Hedging Bets in Markov Decision Processes*

Rajeev Alur[1], Marco Faella[2], Sampath Kannan[3], and Nimit Singhania[4]

1 University of Pennsylvania, Philadelphia, USA
2 Università di Napoli "Federico II", Naples, Italy
3 University of Pennsylvania, Philadelphia, USA
4 University of Pennsylvania, Philadelphia, USA

Abstract

The classical model of Markov decision processes with costs or rewards, while widely used to formalize optimal decision making, cannot capture scenarios where there are multiple objectives for the agent during the system evolution, but only one of these objectives gets actualized upon termination. We introduce the model of *Markov decision processes with alternative objectives* (MDPAO) for formalizing optimization in such scenarios. To compute the strategy to optimize the expected cost/reward upon termination, we need to figure out how to balance the values of the alternative objectives. This requires analysis of the underlying infinite-state process that tracks the accumulated values of all the objectives. While the decidability of the problem of computing the exact optimal strategy for the general model remains open, we present the following results. First, for a Markov chain with alternative objectives, the optimal expected cost/reward can be computed in polynomial-time. Second, for a single-state process with two actions and multiple objectives we show how to compute the optimal decision strategy. Third, for a process with only two alternative objectives, we present a reduction to the minimum expected accumulated reward problem for one-counter MDPs, and this leads to decidability for this case under some technical restrictions. Finally, we show that optimal cost/reward can be approximated up to a constant additive factor for the general problem.

1998 ACM Subject Classification G.3 Probability and Statistics

Keywords and phrases Markov decision processes, Infinite state systems, Multi-objective optimization

Digital Object Identifier 10.4230/LIPIcs.CSL.2016.29

1 Introduction

The mathematical model of *Markov decision processes* (MDP) is suitable for modeling decision making in situations where the evolution of a system is partly probabilistic and partly controlled by strategic choices. To define the notion of an *optimal strategy* we need to associate costs (or equivalently, rewards) with an execution of an MDP. Traditionally, this is done by associating a numerical value with each action in each state, and the cost of a terminating execution is the sum of the costs of all the decisions made along the way. The optimization problem then is to minimize the expected cost (or equivalently, maximize the expected reward) over the set of all strategies. It is well known that there exists a memoryless optimal strategy, where-in the globally optimal decision at any step during an execution is a function of the current state, and the optimization problem can be solved in polynomial-time using linear programming [3].

* This research was partially supported by NSF Expeditions award CCF 1138996.

25th EACSL Annual Conference on Computer Science Logic (CSL 2016).
Editors: Jean-Marc Talbot and Laurent Regnier; Article No. 29; pp. 29:1–29:20
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

While this classical framework is used in a variety of disciplines such as optimal control, finance, and robotic motion planning, it cannot capture scenarios where there are multiple objectives to optimize but only one of these gets realized in the end. This occurs commonly in real life situations, when there are multiple goals only one of which is achieved in the end. Examples of such scenarios include a candidate applying for jobs, a student applying to schools or a company marketing its product to different audiences. As an illustrative scenario, imagine a venture capitalist (VC) investing in pharmaceutical research, where there are multiple companies all competing to develop a vaccine for malaria. At each point in time, the investor has several choices for how much to invest in each company's research. When one company succeeds in creating the vaccine, it patents the discovery and the investor reaps financial benefits proportional to the total amount (s)he has invested in that company, but does not get any reward for investments in the other companies. We can model the evolution of the system as an MDP. Each state of the system corresponds to the current conditions of all the companies. In each state, each company has a probability of succeeding resulting in termination, or the system continues to evolve probabilistically, partially influenced by the VC's investment choice.

The optimal investment strategy is not immediately obvious and is contingent on the company that succeeds in creating the vaccine. Particularly, it depends on the dynamics of the competition and how investment influences each company. If the competition is positive where investing in one firm boosts the other companies to improve their research, then spreading the investment to each firm is optimal. Whereas, if the competition is negative and investing in one firm demotivates others, then investing in a single best firm is optimal. Note that an objective here is to improve the research of a company and depending on the dynamics, the VC might decide how to balance optimization of every objective.

This scenario can be represented in our formal model of MDPs with alternative objectives, where the MDP is augmented with a set of registers corresponding to the different objectives. At each step, based on the current state and the chosen action, each register is updated by a specified integer amount. Then, following action-dependent transition probabilities, the process either continues in another state or terminates. When terminating, the value of one of the registers is probabilistically chosen as the cost/reward of the whole path. The optimization problem then is to minimize the expected cost or equivalently, maximize the expected reward upon termination over the set of all strategies. In this work, we focus on costs and minimization of the expected cost.

It turns out that for an MDP with alternative objectives, the optimal decision at any step depends not only on the current state but also on the register values. Since the space of register values is unbounded, analysis is challenging. For a Markov chain (that is, an MDP with a single action), we show that the expected cost in a given state is a *linear* function of the register values, whose coefficients can be computed in polynomial-time by solving a system of linear equations (see Section 4).

For an MDP, the optimal expected cost is no longer a linear function of the register values. To solve this general case, in Section 7, we present an *approximation* algorithm that can approximate the optimal expected cost with a specified error ϵ in pseudo-polynomial time, that is, polynomial in the number of states, actions, and the binary encoding of probabilities and ϵ, but exponential in the number of registers and binary encoding of register updates.

We present exact solutions for two special cases: systems with a single state and two actions (Section 5), and two-register systems with an arbitrary number of states and actions, subject to a mild condition (Section 6).

For systems with a single state and two actions, we observe that the choice of the optimal action depends on a *linear* function P of register values with one action being optimal when

P is positive and the other action when P is negative or zero. Moreover, only a finite number of distinct values for P needs to be taken into account, so that the minimal cost problem can be solved by a reduction to a Markov chain with alternative objectives.

When instead the input system has only two registers, we show that optimal strategies only need to track the difference between register values, allowing us to relate our model to one-counter MDPs of Bradzil et al. [6, 5]. However, this is not sufficient to achieve decidability, as the corresponding problem for one-counter MDPs has not been solved either. If the input system is additionally *tie-less* (as defined in Section 6), optimal strategies issue the same action whenever the difference between the registers is greater, in absolute value, than a certain threshold. Thanks to this property, we can reduce our problem to a bounded number of expected accumulated reward problems for probabilistic one-counter automata, which are analogous to one-counter MDPs with a single action. Since the latter problem is decidable, we obtain decidability of our original question.

The exact solution for the general case of MDPs with alternative objectives, even establishing decidability, remains an open problem.

Related work

Optimization problems for MDPs with different cost criteria have been studied extensively (see [10] for an overview and [11] for applications to computer-aided verification), but we are not aware of existing results directly relevant to the model we propose. A closely related model is that of MDPs with multiple objectives (see [12] for an overview), where multiple objectives similar to our model are associated with the MDP. These objectives are mixed together via a *scalarization* function to get the final expected cost/reward, which does not correspond to choosing a single cost/reward based on the final state that we want to model.

Our model with alternative costs is a special case of *cost register automata* that associate numerical costs with strings [1], and for such automata, optimization in a two-player game (without probabilistic transitions) is solvable in PSPACE [2]. Optimization problems for MDPs where the state-space is augmented with an unbounded counter have been studied recently [6, 5, 8]. The reduction of the special case of our model considered in Section 6 has similar state-space, but is different since the cost upon termination is proportional to the counter value.

2 Model

We describe here the model of Markov decision processes with alternative objectives (MDPAO). As in a traditional MDP, the process consists of a set of states and a set of actions. At a state, an action is chosen and based on the action, the process probabilistically transitions to the next state or terminates. Upon termination, the cost of the run of the process is calculated and this is where our model differs from the standard MDP. The process maintains a set of registers that start off at some initial values and are updated at every step depending on the action chosen at the current state. For each register and each action, the update consists of the addition of some integer, possibly negative, to the register value and upon termination, the value of one of the registers is probabilistically chosen as the cost. Given an initial state and the initial values for registers, we consider the problem of computing the minimum expected cost of a run of the process for the optimal choice of actions at each state.

We explain this further using an MDPAO with a single state q, two actions α and β and two registers x and y as shown in Figure 1. On choosing action α (β), with probability 0.5, the process returns back to q and increments x (y) by 1, and with probability 0.5, it

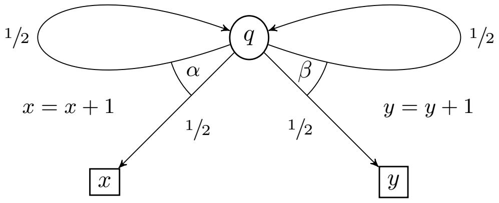


Figure 1 Example MDPAO with two registers x and y.

terminates with value of register x (y) as the cost. A possible path when the process is started at q with $(x, y) = (0, 0)$ is $\rho = (q, (0, 0)) \xrightarrow{\alpha} (q, (1, 0)) \xrightarrow{\beta} (q, (1, 1)) \xrightarrow{\alpha} x$, i.e., the process returns to q with $(x, y) = (1, 0)$ on choosing α, and then with $(x, y) = (1, 1)$ on choosing β and finally, it terminates with value of register $x = 1$ as the cost, on choosing α.

Now we formally define the model. An MDPAO $\mathcal{M}$ consists of the following:

- Q, a finite set of states.
- X, a finite set of registers with $\mathcal{V}$, the set of valuation functions $X \to \mathbb{Z}$ that map registers to their values.
- Γ, a finite set of actions.
- $\delta : Q \times \Gamma \times X \to \mathbb{Z}$, a function that defines the updates to the register values on a state transition. Note that we use $\delta(q, \alpha)$ to refer to a function that maps registers to their updates on action α at state q.
- $p : Q \times \Gamma \times (Q \cup X) \to [0, 1]$, a function that defines the probability of transition to a state or termination with a register value, given a state and the action selected at the state. The probability of transition from q to a state q' when action α is chosen is given by $p(q, \alpha, q')$ and the probability that the process terminates with value of register x as the cost is given by $p(q, \alpha, x)$. Note that, probabilities of transitions out of a state sum to 1, i.e., $\sum_{q' \in Q} p(q, \alpha, q') + \sum_{x \in X} p(q, \alpha, x) = 1$. We assume that $\sum_{x \in X} p(q, \alpha, x) > 0$ for all $q \in Q$ and $\alpha \in \Gamma$ to ensure that probability that a process does not terminate decreases exponentially with the number of steps.

