approximation of the Pareto curve is a useful finite approximation. The approximate Pareto curve allows us to answer trade-off queries about multi-objective MDPs. Specifically, given a multi-objective MDP M with k reward functions $\boldsymbol{r}$ and discount factor β, and a value profile $\boldsymbol{w} = \langle w_1, \ldots, w_k \rangle$, we can check whether $\boldsymbol{w}$ is ϵ-close to a Pareto-optimal point at state s by constructing $P^{\epsilon}_{dis}(M, s, \beta, \boldsymbol{r})$ in polynomial time, and checking that there is some strategy in $P^{\epsilon}_{dis}(M, s, \beta, \boldsymbol{r})$ whose payoff profile is ϵ-close to $\boldsymbol{w}$.

5 Pareto Realizability

In this section we study two related aspects of multi-objective MDPs: Pareto realizability, and pure memoryless Pareto realizability. The *Pareto realizability problem* asks, given a multi-objective MDP M with reward functions $\boldsymbol{r} = \langle r_1, \ldots, r_k \rangle$ and discount factor $0 \leq \beta < 1$, a state s of M, and a value profile $\boldsymbol{w} = \langle w_1, \ldots, w_k \rangle$ of k rational numbers, whether there exists a strategy σ such that $\mathrm{Val}^{\sigma}_{dis}(\boldsymbol{r}, s, \beta) \geq \boldsymbol{w}$. Observe that such a strategy exists if and only if there is a Pareto-optimal strategy σ' such that $\mathrm{Val}^{\sigma'}_{dis}(\boldsymbol{r}, s, \beta) \geq \boldsymbol{w}$. Also observe that it follows from Lemma 2 that a value profile $\boldsymbol{w}$ is realizable if and only if it is realizable by a memoryless strategy. The *pure memoryless* Pareto realizability problem further requires this strategy to be pure and memoryless.

The Pareto realizability problem arises when certain target behaviors are required, and one wishes to check if they can be attained on the model. Pure Pareto realizability arises in situations, such as circuit implementations, where the implemented strategy does not have access to randomization.

Theorem 6. *The Pareto realizability problem for MDPs with multiple discounted reward objectives can be solved in polynomial time. The pure memoryless Pareto realizability problem for MDPs with multiple discounted reward objectives is NP-complete.*

Proof. We show that Pareto realizability is in polynomial time by reduction to linear programming. The reduction is obtained as follows: along with the constraints defined by Equation (1) we add the constraints

$$w_i \leq \sum_{t \in S} \sum_{a \in A} x(t, a) \cdot r_i(t, a), \quad \text{for } i \in \{ 1, \ldots, k \}.$$

The original constraints from Equation (1) provide constraints on the discounted frequencies. The additional new constraints ensure that the payoff value for each reward function r_i is greater than or equal to the corresponding profile value w_i. Thus, if the set is consistent, then the answer to the Pareto realizability problem is "yes," and if inconsistent, the answer is "no." Consistency of this set can be checked in polynomial time using linear programming.

Pure and memoryless Pareto realizability is in NP since we can guess a pure memoryless strategy and compute its payoff values in polynomial time. We can then check that each payoff value is greater than or equal to the given profile value. It is NP-hard by reduction from subset sum. The subset sum problem takes as input natural numbers $\{ a_1, \ldots, a_n \}$, and a natural number p, and asks if there exist $v_1, \ldots, v_n$ in $\{ 0, 1 \}$ such that $a_1 \cdot v_1 + \cdots + a_n \cdot v_n = p$. It is NP-complete [3].

For an instance of the subset sum problem, we construct an MDP with two reward functions as follows. We assume for clarity that $\beta = 1$. The construction can be adapted for any fixed discount factor by suitably scaling the rewards. The MDP has $n + 1$ states, numbered from 1 to $n + 1$. We fix the start state to be 1. There are two actions, L and R. The transition relation is deterministic, for state $i \in \{ 1, \ldots, n \}$, we have $\delta(i, L)(i + 1) = \delta(i, R)(i + 1) = 1$. For state $n + 1$, we have $\delta(n + 1, L)(n + 1) = \delta(n + 1, R)(n + 1) = 1$. The reward function r_1 is defined as $r_1(i, L) = a_i$, and $r_1(i, R) = 0$ for $i \in \{ 1, \ldots, n \}$, and $r_1(n + 1, L) = r_1(n+1, R) = 0$. Similarly, the reward function r_2 is defined as $r_2(i, R) = a_i$, and $r_2(i, L) = 0$ for $i \in \{ 1, \ldots, n \}$, and $r_2(n + 1, L) = r_2(n + 1, R) = 0$. We now ask if the value profile $(p, \sum_i a_i - p)$ is pure Pareto realizable for this MDP. From the construction, it is clear that this profile is pure memoryless Pareto realizable iff the answer to the subset sum problem is "yes". In fact, the pure strategy that realizes the profile provides the required v_i's: if action L is played at state i, then $v_i = 1$, else $v_i = 0$. Since the MDP is a DAG, the hardness construction holds if we require the realizing strategy to be pure (not necessarily memoryless). ∎