Strategy. A *(full) strategy* is a function $\sigma : Q^+ \times \mathcal{V} \to \Gamma$, that given a sequence of states and initial valuation of registers determines the next action to be chosen at the current state. A *path-oblivious* strategy is a strategy that only depends on the current state and the current value of the registers and hence is a function of type $\sigma : Q \times \mathcal{V} \to \Gamma$. Given a strategy, the choice of action at each state is fixed and the Markov decision process is transformed into a Markov chain.

Path. A *(finite) path* $\rho = (q_0, \nu_0) \xrightarrow{\alpha_0} (q_1, \nu_1) \xrightarrow{\alpha_1} \dots \xrightarrow{\alpha_{n-1}} (q_n, \nu_n) \xrightarrow{\alpha_n} x, q_i \in Q, \alpha_i \in \Gamma, \nu_i \in \mathcal{V}, x \in X$, is a sequence of states, register values and actions chosen such that, for every transition $(q_i, \nu_i) \xrightarrow{\alpha_i} (q_{i+1}, \nu_{i+1})$, its probability is greater than 0, i.e., $p(q_i, \alpha_i, q_{i+1}) > 0$ and $\nu_{i+1} = \nu_i + \delta(q_i, \alpha_i)$, and the path terminates with value of register x with a probability greater than 0, i.e., $p(q_n, \alpha_n, x) > 0$. A path ρ is *consistent* with a strategy σ if every action α_i chosen in ρ is consistent with σ, i.e., $\alpha_i = \sigma(q_0 q_1 \dots q_i, \nu_0)$. The probability of the process following the path ρ is given by $Pr(\rho) = (\prod_{i=0}^{n-1} p(q_i, \alpha_i, q_{i+1}))p(q_n, \alpha_n, x)$. The cost of the path ρ is defined as $f(\rho) = \nu_n(x)$.

Cost. Let $\Pi_\sigma(q,\nu)$ be the set of all paths that start in $q \in Q$ with register values ν and are consistent with σ, i.e., $\Pi_\sigma(q,\nu) = \{\rho | (q_0,\nu_0) = (q,\nu), \rho \text{ is consistent with } \sigma\}$. Given an initial valuation of registers ν and a strategy σ, the expected cost of a run started from a state q is $f_\sigma(q,\nu) = \sum_{\rho \in \Pi_\sigma(q,\nu)} Pr(\rho) \cdot f(\rho)$.

In the example shown in Figure 1, suppose the process starts with $\nu(x) = 0$ and $\nu(y) = 0$. For a strategy that always chooses α i.e. $\sigma(q^i,\nu) = \alpha$, the process terminates after k steps with probability 0.5^k and cost $k-1$ and thus, the expected cost is $(0\times 0.5+1\times 0.5^2+2\times 0.5^3+\dots) = 1$. Similarly for a strategy that always chooses β, the expected cost is 1. Now for a strategy that alternately chooses α and β, i.e., $\sigma(q^{2i},\nu) = \alpha$ and $\sigma(q^{2i+1},\nu) = \beta$, we get a lower expected cost of $\frac{1}{3}$. In fact, this is the minimum expected cost. Note that, unlike traditional MDPs, the optimal strategy chooses multiple actions at state q.

Problem. Given an MDPAO $\mathcal{M}$, we consider here the problem of computing the minimum expected cost of a run of $\mathcal{M}$ started in an initial state $q \in Q$ with initial valuation of registers $\nu \in \mathcal{V}$. Let $f(q,\nu)$ denote this minimum expected cost and σ^* the optimal strategy, i.e., $f(q,\nu) = \min_\sigma f_\sigma(q,\nu), \sigma^* = \text{argmin}_\sigma f_\sigma(q,\nu)$.

Variations of the model. The above model can be further extended as follows. A first extension is one where the update function δ depends also on the next state, along with the current state and the action chosen. A second extension is one where, upon termination, the cost is a non-negative linear combination of register values rather than a single register's value. Note that our algorithms generalize to these extensions immediately and to keep the presentation simple, we describe solutions only for the model defined above.

3 Useful results

We describe here some results that are useful in solving the problem of computing the minimum expected cost.

We first show that $|f_\sigma(q,\nu)|$ is bounded by a linear function of $\max_{x\in X} |\nu(x)|$ for all strategies σ and thus, $|f(q,\nu)|$ is bounded by this function. We use this result to compute an approximation for the minimum expected cost in Section 7 and to prove subsequent results in this section. Let p_M be the maximum probability of continuation and δ_M be magnitude of maximum change made to any register, in one step of the process. We state the result in the following lemma.

▶ **Lemma 1.** *Given $\nu \in \mathcal{V}$, for all $q \in Q$ and strategies σ,*

$$|f_\sigma(q,\nu)| \leq \left(\frac{\max_{x\in X} |\nu(x)|}{1-p_M} + \frac{p_M \delta_M}{(1-p_M)^2} \right) \triangleq B(\max_{x\in X} |\nu(x)|),$$

where $p_M = \max_{q\in Q, \alpha\in\Gamma} \left(\sum_{q' \in Q} p(q,\alpha,q') \right)$ and $\delta_M = \max_{q\in Q,\alpha\in\Gamma,x\in X} |\delta(q,\alpha,x)|$.

Proof. We claim that for any strategy σ and for all $i \geq 0$, the probability that the process does not terminate in first i steps, p_i is bounded by p_M^i, the absolute value of any register after i steps, ν_i is bounded by $(\max_{x\in X} |\nu(x)| + i\delta_M)$ and thus, the absolute value of the expected cost paid in the $(i+1)$th step by the process, c_i, is bounded by $p_M^i(\max_{x\in X} |\nu(x)| + i\delta_M)$.

We prove this by induction. It is trivially true when $i = 0$. Assuming it is true in the ith step, we prove that it is also true in $(i+1)$th step. Probability that the process continues to the next step at the $(i+1)$th step at any state and for any action chosen is less than p_M and thus, $p_{i+1} \leq p_i p_M = p_M^{i+1}$. Similarly, the register values can be changed by at most δ_M

in a step and thus, the absolute value of registers after $(i+1)$ steps must be bounded by $\max_{x\in X}|\nu(x)|+(i+1)\delta_M$. Note that the absolute value of expected cost paid at a state q with register values ν' for an action α is

$$\big|\sum_{x\in\Gamma} p(q,\alpha,x)\nu'(x)\big| \;\leq\; \big(\max_{x\in X}|\nu'(x)|\big)\sum_{x\in\Gamma} p(q,\alpha,x) \;\leq\; \max_{x\in X}|\nu'(x)|.$$

Therefore, c_{i+1} is bounded by the maximum probability to reach the $(i+2)$th step × the expected cost paid in the $(i+2)$th step $\leq p_M^{i+1}(\max_{x\in X}|\nu(x)|+(i+1)\delta_M)$.

Now, the absolute value of the total expected cost of strategy σ is less than $\sum_{i=0}^{\infty} c_i \leq \sum_{i=0}^{\infty}\big(p_M^i(\max_{x\in X}|\nu(x)|+i\delta_M)\big)$. Since the probability of termination in each step is strictly greater than 0, the maximum probability $p_M<1$ and the required result follows. ◀

Next, minimum expected cost $f(q,\nu)$ can be expressed recursively using the costs at the next step, $f(q',\nu+\delta(q,\alpha))$. Let a cost function be a function $Q\times\mathcal{V}\to\mathbb{R}$ that maps a state and valuation of registers to a real valued cost and let the set of cost functions be $\mathcal{C}$. We define an operator $T:\mathcal{C}\to\mathcal{C}$ as follows.

$$Tg(q,\nu)=\min_{\alpha\in\Gamma}\Big(\sum_{x\in X} p(q,\alpha,x)\nu(x)+\sum_{q'\in Q} p(q,\alpha,q')g(q',\nu+\delta(q,\alpha))\Big)$$

The minimum expected cost function f is a fixed point of this operator, i.e., $Tf=f$. In fact, we can show that a strategy σ is an optimal strategy if f_σ is a fixed point of T, as stated in Lemma 2. Our proof for Lemma 2 closely follows the proof by Bertsekas et al in [4] to show that the recursive equation describing the stochastic shortest path problem has a unique solution which represents the optimal cost vector. We use this lemma to compute the optimal strategies in Sections 5 and 6. Another consequence of this lemma is that we can limit our investigation to path-oblivious strategies, i.e., strategies that only depend on the current state and the current value of the registers.

▶ **Lemma 2.** *Given a strategy σ in an MDPAO $\mathcal{M}$, the cost function f_σ is a fixed point of T, i.e., $Tf_\sigma(q,\nu)=f_\sigma(q,\nu)$ for all $q\in Q$ and $\nu\in\mathcal{V}$, if and only if σ is an optimal strategy and f_σ is the minimum expected cost function.*

Proof. To prove this lemma, we show that set of the cost functions, realizable by strategies, forms a partially ordered set such that the operator T has a unique fixed point in this set. Since the minimum expected cost function must be the least fixed point of T, any fixed point of T is the required minimum expected cost function.

For a strategy σ, we define a new operator on cost functions, $T_\sigma:\mathcal{C}\to\mathcal{C}$, where $T_\sigma g$ computes the cost of using strategy σ for one step and then paying the cost as given by g. Let $\alpha=\sigma(q,\nu)$, we have:

$$T_\sigma g(q,\nu)=\sum_{x\in X} p(q,\alpha,x)\nu(x)+\sum_{q'\in Q} p(q,\alpha,q')g(q',\nu+\delta(q,\alpha))$$

We also define a complete partial order L on cost functions as follows. Let $f_\top,f_\bot\in\mathcal{C}$ be the two cost functions defined by $f_\top(q,\nu)=B(\max_{x\in X}|\nu(x)|)$ and $f_\bot(q,\nu)=-B(\max_{x\in X}|\nu(x)|)$, where B is defined in Lemma 1. Further, $g\leq h$, where $g,h\in\mathcal{C}$, if for all $q\in Q,\nu\in\mathcal{V}$, $g(q,\nu)\leq h(q,\nu)$. Now, L is a complete partial order for relation $\leq$ on the set $\{g\in\mathcal{C}\mid f_\bot\leq g\leq f_\top\}$. Note that by Lemma 1, for all strategies σ, $f_\bot\leq f_\sigma\leq f_\top$ and hence $f_\sigma\in L$.

The operators T and T_σ are closed in L, i.e. $g\in L\Rightarrow Tg,T_\sigma g\in L$ since $Tf_\top\leq f_\top, T_\sigma f_\top\leq f_\top$ and $Tf_\bot\geq f_\bot, T_\sigma f_\bot\geq f_\bot$ (by Lemma 1). Further, operators T and T_σ

are monotonic and continuous, since they apply linear transformations and the minimum operator on cost functions, which preserve both properties. Hence, by Kleene's fixed point theorem, both T and T_σ must have a least fixed point in L given by $\lim_{k\to\infty} T^k f_\perp$ and $\lim_{k\to\infty} T_\sigma^k f_\perp$ respectively.

Next, we prove that T_σ has a unique fixed point in L. To prove this, we show that for all $g, h \in L$, $\lim_{k\to\infty} T_\sigma^k g(q,\nu) = \lim_{k\to\infty} T_\sigma^k h(q,\nu)$. Note that, $T_\sigma^k g$ corresponds to the expected cost of using strategy σ for first k steps and then terminating with the cost given by g at the $(k+1)$th step. Hence, the difference $T_\sigma^k g(q,\nu) - T_\sigma^k h(q,\nu)$ corresponds to the difference in costs paid at the $(k+1)$th step, since the cost paid till k steps is same for both $T_\sigma^k g(q,\nu)$ and $T_\sigma^k h(q,\nu)$. As described in the proof for Lemma 1, the probability of the process not terminating in first k steps is bounded by p_M^k and the magnitude of register values are bounded by $(\max_{x\in X} |\nu(x)| + k\delta_M)$. Since $g, h \in L$, we have:

$$|T_\sigma^k g(q,\nu) - T_\sigma^k h(q,\nu)| \le 2p_M^k B\big(\max_{x\in X} |\nu(x)| + k\delta_M\big).$$

Note that p_M^k decreases exponentially with k, whereas $B(\max_{x\in X} |\nu(x)| + k\delta_M)$ increases only linearly with k. Hence, $\lim_{k\to\infty}(T_\sigma^k g(q,\nu) - T_\sigma^k h(q,\nu)) = 0$, and thus T_σ has a unique fixed point in L.

Now we prove that T has a unique fixed point in L. Suppose T had two fixed points in L, say f_1, f_2. Then we can find strategies σ_1, σ_2 such that $T_{\sigma_1} f_1 = f_1$ and $T_{\sigma_2} f_2 = f_2$ by using the actions that minimize Tf_1 and Tf_2 respectively. Further, $Tf_1 \le T_{\sigma_2} f_1$ since Tf_1 corresponds to the minimum of the costs for different actions, while $T_{\sigma_2} f_1$ corresponds to one of these costs. This implies $f_1 = \lim_{k\to\infty} T^k f_1 \le \lim_{k\to\infty} T_{\sigma_2}^k f_1 = f_2$ and thus, $f_1 \le f_2$. By a symmetric argument, $f_2 \le f_1$. Hence, T must have a unique fixed point in L. The minimum expected cost f is the least fixed point of T in L and hence, any fixed point of T in L is the required minimum expected cost. ◀

Our final result shows that the optimal strategy σ^* depends only on the relative difference between the initial register values and not their absolute values. We state it in Lemma 3. We prove this by showing that $f(q, \nu + k) - k$ is also a fixed point of T and hence, must be equal to $f(q,\nu)$. Note that, this result can be used to reduce a model with $|X|$ registers into a model with $|X| - 1$ registers. We use it to simplify the problem in Section 6.

▶ **Lemma 3.** *Given an MDPAO $\mathcal{M}$, for all $q \in Q, \nu \in \mathcal{V}, k \in \mathbb{Z}$, it holds that $f(q, \nu + k) = f(q,\nu) + k$, where $(\nu + k)(x) = \nu(x) + k$ for all $x \in X$.*

Proof. We use Lemma 2 to prove this. Let $g \in \mathcal{C}$ be a function such that $g(q,\nu) = f(q,\nu+k) - k$ for all $q \in Q$ and $\nu \in \mathcal{V}$. Note that g may not lie in the complete partial order L as described in the proof for Lemma 2. However if we define the partial order L' using $f'_\top = f_\top + |k|(1 + \frac{1}{1-p_M})$ and $f'_\perp = f_\perp - |k|(1 + \frac{1}{1-p_M})$, the proof of Lemma 2 still follows and T must have a unique fixed point in L'. Also g belongs to L' since $|g(q,\nu)| \le |f(q,\nu+k)| + |k|$. Further,

$$\begin{aligned} Tg(q,\nu) &= T(f(q,\nu+k) - k) \\ &= Tf(q,\nu+k) - k \\ &= f(q,\nu+k) - k \\ &= g(q,\nu). \end{aligned}$$

Hence, g is a fixed point of T and thus, by Lemma 2, it must be the minimum expected cost function f. ◀

4 Markov chains with alternative objectives

We consider a special case of the problem when the set of actions consists of a single action, i.e. $\Gamma = \{\alpha\}$. Since f is a fixed point for the operator T as defined in Section 3, for all $q \in Q$ and $\nu \in \mathcal{V}$,

$$f(q,\nu) = \sum_{x\in X} p(q,x)\nu(x) + \sum_{q'\in Q} p(q,q')f(q',\nu+\delta(q)). \tag{1}$$

The above system of equations may have more than one solutions. However, we show that $f(q,\nu)$ is a linear function of the initial register values ν for all $q \in Q$, as stated in Lemma 4. The cost incurred by a path ρ consists of two components: the initial value of a register x, $\nu(x)$ and the updates to x along the path which are independent of ν. Also, the contribution of $\nu(x)$ to the expected cost $f(q,\nu)$ depends only on the probability with which paths starting at (q,ν) end in x, which again is independent of ν. Therefore, $f(q,\nu)$ must be linear in ν.

▶ **Lemma 4.** *Given an MDPAO with a single action, the minimum expected cost $f(q,\nu)$ is linear in ν, i.e., for all $q \in Q$ and $\nu \in \mathcal{V}$, it holds that $f(q,\nu) = \sum_{x\in X} a_{q,x}\nu(x) + f(q,\nu_0)$, where $\nu_0(x) = 0$ for all $x \in X$ and $a_{q,x}$ is the probability that a path starting at (q,ν) ends in register x.*

Proof. To prove the lemma we show that $f(q,\nu+\delta_x) = f(q,\nu) + a_{q,x}d$, where $\delta_x(x) = d$ and $\delta_x(y) = 0$ for all $y \in X \setminus \{x\}$, for some $d \in \mathbb{Z}$. Let $\rho = (q_0,\nu_0) \xrightarrow{\alpha_0} (q_1,\nu_1) \xrightarrow{\alpha_1} \dots \xrightarrow{\alpha_{n-1}} (q_n,\nu_n) \xrightarrow{\alpha_n} y$ and $\rho' = (q_0,\nu'_0) \xrightarrow{\alpha_0} (q_1,\nu'_1) \xrightarrow{\alpha_1} \dots \xrightarrow{\alpha_{n-1}} (q_n,\nu'_n) \xrightarrow{\alpha_n} y$ be paths that follow the same sequence of states but are started with different valuations of registers ν and $\nu+\delta_x$ respectively. We can see that $\nu'_i = \nu_i + \delta_x$ for all i, since the same changes are made to both ν and ν' along the way. Therefore, if a path ρ' terminates in x, denoted by $\rho' \rightsquigarrow x$, then $f(\rho') = \nu'_n(x) = \nu_n(x) + d = f(\rho) + d$, and $f(\rho') = f(\rho)$ otherwise. Also note that $Pr(\rho') = Pr(\rho)$.

By definition, $f(q,\nu+\delta_x) = \sum_{\rho'\in\Pi(q,\nu+\delta_x)}(Pr(\rho')f(\rho'))$. Since $f(\rho') = f(\rho)+d$ for all runs $\rho' \rightsquigarrow x$, $f(q,\nu+\delta_x) = f(q,\nu) + d\sum_{\rho\in\Pi(q,\nu),\rho\rightsquigarrow x} Pr(\rho)$ which implies $f(q,\nu+\delta_x) = f(q,\nu) + a_{q,x}d$. The required lemma follows immediately. ◀

Now, from (1) and Lemma 4, we can compute $f(q,\nu)$ by solving the following system of linear equations.

$$\begin{aligned} a_{q,x} &= p(q,x) + \sum_{q'\in Q} p(q,q')a_{q',x} \\ f(q,\nu_0) &= \sum_{q'\in Q} p(q,q')\left(\sum_{x\in X} a_{q',x}\delta(q,x) + f(q',\nu_0)\right) \end{aligned}$$

The above system consists of $|Q| \times (|X|+1)$ equations and variables and thus, can be solved in $O(|Q|^3|X|^3)$ time using a linear constraint solver.

▶ **Theorem 5.** *The problem of computing $f(q,\nu)$ for $q \in Q$ and $\nu \in \mathcal{V}$ for an MDPAO $\mathcal{M}$ with a single action can be solved in $O(|Q|^3|X|^3)$ time.*

5 Two action single state MDPs with alternative objectives

In this section, we solve the problem for an MDPAO $\mathcal{M}$ where $Q = \{q\}$ and $\Gamma = \{\alpha,\beta\}$. To simplify notation, let $a_x = p(q,\alpha,x)$, $b_x = p(q,\beta,x)$, $a_0 = p(q,\alpha,q)$ and $b_0 = p(q,\beta,q)$. Further, let the register updates be $\delta_\alpha = \delta(q,\alpha)$ and $\delta_\beta = \delta(q,\beta)$.

We observe that in this case while the minimum expected cost $f(\nu)$ is no longer a linear function of ν, the choice of optimal action in the optimal strategy does depend on a linear preference function P of current register values. So if $P(\nu) \leq 0$ then the optimal action at ν is α i.e. $\sigma^*(\nu) = \alpha$, and otherwise $\sigma^*(\nu) = \beta$.

To define P, we need to consider the change in preference ΔP on taking actions α and β. Since $P(\nu)$ is a linear function of ν, the change in preference $\Delta P(\delta) = P(\nu + \delta) - P(\nu)$ depends only on the change in register values δ. We define $\Delta P(\delta)$ as follows:

$$\Delta P(\delta) = \sum_{x \in X} \left(\frac{a_x}{1 - a_0} - \frac{b_x}{1 - b_0} \right) \delta(x).$$

Now $\Delta P(\delta_\alpha)$ and $\Delta P(\delta_\beta)$ capture the change in preference on taking the two actions, where δ_α and δ_β are the corresponding register updates. Depending on whether these values are positive or negative, we can have four possible scenarios.