The *pure* Pareto realizability problem requires the realizing strategy to be pure, but not necessarily memoryless. It follows from the reduction given above that the pure Pareto realizability problem for MDPs with multiple discounted reward objectives is NP-hard; however, we do not have a characterization of the exact complexity of the problem.

6 Limit-Average Reward Objectives

We now briefly discuss the class of limit-average reward objectives, which is widely studied in the context of MDPs. Given a reward function r, the *limit-average reward* for a strategy σ at an initial state s is $\mathrm{Val}^{\sigma}_{avg}(r, s) = \limsup_{k\to\infty} \frac{1}{k} \cdot \sum_{t=0}^{k} \mathbb{E}^{\sigma}_{s} [r(X_t, \theta_t)]$. With this definition, Theorem 1 holds for a single limit-average reward objective, and Proposition 1 extends to multiple limit-average reward objectives. Moreover, a simple adaptation of Example 1 shows that the pure strategies do not suffice for Pareto optimality for limit-average reward objectives. Unfortunately, Theorem 3 does not generalize. Example 2 below shows that for limit-average reward objectives, the family of

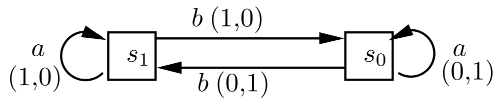

Fig. 2. MDP for Example 2

memoryless strategies does not capture all Pareto-optimal strategies. However, it is still possible that the Pareto curve for limit-average reward objectives can be approximated in polynomial time. This remains an open problem.

Example 2. Fig. 2 shows an MDP with two actions a and b, and two reward functions r_1 and r_2. The transitions and the respective rewards are shown as labeled edges in the figure. Consider the limit-average reward objectives for r_1 and r_2. Given any memoryless strategy σ_m, at s_0 we have $\mathrm{Val}^{\sigma_m}_{avg}(s_0, r_1) + \mathrm{Val}^{\sigma_m}_{avg}(s_0, r_2) = 1$. We now consider the following strategy σ, which is played in rounds. In round j, the strategy σ first goes to state s_1, chooses action a (i.e., stays in s_1) unless the average for reward r_1 is at least $1 - \frac{1}{j}$, then goes to state s_0, chooses action a unless the average reward for reward r_2 is at least $1 - \frac{1}{j}$, and then proceeds to round $j+1$. Given σ, we have $\mathrm{Val}^{\sigma}_{avg}(s_0, r_1) = 1$ and $\mathrm{Val}^{\sigma}_{avg}(s_0, r_2) = 1$. There is no memoryless Pareto-optimal strategy to achieve this value vector. ∎

References

1. O. Etzioni, S. Hanks, T. Jiang, R.M. Karp, O. Madari, and O. Waarts. Efficient information gathering on the internet. In *FOCS 96*, pages 234–243. IEEE, 1996.
2. J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer, 1997.
3. M.R. Garey and D.S. Johnson. *Computers and Intractability*. W.H. Freeman, 1979.
4. R. Hartley. *Finite Discounted Vector Markov Decision Processes*. Technical Report, Department of Decision Theory, Manchester University, 1979.
5. J. Koski. Multicriteria truss optimization. In *Multicriteria Optimization in Engineering and in the Sciences*. 1988.
6. G. Owen. *Game Theory*. Academic Press, 1995.
7. C.H. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *FOCS 00*, pages 86–92. IEEE, 2000.
8. M.L. Puterman. *Markov Decision Processes*. Wiley, 1994.
9. R. Szymanek, F. Catthoor, and K. Kuchcinski. Time-energy design space exploration for multi-layer memory architectures. In *DATE 04*. IEEE, 2004.
10. D.J. White. Multi-objective infinite-horizon discounted Markov decision processes. *Journal of Mathematical Analysis and Applications*, 89:639–647, 1982.
11. P. Yang and F. Catthoor. Pareto-optimization based run time task scheduling for embedded systems. In *CODES-ISSS 03*, pages 120–125. ACM, 2003.