To illustrate this further, consider the example shown in Figure 1 which is also an instance of this model. On choosing α, the process either terminates with value of register x or increments x and returns to q. Note that if the process does not terminate, the value of x increases and we are likely to pay a higher cost by choosing α in the next step. Hence, our preference P to choose β increases. Similarly on choosing β, our preference P to choose β decreases. This corresponds to the case where $\Delta P(\delta_\alpha) \geq 0$ and $\Delta P(\delta_\beta) < 0$ and the optimal strategy oscillates between choosing the two actions.

Now we give a concrete definition of the preference function P for the different scenarios described above. Let $f_w(\nu)$ be the cost of an infinite sequence of actions given by an infinite word $w = w_1 w_2 w_3 \ldots$ in $\{\alpha, \beta\}^\omega$. Note that $f_{\alpha w}(\nu) = \sum_{x \in X} a_x \nu(x) + a_0 f_w(\nu + \delta_\alpha)$ and $f_{\beta w}(\nu) = \sum_{x \in X} b_x \nu(x) + b_0 f_w(\nu + \delta_\beta)$. We can also compute $f_{\alpha^\omega}(\nu)$ by reducing $\mathcal{M}$ to a Markov chain where action α is chosen always, and $f_{\alpha^\omega}(\nu) = \sum_{x \in X} \left(\frac{a_x \nu(x)}{1 - a_0} + \frac{a_0 a_x \delta_\alpha(x)}{(1 - a_0)^2} \right)$. Using this, we can further compute $f_{\beta\alpha^\omega}(\nu)$. We can compute $f_{\beta^\omega}(\nu)$ and $f_{\alpha\beta^\omega}(\nu)$ similarly. Further, for all infinite words w, the difference $f_{\alpha\beta w}(\nu) - f_{\beta\alpha w}(\nu) = \sum_{x \in X}(((1 - b_0)a_x - (1 - a_0)b_x)\nu(x) + a_0 b_x \delta_\alpha(x) - b_0 a_x \delta_\beta(x))$ and is independent of w. We abbreviate this difference as $f_{\alpha\beta\cdot}(\nu) - f_{\beta\alpha\cdot}(\nu)$. We use these quantities to define $P(\nu)$ and the optimal strategy σ^* in Lemma 6.

▶ **Lemma 6.** *In a two action single state MDPAO $\mathcal{M}$, if $P(\nu) \leq 0$, then the optimal strategy at ν, $\sigma^*(\nu) = \alpha$ and otherwise $\sigma^*(\nu) = \beta$, where $P(\nu)$ is defined as follows:*

1. *If $\Delta P(\delta_\alpha) \geq 0$ and $\Delta P(\delta_\beta) < 0$, $P(\nu) = f_{\alpha\beta\cdot}(\nu) - f_{\beta\alpha\cdot}(\nu)$.*
2. *If $\Delta P(\delta_\alpha) \geq 0$ and $\Delta P(\delta_\beta) \geq 0$, $P(\nu) = f_{\alpha\beta^\omega}(\nu) - f_{\beta^\omega}(\nu)$.*
3. *If $\Delta P(\delta_\alpha) < 0$ and $\Delta P(\delta_\beta) < 0$, $P(\nu) = f_{\alpha^\omega}(\nu) - f_{\beta\alpha^\omega}(\nu)$.*
4. *If $\Delta P(\delta_\alpha) < 0$ and $\Delta P(\delta_\beta) \geq 0$, $P(\nu) = f_{\alpha^\omega}(\nu) - f_{\beta^\omega}(\nu)$.*

Proof. To prove this lemma, we show that in each case, f_{σ^*} is a fixed point of T and then by Lemma 2, σ^* is the optimal strategy. To compute the optimal action at ν, we need to consider the difference between the minimum expected cost on choosing action α and that on choosing β. This would require us to consider all possible sequences of actions starting from α and β and compute the expected cost for each sequence, which would be difficult. However, Lemma 2 helps us break down the problem into recursive cases, and consider only a few of sequences of actions for both α and β and whichever leads to a lower cost gives the desired optimal action. For Cases 1, 2 and 3, we also give alternate proofs, since f_{σ^*} is only recursively defined in these cases and a closed form representation is not available.

Case 4: $\Delta P(\delta_\alpha) < 0, \Delta P(\delta_\beta) \geq 0, P(\nu) = f_{\alpha^\omega}(\nu) - f_{\beta^\omega}(\nu)$. It is easy to see that $f_{\sigma^*}(\nu) = \min(f_{\alpha^\omega}(\nu), f_{\beta^\omega(\nu)})$. This is because, if α is preferred in the current state at ν, it is also preferred in all subsequent steps and thus, α^ω should lead to the minimum expected cost. Similarly, for β. We need to show that $Tf_{\sigma^*}(\nu) = f_{\sigma^*}(\nu)$ for all $\nu \in \mathcal{V}$. Suppose $P(\nu) \leq 0$. Then, $P(\nu + \delta_\alpha) \leq 0$ since $\Delta P(\delta_\alpha) < 0$ and thus, $f_{\sigma^*}(\nu + \delta_\alpha) = f_{\alpha^\omega}(\nu + \delta_\alpha)$. Therefore, the cost of taking action α at ν is $c_\alpha = f_{\alpha^\omega}(\nu)$. If $P(\nu + \delta_\beta) > 0$, then the cost of taking action β, at ν is $c_\beta = f_{\beta^\omega}(\nu)$ and therefore, $Tf_{\sigma^*}(\nu) = \min(c_\alpha, c_\beta) = f_{\sigma^*}(\nu)$. If $P(\nu + \delta_\beta) < 0$, then $c_\beta = f_{\beta\alpha^\omega}(\nu)$. But we can show that $f_{\alpha^\omega}(\nu) - f_{\beta\alpha^\omega}(\nu) = f_{\alpha^\omega}(\nu) - f_{\beta^\omega}(\nu) - (f_{\beta\alpha^\omega}(\nu) - f_{\beta^\omega}(\nu)) = P(\nu) - b_0 P(\nu + \delta_\beta) = (1 - b_0)P(\nu) - b_0 \Delta P(\delta_\beta) \leq 0$. This implies $f_{\alpha^\omega}(\nu) \leq f_{\beta\alpha^\omega}(\nu)$ and therefore, $Tf_{\sigma^*}(\nu) = f_{\sigma^*}(\nu)$. The other case, $P(\nu) > 0$ is symmetric and thus, σ^* is the optimal strategy.

Case 1: $\Delta P(\delta_\alpha) \geq 0, \Delta P(\delta_\beta) < 0, P(\nu) = f_{\alpha\beta\cdot}(\nu) - f_{\beta\alpha\cdot}(\nu)$. Note that the optimal strategy σ^* here oscillates between α and β. Thus, the cost f_{σ^*} can be defined only recursively and its exact representation can not be known a priori. Therefore, we use the recursive definition to show that it is the fixed point. We also give an alternate proof based on an exchange argument for clarity.

We define $f_{\sigma^*}(\nu)$ as follows. If $P(\nu) \leq 0, f_{\sigma^*}(\nu) = f_{\alpha\sigma^*}(\nu) = \sum_{x \in X} a_x \nu(x) + a_0 f_{\sigma^*}(\nu + \delta_\alpha)$ else, $f_{\sigma^*}(\nu) = f_{\beta\sigma^*}(\nu)$. We also assume that $f_{\sigma^*}(\nu)$ is the minimum expected cost i.e. $f_{\sigma^*}(\nu) \leq f_w(\nu)$ for all $w \in \{\alpha, \beta\}^\omega$ and $\nu \in \mathcal{V}$. Now we show that $Tf_{\sigma^*} = f_{\sigma^*}$. Suppose $P(\nu) \leq 0$. Then since $\Delta P(\delta_\beta) < 0, P(\nu + \delta_\beta) < 0$ and $f_{\sigma^*}(\nu + \delta_\beta) = f_{\alpha\sigma^*}(\nu + \delta_\beta)$ or α is the preferred action at $\nu + \delta_\beta$. Thus on taking action β at ν, the cost is $c_\beta = f_{\beta\alpha\sigma^*}(\nu)$. Since $P(\nu) \leq 0$, $f_{\alpha\beta\sigma^*}(\nu) \leq f_{\beta\alpha\sigma^*}(\nu)$ and thus, $f_{\alpha\beta\sigma^*}(\nu) \leq c_\beta$. Further by assumption, $c_\alpha = f_{\alpha\sigma^*}(\nu)$ is the minimum expected cost of taking action α at ν and thus, $c_\alpha \leq f_{\alpha\beta\sigma^*}(\nu) \leq c_\beta$. Hence $Tf_{\sigma^*}(\nu) = \min\{c_\alpha, c_\beta\} = f_{\alpha\sigma^*}(\nu) = f_{\sigma^*}(\nu)$. Further by assumption c_α and c_β are the minimum expected costs of taking actions α and β and thus $Tf_{\sigma^*}(\nu)$ is the minimum expected cost for all ν. The sub-case when $P(\nu) > 0$ is symmetric and thus, σ^* is the optimal strategy.

Alternate proof. We now show a proof based on an exchange argument. Suppose $P(\nu) \leq 0$. We can show that every infinite sequence of actions at ν can be transformed to a sequence starting with action α with a lower expected cost, by exchanging pairs $\beta\alpha$ with $\alpha\beta$. Formally, we show that for all infinite sequence of actions βw,, there exists w' such that $f_{\alpha w'}(\nu) \leq f_{\beta w}(\nu)$. First, we can show by induction on k that for all $k > 0, z \in \{\alpha, \beta\}^\omega$, $f_{\alpha\beta^k z}(\nu) \leq f_{\beta^k \alpha z}(\nu)$. Base case is implied by $P(\nu) \leq 0$. Assuming it is true for $k - 1$, we show that it holds for k. Since $\Delta P(\delta_\beta) \leq 0$, we can see that $P(\nu + (k-1)\delta_\beta) \leq 0$ and thus, $f_{\beta^{k-1}\alpha\beta z}(\nu) \leq f_{\beta^k \alpha z}(\nu)$. Further, by inductive hypothesis $f_{\alpha\beta^{k-1}z}(\nu) \leq f_{\beta^{k-1}\alpha z}(\nu)$ for all z and thus, $f_{\alpha\beta^k z}(\nu) \leq f_{\beta^k \alpha z}(\nu)$. Further, we can show that $f_{\alpha\beta^\omega}(\nu) \leq f_{\beta^\omega}(\nu)$ because, $f_{\alpha\beta^\omega}(\nu) - f_{\beta^\omega}(\nu) = \sum_{i=0}^{\infty} b_0^i P(\nu + i\delta_\beta) \leq 0$. Hence, for infinite sequence of actions, βw, there is a sequence starting with α that has a lower cost and thus, α must be the optimal action at ν. Similarly, we can show that when $P(\nu) > 0, \sigma^*(\nu) = \beta$.

Case 2: $\Delta P(\delta_\alpha) \geq 0, \Delta P(\delta_\beta) \geq 0, P(\nu) = f_{\alpha\beta^\omega}(\nu) - f_{\beta^\omega}(\nu)$. We again give two proofs, one based on fixed point and other based on an exchange argument like in Case 1.

We define $f_{\sigma^*}(\nu)$ as follows. If $P(\nu) > 0, f_{\sigma^*}(\nu) = f_{\beta^\omega}(\nu)$ else $f_{\sigma^*}(\nu) = f_{\alpha\sigma^*}(\nu)$. Also, like in Case 1, we assume that $f_{\sigma^*}(\nu)$ is the minimum expected cost i.e. $f_{\sigma^*}(\nu) \leq f_w(\nu)$ for all $w \in \{\alpha, \beta\}^\omega$ and $\nu \in \mathcal{V}$. We show that $Tf_{\sigma^*} = f_{\sigma^*}$. Suppose $P(\nu) \geq 0$. Then $P(\nu + \delta_\beta) \geq 0$, $P(\nu + \delta_\alpha) \geq 0$, and $c_\alpha = f_{\alpha\beta^\omega}(\nu)$, $c_\beta = f_{\beta^\omega}(\nu)$. Since $P(\nu) \geq 0$, $c_\alpha \geq c_\beta$

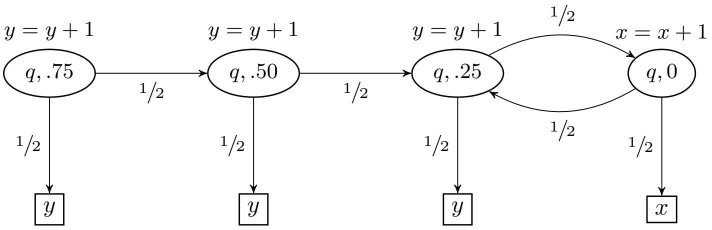


Figure 2 Markov chain conversion of example in Figure 1 when initially $\nu(x) = 8$ and $\nu(y) = 5$ where $f(\nu)$ is the expected cost at $(q, .75)$.

and $Tf_{\sigma^*}(\nu) = f_{\beta^\omega}(\nu) = f_{\sigma^*}(\nu)$. Next, suppose $P(\nu) < 0$. Then, if $P(\nu + \delta_\beta) > 0$, $c_\beta = f_{\beta^\omega}(\nu) > f_{\alpha\beta^\omega}(\nu) \geq c_\alpha$. If $P(\nu + \delta_\beta) \leq 0$, then $c_\beta = f_{\beta\alpha w}(\nu)$ for some $w \in \{\alpha, \beta\}^\omega$. However, $f_{\alpha\beta w}(\nu) - f_{\beta\alpha w}(\nu) = f_{\alpha\beta^\omega}(\nu) - f_{\beta\alpha\beta^\omega}(\nu) = f_{\alpha\beta^\omega}(\nu) - f_{\beta^\omega}(\nu) + f_{\beta^\omega}(\nu) - f_{\beta\alpha\beta^\omega}(\nu) = P(\nu) - b_0 P(\nu + \delta_\beta) = (1 - b_0)P(\nu) - b_0(1 - a_0)\Delta P(\delta_\beta) < 0$. Thus, $c_\beta > f_{\alpha\beta w}(\nu) \geq c_\alpha = f_{\alpha\sigma^*}(\nu)$. Hence, $Tf_{\sigma^*}(\nu) = f_{\sigma^*}(\nu)$ for all ν and therefore σ^* is the optimal strategy.

Alternate proof. We again use an exchange argument to give an alternate proof. First we show that the the optimal strategy at ν is of the form $\alpha^k\beta^\omega$ for some $k \geq 0$. Let $R(\nu) = f_{\alpha\beta w}(\nu) - f_{\beta\alpha w}(\nu)$. Note that $\Delta R(\nu) = (1 - b_0)\Delta P(\nu)$. If $R(\nu) \geq 0$, then by similar argument as in Case 1, $\sigma^*(\nu) = \beta$. In fact, $R(\nu + i\delta_\beta) \geq 0$ and hence, $\sigma^*(\nu + i\delta_\beta) = \beta$ for all $i \geq 0$. Therefore, the optimal sequence of actions at ν is β^ω or $f_{\sigma^*}(\nu) = f_{\beta^\omega}(\nu)$. Suppose $R(\nu) < 0$. We show that for each sequence of actions $w = w_1w_2\cdots \in \{\alpha, \beta\}^\omega$, $f_{\alpha^k\beta^\omega}(\nu) \leq f_w(\nu)$ for some k. Let ν_i represent the value of registers after the sequence of actions $w_1w_2\ldots w_i$. Let $l = \{i \mid R(\nu_i) \geq 0, R(\nu_j) < 0 \text{ for all } j < i\}$. Thus, l is the first index such that $R(\nu_l) \geq 0$ and for all $i < l, R(\nu_i) < 0$. From the argument above, cost of $w' = w_1w_2\ldots w_l\beta^\omega$ is lower than that of w. Further, we can show that cost of $\alpha^k\beta^{l-k}\beta^\omega$ is smaller than w'. Since $R(\nu_j) < 0$ for all $j < l$, every pair $\beta\alpha$ in $w_1w_2\ldots w_l$ can be exchanged with $\alpha\beta$ to improve the strategy and thus, give the required result. Thus, the optimal strategy is of the form $\alpha^k\beta^\omega$ for some $k \geq 0$.

If $P(\nu) < 0$, $f_{\alpha\beta^\omega}(\nu) < f_{\beta^\omega}(\nu)$ and hence, α is the optimal action at ν. Further, if $P(\nu) \geq 0$, then for all k, $P(\nu + k\delta_\alpha) \geq 0$, thus, $f_{\beta^\omega}(\nu) \leq f_{\alpha^k\beta^\omega}(\nu)$. Therefore, β is the optimal action at ν and hence, σ^* is the required optimal strategy.

Case 3: $\Delta P(\delta_\alpha) < 0, \Delta P(\delta_\beta) < 0, P(\nu) = f_{\alpha^\omega}(\nu) - f_{\beta\alpha^\omega}(\nu)$. This case is symmetric to Case 2 and can be proven similarly. ◀

Given a two action single state MDPAO $\mathcal{M}$, we compute the minimum expected cost $f(\nu)$ as follows. We first compute the optimal strategy using Lemma 6, then convert the MDP $\mathcal{M}$ into a Markov chain $\mathcal{M}'$ and use $\mathcal{M}'$ to compute the expected cost. In the conversion, since the optimal strategy at a state depends on $P(\nu)$, we associate this value with each state in $\mathcal{M}'$. We start with a state annotated with $P(\nu)$ and, based on the action chosen, transition to a state labelled with $P(\nu + \delta)$, where the registers are incremented by δ. Note that if we reach a state in $\mathcal{M}'$ such that $P(\nu) > 0$ and $\Delta P(\delta_\beta) \geq 0$, then we always choose β after this step, and thus transition back to the same state. Similarly when $P(\nu) \leq 0$ and $\Delta P(\delta_\alpha) < 0$. Figure 2 shows the transformation of the example in Figure 1 into a Markov chain where initially, $\nu(x) = 8$ and $\nu(y) = 5$ and $f(\nu)$ is the expected cost at state $(q, .75)$.

The converted chain consists of a cycle with $\left(\frac{L}{|\Delta P(\delta_\alpha)|} + \frac{L}{|\Delta P(\delta_\beta)|}\right)$ nodes, where L is the least common multiple of $|\Delta P(\delta_\alpha)|$ and $|\Delta P(\delta_\beta)|$. Note that the number of nodes is finite only

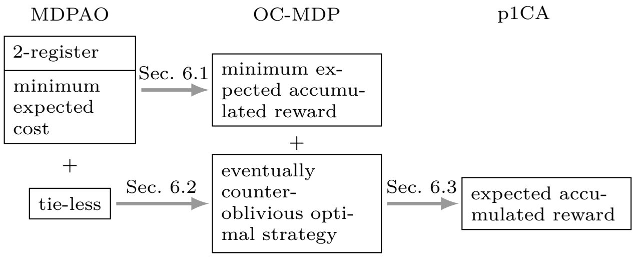

MDPAO
OC-MDP
p1CA
2-register
minimum expected cost
Sec. 6.1
minimum expected accumulated reward
+
+
tie-less
Sec. 6.2
eventually counter-oblivious optimal strategy
Sec. 6.3
expected accumulated reward

Figure 3 Diagram of the reductions between different models developed in Section 6.

if L is finite. Further, it might consist of a sequence of $O\Big(\frac{P(\nu)}{|\Delta P(\delta_\alpha)|}+\frac{P(\nu)}{|\Delta P(\delta_\beta)|}\Big)$ states leading to the cycle. Thus, time to compute cost function in the cycle is $O\Big(\Big(\frac{L}{|\Delta P(\delta_\alpha)|}+\frac{L}{|\Delta P(\delta_\beta)|}\Big)^3|X|^3\Big)$ and $f(\nu)$ can be computed in $O\Big(\Big(\frac{P(\nu)}{|\Delta P(\delta_\alpha)|}+\frac{P(\nu)}{|\Delta P(\delta_\beta)|}\Big)|X|\Big)$ time using the costs in the cycle.

▶ **Theorem 7.** *The problem of computing $f(q,\nu)$ for a two action single state MDPAO $\mathcal{M}$, i.e., $\Gamma=\{\alpha,\beta\}$ and $Q=\{q\}$, can be solved in time*

$$O\Big(\Big(\frac{L}{|\Delta P(\delta_\alpha)|}+\frac{L}{|\Delta P(\delta_\beta)|}\Big)^3|X|^3+\Big(\frac{P(\nu)}{|\Delta P(\delta_\alpha)|}+\frac{P(\nu)}{|\Delta P(\delta_\beta)|}\Big)|X|\Big),$$

where L is the l.c.m. of $|\Delta P(\delta_\alpha)|$ and $|\Delta P(\delta_\beta)|$. Note that if $|\Delta P(\delta_\alpha)|/|\Delta P(\delta_\beta)|$ is irrational, then L is infinite.

6 Two register MDPs with alternative objectives

In this section, we tackle the problem for MDPAOs with two registers x,y, and an arbitrary number of states and actions. Note that by Lemma 3, the optimal strategy depends only on the relative difference between the registers and the strategy needs to maintain a single counter that keeps track of the difference between the values of the two registers.

We show the following results in this section. First, we reduce our problem to the minimum expected accumulated reward problem for one-counter MDPs (OC-MDPs), which is another decision problem whose decidability status is open. Next, we show that for the class of *tie-less* MDPAOs, the optimal strategy is *eventually counter-oblivious*, i.e., there exists a natural number k such that in each state the strategy plays in the same way for all values of the counter higher than k. In turn, this property implies decidability by a reduction of the corresponding OC-MDP to the expected accumulated reward problem for probabilistic 1-counter automata (p1CAs). This sequence of results is summarized in Figure 3.

6.1 Reduction to one-counter Markov decision processes

The problem of computing the minimum expected cost of a 2-register MDPAO can be easily reduced to the minimum expected accumulated reward problem for one-counter Markov decision processes (OC-MDPs) [6, 5]. We employ OC-MDPs *with boundary*, which means that the counter value is always non-negative.

A one-counter MDP is a tuple $\mathcal{A}=(S,\Gamma,\Delta_0,\Delta_{>0})$ [1], where S is a finite set of control states, Γ is a finite set of actions, and Δ_0, $\Delta_{>0}$ are the transitions, with the intended meaning

[1] For technical convenience, our presentation of OC-MDPs includes explicit actions and rewards attached to transitions. It is straightforward to convert this form to the one of Brádzil et al. [6, 8], where rewards are represented by a *simple reward function*.

that Δ_0 applies when the counter is zero and $\Delta_{>0}$ applies otherwise. Each transition is a tuple of the type:

$$\text{(source, action, probability, counter update, reward, destination)}.$$

So, we have $\Delta_0 \subseteq S \times \Gamma \times [0,1] \times \{0,1\} \times \mathbb{Q}^+ \times S$ and $\Delta_{>0} \subseteq S \times \Gamma \times [0,1] \times \{-1,0,1\} \times \mathbb{Q}^+ \times S$. Moreover, for all $s \in S$ and $\gamma \in \Gamma$, the sum of the probabilities of all transitions in Δ_0 (resp., $\Delta_{>0}$) starting from s and γ is 1. A transition $(s, \alpha, p, d, r, s') \in \delta_{>0}$ signifies that when the system is in control state s with a positive value of its counter, action α may lead to state s' with probability p, while updating the counter value from n to $n+d$ and obtaining reward r.

The reward associated with a finite run is the sum of the rewards of each transition. The minimum expected accumulated reward problem takes as inputs an OC-MDP, two control states s, s', and an initial counter value n, and consists of computing the minimum over all policies of the expected reward along all runs that start in (s, n) and end in $(s', 0)$. As usual, by "computing a value" we mean decide whether such value is smaller than or equal to a given rational number.

Consider a 2-register MDPAO $\mathcal{M}$ with non-negative register updates. In order to convert it into an OC-MDP, we first restrict the register updates to the values $\{0,1\}$. This can be easily accomplished by adding more states and splitting a transition with updates $(+d_x, +d_y)$ into a sequence of $\max\{d_x, d_y\}$ transitions with $\{0,1\}$ updates. Notice that the newly added transitions violate the assumption that each transition carries a positive probability of termination. However, in the resulting system each *sequence of k transitions* carries a positive probability of termination, for a bounded k. Hence, the only consequence is a slight modification to the bound provided by Lemma 1.

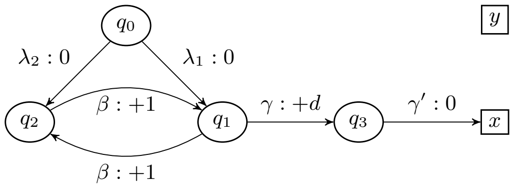
Once register updates have been simplified, the reduction is based on the following idea. Let $f(q,x,y)$ represent the minimum expected cost when the initial state is q and initial values of registers are given by x and y. By Lemma 3, we know that

$$f(q,x,y) = f(q, x-y, 0) + y = f(q, 0, y-x) + x. \tag{2}$$

Clearly, at least one of $x - y$ and $y - x$ is non-negative. By (2), to compute the minimum expected cost of an arbitrary configuration (q,x,y) it is sufficient to know the minimum expected cost of all configurations of the type $(q, z, 0)$ and $(q, 0, z)$ for all $z \geq 0$. All these configurations can be encoded by an OC-MDP, whose counter encodes z and whose control states are pairs (q, r), where $r \in \{x, y\}$ identifies which register has value z. An equivalent way to look at this encoding is that the counter stores the difference between the register that currently holds the highest value and the other register, and the flag r encodes which of the two registers holds the highest value.

It remains to encode the cost structure of the MDPAO. To this purpose, suppose that we are in the configuration $(q, z, 0)$, for some $z \geq 0$ (encoded by the OC-MDP state (q, x) with counter value z), and we want to simulate the effect of an action γ, which leads to state q' with probability $p(q, \gamma, q')$ and carries register updates $\delta(q, \gamma, x) = d_x$ and $\delta(q, \gamma, y) = d_y$. Then, Lemma 3 tells us that $f(q', z + d_x, d_y) = f(q', z + d_x - d_y, 0) + d_y$. So, assuming that $z + d_x - d_y \geq 0$, the corresponding OC-MDP will move with probability $p(q, \gamma, q')$ from state (q, x) to state (q', x), while updating the counter from z to $z + d_x - d_y$, and gaining an immediate reward of d_y. Two special states called *accrue* and *stop* model the termination of the MDPAO. We restrict the analysis to non-negative register updates so that the rewards in the OC-MDP will be non-negative too.

Formally, given a 2-register MDPAO $\mathcal{M}$, we define the corresponding OC-MDP $\mathcal{A}$ as follows. The set of control states S comprises pairs (q, r), where $q \in Q$ and $r \in \{x, y\}$,



action	$p(x)$	$p(y)$
λ_1	ε	ε
λ_2	ε	ε
β	$\frac{1}{4}$	$\frac{1}{4}$
γ	ε	0
γ'	1	0

Figure 4 A 2-register MDPAO. Edges are labeled with an action name followed by the update to variable x. The probabilities of terminating with x or y for each action are reported in the table.

plus the special states *accrue* and *stop*. The actions are the same as in the MDPAO. For all $q, q' \in Q$, $r \in \{x, y\}$, and $\gamma \in \Gamma$, the following transitions belong to Δ_0 (recall that $\delta(\cdot,\cdot,\cdot) \in \{0,1\}$):

$$((q,r),\gamma,p(q,\gamma,q'),\delta(q,\gamma,x)-\delta(q,\gamma,y),\delta(q,\gamma,y),(q',x)) \text{ whenever } \delta(q,\gamma,x) \geq \delta(q,\gamma,y)$$

$$((q,r),\gamma,p(q,\gamma,q'),\delta(q,\gamma,y)-\delta(q,\gamma,x),\delta(q,\gamma,x),(q',y)) \text{ whenever } \delta(q,\gamma,x) < \delta(q,\gamma,y)$$

$$(\mathit{accrue},\gamma,1,0,0,\mathit{stop})$$

Similarly, the following transitions belong to $\Delta_{>0}$:

$$((q,x),\gamma,p(q,\gamma,q'),\delta(q,\gamma,x)-\delta(q,\gamma,y),\delta(q,\gamma,y),(q',x))$$
$$((q,y),\gamma,p(q,\gamma,q'),\delta(q,\gamma,y)-\delta(q,\gamma,x),\delta(q,\gamma,x),(q',y))$$
$$(\mathit{accrue},\gamma,1,-1,1,\mathit{accrue})$$
$$(\mathit{stop},\gamma,1,-1,0,\mathit{stop})$$

Moreover, both in Δ_0 and in $\Delta_{>0}$ we find the following transitions, where $\neg r$ denotes the register different from r:

$$((q,r),\gamma,p(q,\gamma,r),0,0,\mathit{accrue}) \qquad ((q,r),\gamma,p(q,\gamma,\neg r),0,0,\mathit{stop})$$

As an example, consider the 2-register MDPAO in Figure 4. To keep the figure readable, we do not draw the edges connecting each state to the exit nodes x and y, except for action γ' from state q_3, because the only effect of γ' is to exit with value x. The corresponding probabilities are reported in the table on the r.h.s. of the figure. No action modifies register y, whereas updates to register x appear next to the action name. As an example, for action β taken from q_2 we have $p(q_2,\beta,x) = {}^1\!/_4$, $p(q_2,\beta,y) = {}^1\!/_4$, and $p(q_2,\beta,q_1) = {}^1\!/_2$, with register updates $\delta(q_2,\beta,x) = 1$ and $\delta(q_2,\beta,y) = 0$.

Figure 5 shows a fragment of the corresponding OC-MDP, starting from state (q_2, x), representing the original state q_2 when $y = 0$. Termination of the original MDP in register x is simulated by moving to *accrue*. In that state, the counter is repeatedly decremented until it reaches 0, with each iteration adding 1 to the reward. When the counter reaches 0, the OC-MDP moves to the final state *stop*. Termination in register y is simulated by moving directly to *stop*, as no further reward needs to be accrued, because we are assuming that $y = 0$ and the counter contains the value of register x.

▶ **Theorem 8.** *Let $\mathcal{M}$ be a 2-register MDPAO with non-negative register updates. The minimum expected cost $f(q,0,0)$ in $\mathcal{M}$ is equal to the minimum expected accumulated reward in $\mathcal{A}$ from $((q,x),0)$ to $(\mathit{stop},0)$.*

The minimum expected accumulated reward problem is a generalization of the minimum expected termination time problem considered in [8]. Neither problem is known to be

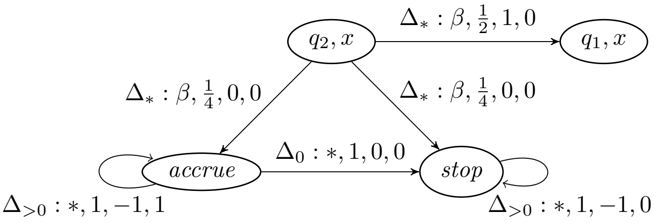

q_2, x
q_1, x
accrue
stop
$\Delta_* : \beta, \frac{1}{2}, 1, 0$
$\Delta_* : \beta, \frac{1}{4}, 0, 0$
$\Delta_* : \beta, \frac{1}{4}, 0, 0$
$\Delta_0 : *, 1, 0, 0$
$\Delta_{>0} : *, 1, -1, 1$
$\Delta_{>0} : *, 1, -1, 0$

Figure 5 A fragment of the OC-MDP corresponding to the MDPAO in Figure 4. Edges report action, probability, counter update, and reward. The asterisk is used as a wildcard.

decidable for OC-MDPs, whereas both are solvable in PSPACE for probabilistic 1-counter automata (p1CAs) [9, 7], which are equivalent to OC-MDPs with a single action. In Subsection 6.3, we employ the decidability on p1CAs to solve the minimum expected cost problem on a class of 2-register MDPAOs.

6.2 Tie-less MDPAOs

In this subsection, we introduce the class of *tie-less* MDPAOs and show that for this class there exists an optimal strategy that is eventually counter-oblivious.

Note that from equation (2), $f(q, x, y)$ can be computed from $f(q, x - y, 0)$ and thus it suffices to compute $f(q, n, 0)$, with $n \in \mathbb{Z}$. In the following we write $f(q, n)$ as a shortcut for $f(q, n, 0)$. A *simple strategy* is a function $\sigma : Q \to \Gamma$ and we denote by $\mathcal{S}$ the set of all simple strategies. An *ω-strategy* is an infinite sequence of simple strategies, i.e., an element of $\mathcal{S}^\omega$. Notice that given a full strategy σ, an initial state q and an initial valuation ν there exists at least one ω-strategy π that is equivalent to it. In particular, it holds $f_\sigma(q, \nu) = f_\pi(q, \nu)$. Hence, in order to compute $f(q, 0, 0)$ it is sufficient to consider ω-strategies rather than full strategies.

Similarly to Lemma 4, we can show that for all ω-strategies π, $f_\pi(q, n)$ is a linear function of n, i.e.: $f_\pi(q, n) = \mathit{slope}_{q,x}(\pi) \cdot n + f_\pi(q, 0)$, where $\mathit{slope}_{q,x}(\pi)$ is the probability that a path starting at q ends in register x under strategy π. Notice that $\mathit{slope}_{q,x}(\pi)$ generalizes the notation $a_{q,x}$ used in Section 4 to denote the probability of terminating with a specific register.

In fact, if the same simple strategy $\sigma \in \mathcal{S}$ is used at every step (equivalently, the ω-strategy σ^ω is used), the system becomes equivalent to an MDPAO with a single action and therefore $\mathit{slope}_{q,x}(\sigma^\omega)$ can be computed for all q in polynomial time as explained in Section 4. In particular, the simple strategy α with the minimum x-slope can be found by solving the following linear program, on the set of variables $\{a_q \mid q \in Q\}$.

$$\text{maximize: } \sum_{q \in Q} a_q \tag{3}$$

$$\text{subject to: } a_q \leq p(q, \gamma, x) + \sum_{q' \in Q} p(q, \gamma, q') a_{q'} \quad \forall\, q \in Q, \gamma \in \Gamma$$

The optimal solution a^* provides the minimal x-slope in each state. We say that the MDPAO is *x-tie-less* if there exists a unique simple strategy α that has the minimum x-slope from all states q. Equivalently, the MDPAO is x-tie-less if at a^* exactly one of the constraints for each state is tight, i.e., for all q there exists a unique $\alpha_q \in \Gamma$ such that

$$a^*_q = p(q, \alpha_q, x) + \sum_{q' \in Q} p(q, \alpha_q, q') a^*_{q'}. \tag{4}$$

We can similarly define the y-slope of a strategy as the probability of terminating in register y using that strategy. Since runs terminate with probability one, for each strategy the sum of the x-slope and the y-slope is 1. We then say that the MDPAO is *y-tie-less* if there exists a unique simple strategy that has the minimum y-slope (or equivalently the maximum x-slope) from all states. Finally, the MDPAO is *tie-less* if it is both x-tie-less and y-tie-less.

Notice that if an MDPAO is not tie-less then it can be made tie-less by an arbitrarily small perturbation of its transition probabilities. In contrast, a tie-less MDPAO can be perturbed by a sufficiently small amount and still be tie-less. So, being tie-less is a *robust* property of MDPAOs, whereas its opposite is not. In the following Lemmas 9-12 we assume that the MDPAO is x-tie-less and α is the simple strategy with minimum x-slope from all states.

By inspecting the linear program (3), we observe that, for all $q \in Q$, the slope of using α indefinitely cannot be improved (i.e., lowered) by starting with any other action. This is formalized by the following result.

▶ **Lemma 9.** *For all $\sigma \in \mathcal{S}$ and $q \in Q$, we have that $\mathit{slope}_{q,x}(\sigma\alpha^\omega) \geq \mathit{slope}_{q,x}(\alpha^\omega)$.*

The following lemma extends the previous one from simple strategies to arbitrary ω-strategies.

▶ **Lemma 10.** *For all $\pi \in \mathcal{S}^\omega$ and $q \in Q$, we have that $\mathit{slope}_{q,x}(\pi) \geq \mathit{slope}_{q,x}(\alpha^\omega)$.*

Proof. First, we prove the result for an ω-strategy that starts with an arbitrary finite prefix and then switches permanently to α, i.e., a strategy of the form $\pi = \tau\alpha^\omega$, for $\tau \in \mathcal{S}^*$. We proceed by induction on $|\tau|$. If $|\tau| = 0$, the result is trivial. Otherwise, $\tau = \tau_0\tau'$ and the induction hypothesis guarantees that $\mathit{slope}_{q,x}(\tau'\alpha^\omega) \geq \mathit{slope}_{q,x}(\alpha^\omega)$. Then,

$$\begin{aligned}
\mathit{slope}_{q,x}(\tau\alpha^\omega) &= p(q, \tau_0(q), x) + \sum_{q' \in Q} p(q, \tau_0(q), q') \mathit{slope}_{q',x}(\tau'\alpha^\omega) \\
&\geq p(q, \tau_0(q), x) + \sum_{q' \in Q} p(q, \tau_0(q), q') \mathit{slope}_{q',x}(\alpha^\omega) \quad \text{by ind. hyp.} \\
&= \mathit{slope}_{q,x}(\tau_0\alpha^\omega) \\
&\geq \mathit{slope}_{q,x}(\alpha^\omega) \qquad \text{by Lemma 9.}
\end{aligned}$$

Next, consider an arbitrary ω-strategy π. By contradiction, let $q \in Q$ be such that $\mathit{slope}_{q,x}(\alpha^\omega, q) - \mathit{slope}_{q,x}(\pi) = c > 0$. Let $p^0_{\max}$ be the maximum probability of continuation, i.e., $p^0_{\max} = \max_{q \in Q, \gamma \in \Gamma} \big(1 - p(q, \gamma, x) - p(q, \gamma, y)\big)$. Clearly $p^0_{\max} < 1$.

One can easily prove the following **claim:** Let $\pi^1, \pi^2 \in \mathcal{S}^\omega$ be two ω-strategies that coincide on the first k steps. Then, $|\mathit{slope}_{q,x}(\pi^1) - \mathit{slope}_{q,x}(\pi^2)| \leq (p^0_{\max})^k$.

Now, let $n > 0$ be such that $(p^0_{\max})^n < c$ (i.e., $n > \frac{\log c}{\log p^0_{\max}}$). It holds

$$\mathit{slope}_{q,x}(\pi_{\leq n}\alpha^\omega) \leq \mathit{slope}_{q,x}(\pi) + (p^0_{\max})^n < \mathit{slope}_{q,x}(\pi) + c = \mathit{slope}_{q,x}(\alpha^\omega).$$

This contradicts the above argument for finite prefixes and the thesis follows. ◀

The following lemma shows that when the initial value of register x grows, the x-slope of any optimal strategy approaches the minimum possible x-slope, i.e., the x-slope of the strategy α^ω. We denote by M an upper bound to $|f_\pi(q, 0)|$, for all π and q, as provided by Lemma 1.

▶ **Lemma 11.** *For all $q \in Q$, let $(\pi^{(k)})_{k\in\mathbb{N}}$ be a sequence of ω-strategies s.t. for all k, $f_{\pi^{(k)}}(q,k) = f(q,k)$ (i.e., $\pi^{(k)}$ is optimal from q and k). We have that $\lim_{k\to\infty} \mathit{slope}_{q,x}(\pi^{(k)}) = \mathit{slope}_{q,x}(\alpha^\omega)$.*

Proof. By Lemma 10, $\mathit{slope}_{q,x}(\pi^{(k)}) \geq \mathit{slope}_{q,x}(\alpha^\omega)$. We prove that for all $\epsilon > 0$ there exists $k > 0$ such that for all $m \geq k$ it holds $\mathit{slope}_{q,x}(\pi^{(m)}) < \mathit{slope}_{q,x}(\alpha^\omega) + \epsilon$. Let $k > \frac{2M}{\epsilon}$ and $m \geq k$. Assume by contradiction that $\mathit{slope}_{q,x}(\pi^{(m)}) \geq \mathit{slope}_{q,x}(\alpha^\omega) + \epsilon$. Then, we have:

$$\begin{aligned}
f_{\pi^{(m)}}(q,m) &= \mathit{slope}_{q,x}(\pi^{(m)})m + f_{\pi^{(m)}}(q,0)\\
&\geq \mathit{slope}_{q,x}(\pi^{(m)})m - M\\
&\geq \big(\mathit{slope}_{q,x}(\alpha^\omega) + \epsilon\big)m - M\\
&> \mathit{slope}_{q,x}(\alpha^\omega)m + \epsilon\frac{2M}{\epsilon} - M\\
&= \mathit{slope}_{q,x}(\alpha^\omega)m + M\\
&\geq f_{\alpha^\omega}(q,m).
\end{aligned}$$

This contradicts the fact that $\pi^{(m)}$ is optimal from q and m, and the thesis follows. ◀

For a state q, let $\mathcal{S}_q$ be the set of simple strategies σ such that $\sigma(q) \neq \alpha(q)$, and let

$$\begin{aligned}
c_q &= \min_{\sigma\in\mathcal{S}_q} \mathit{slope}_{q,x}(\sigma\alpha^\omega) - \mathit{slope}_{q,x}(\alpha^\omega)\\
&= \min_{\gamma\in\Gamma\setminus\{\alpha(q)\}} \Big(p(q,\gamma,x) + \sum_{q'\in Q} p(q,\gamma,q')\mathit{slope}_{q',x}(\alpha^\omega)\Big) - \mathit{slope}_{q,x}(\alpha^\omega).
\end{aligned}$$

If $\mathcal{M}$ is x-tie-less then $c_q > 0$ and the following lemma states that for large enough m the optimal strategy $\pi^{(m)}$ starts with the action $\alpha(q)$.

▶ **Lemma 12.** *For all $q \in Q$ and $m > \frac{2M}{c_q}$, let $\pi^{(m)}$ be an ω-strategy that is optimal from q and m. We have that $\pi_0^{(m)}(q) = \alpha(q)$.*

Proof. According to the proof of Lemma 11, for all $m > \frac{2M}{c_q}$ it holds $\mathit{slope}_{q,x}(\pi^{(m)}) < \mathit{slope}_{q,x}(\alpha^\omega) + c_q$. Assume by contradiction that there exists m such that $\pi_0^{(m)}(q) = \gamma \neq \alpha(q)$. Then, we have

$$\begin{aligned}
\mathit{slope}_{q,x}(\pi^{(m)}) &= p(q,\gamma,x) + \sum_{q'\in Q} p(q,\gamma,q')\mathit{slope}_{q',x}(\pi_{\geq 1}^{(m)})\\
&\geq p(q,\gamma,x) + \sum_{q'\in Q} p(q,\gamma,q')\mathit{slope}_{q',x}(\alpha^\omega) \quad \text{by Lemma 10}\\
&\geq \mathit{slope}_{q,x}(\alpha^\omega) + c_q \quad \text{by def. of } c_q.
\end{aligned}$$

This is a contradiction and the thesis follows. ◀

A series of symmetrical arguments shows that for $m \ll 0$, the optimal strategies from a configuration $(q,m,0)$ start with the simple strategy that has the *maximum* x-slope, or equivalently the minimum y-slope. By combining Lemma 12 and its symmetrical counterpart for negative m, we obtain the following, where a path-oblivious strategy σ is said to also be *counter-oblivious beyond* m if for all $m_1, m_2 \geq m$ and all $q \in Q$, it holds $\sigma(q,m_1,0) = \sigma(q,m_2,0)$ and $\sigma(q,-m_1,0) = \sigma(q,-m_2,0)$.

▶ **Theorem 13.** *For all tie-less 2-register MDPAOs, we can compute in polynomial time a number m such that there exists a strategy that is counter-oblivious beyond m and optimal.*

In the following subsection, Theorem 13 is used to prove computability of the minimum expected cost.

To conclude this subsection, we show that the property of Lemma 12 does not hold for general 2-register MDPAOs. The already mentioned MDPAO in Figure 4 is such that the optimal strategy is *not* eventually counter-oblivious. This example is inspired by the proof that approximating the minimum expected termination time of an OC-MDP is computationally hard [8]. Notice that said MDPAO is not x-tie-less, because both simple strategies $\sigma_1 = \{q_0 \to \lambda_1, q_1 \to \beta, q_2 \to \beta, q_3 \to \gamma'\}$ and $\sigma_2 = \{q_0 \to \lambda_2, q_1 \to \beta, q_2 \to \beta, q_3 \to \gamma'\}$ achieve the minimal x-slope from each state.

When starting from q_1 or q_2, with register values $x \ll 0$ and $y = 0$, the optimal strategy consists in staying in the $q_1 q_2$ loop for some time and then eventually exiting the loop by choosing action γ in q_1. This is because the x-slope of the strategy that stays in the loop is $1/2$, smaller than the x-slope of the strategy that exits the loop, which is 1. Later, when the value of x gets close to 0 and then positive, it becomes convenient to exit the loop, increase x by d and pay the current value of x. Moreover, depending on parameters d and ε, there is a specific value k for x at which it is maximally convenient to exit the loop. If the system is in q_1 when $x = k$, then the optimal strategy picks γ and exits the loop. If instead the system is in q_2, the strategy cannot immediately exit from the loop, so it must either exit the loop at $x = k - 1$ or at $x = k + 1$, whichever gives the least cost.

When starting in q_0, the optimal move is the one that ensures that the system will be in q_1 when x strikes the critical value k. Specifically, one can check that for sufficiently small ε and for $d \in (5, 5.5)$ [2] we obtain $k = 8$, so that the optimal move from $(q_0, -n, 0)$ is λ_1 when n is even and λ_2 when n is odd.

6.3 From tie-less MDPAOs to probabilistic 1-counter automata

Given a tie-less 2-register MDPAO $\mathcal{M}$ and a state q, we reduce its minimum expected cost problem from $(q, 0, 0)$ to a finite number of expected accumulated reward problems for probabilistic 1-counter automata (p1CA), each of which is decidable via an encoding in the existential Theory of Reals [7]. A p1CA is essentially an OC-MDP with a single action, also equivalent to a probabilistic pushdown automaton with a single stack symbol.

By Theorem 8, let $\mathcal{A} = (S, \Gamma, \Delta_0, \Delta_{>0})$ be the OC-MDP equivalent to $\mathcal{M}$. Let n be the threshold provided by Theorem 13 for $\mathcal{M}$. When looking for an optimal strategy for $\mathcal{M}$, we can limit our search to strategies that are path-oblivious and counter-oblivious beyond n. The set of all such strategies has cardinality $|\Gamma|^{|Q| \cdot n}$, which is doubly exponential in the size of the original MDPAO.

Given such a strategy σ, we build a single-action OC-MDP $\mathcal{A}_\sigma = (S', \{\gamma\}, \Delta'_0, \Delta'_{>0})$, whose expected accumulated reward from a distinguished state is equal to the expected cost of σ from $(q, 0, 0)$. The automaton $\mathcal{A}_\sigma$ is obtained from $\mathcal{A}$ as follows:

- By embedding the counter values $\{0, \ldots, n\}$ into the state, i.e., $S' = S \times \{0, \ldots, n\}$, with the intended meaning that each enlarged state $\langle s, k \rangle$, with $k < n$, is only visited with counter value 0 and corresponds to state s with counter value k in $\mathcal{A}$, whereas each enlarged state $\langle s, n \rangle$ with counter value l corresponds to state s and counter value $n + l$ in $\mathcal{A}$.

[2] A rational register update d can be easily simulated by probabilistically inducing the appropriate convex combination of the integer updates $\lfloor d \rfloor$ and $\lceil d \rceil$.

- By modifying the transitions according to the above encoding, while retaining only the actions chosen by the strategy σ. For instance, consider the enlarged state $\langle(q,x),k\rangle \in S'$, with $0 < k < n$, and let $\alpha = \sigma(q,k,0)$. Assume that the following transition occurs in $\mathcal{A}$: $((q,x),\alpha,p,d,r,(q',x)) \in \Delta_{>0}$. Then, the following occurs in $\mathcal{A}_\sigma$: $\big(\langle(q,x),k\rangle,\gamma,p,0,r,\langle(q',x),k+d\rangle\big) \in \Delta'_0$.
  Notice that under our assumptions, $d \in \{-1,0,1\}$ and so $k+d \in \{0,\dots,n\}$. On the other hand, $\Delta'_{>0}$ contains no transitions starting from $\langle(q,x),k\rangle$, because that state is only intended to be visited with counter value zero.

It is easy to prove by construction that the expected accumulated reward from $\langle(q,x),0\rangle$ in $\mathcal{A}_\sigma$ is equal to the expected cost of σ from $(q,0,0)$ in $\mathcal{M}$. Hence, we obtain the following.

▶ **Theorem 14.** *The minimum expected cost problem for tie-less 2-register MDPAOs with non-negative register updates is decidable in 2EXPTIME.*

7 Approximation algorithm for MDPs with alternative objectives

While computing the minimum expected cost even for simple models proves to be difficult, it is possible to compute the minimum expected cost in a general MDPAO $\mathcal{M}$ up to an additive error ϵ given $q \in Q$ and $\nu \in \mathcal{V}$. The idea is to compute the minimum expected cost of paths of length at most k and for large values of k, we can show that it is close to the actual minimum expected cost. Let $f^k(q,\nu)$ be this cost, i.e. $f^k(q,\nu) = \min_\sigma \sum_{\rho\in\Pi^k_\sigma(q,\nu)} Pr(\rho)f(\rho)$ where $\Pi^k_\sigma(q,\nu)$ is the set of paths of length at most k that start in (q,ν) and are consistent with σ.

Now, we show a result in Lemma 15 that bounds the difference between the actual cost $f(q,\nu)$ and $f^k(q,\nu)$ for any positive k. Let δ_M be the maximum change to a register in a step in the process and p_M be the maximum probability of continuation as defined in Lemma 1.

▶ **Lemma 15.** *Given a state $q \in Q$ and $\nu \in \mathcal{V}$ in an MDPAO $\mathcal{M}$,*

$$-p_M^k B(\max_{x\in X}|\nu(x)| + k\delta_M) \ \le\ f(q,\nu) - f^k(q,\nu) \ \le\ p_M^k B(\max_{x\in X}|\nu(x)| + k\delta_M) \triangleq z,$$

where B is a function of δ_M and p_M is described in Lemma 1.

Proof. To prove the lower bound, we split the cost $f(q,\nu)$ into two components f_1^k, which is the cost paid in the first k steps, and f_{k+1}^∞, the cost paid in the remaining steps. Note that $f_1^k \ge f^k(q,\nu)$, since $f^k(q,\nu)$ is the minimum expected cost of k steps. Further, after k steps, the probability that the process does not terminate is at most p_M^k and the register values are at least $-(\max_{x\in X}|\nu(x)| + k\delta_M)$. Thus, using Lemma 1, we can see that $f_{k+1}^\infty > -z$ which gives the required lower bound. To show the upper bound, we consider a strategy σ, where the first k actions minimize the expected cost of paths of length k, and the subsequent actions are arbitrary. By a similar analysis as above, $f_\sigma(q,\nu) \le f^k(q,\nu) + z$. Also, $f(q,\nu) \le f_\sigma(q,\nu)$ and hence, we have the required upper bound. ◀

Now, if we choose k such that $z = \epsilon$, the difference $|f(q,\nu) - f^k(q,\nu)|$ is bounded by ϵ and thus, k is $O(|\frac{\log\epsilon}{\log p_M}|)$. To compute $f^k(q,\nu)$, we transform $\mathcal{M}$ into an MDP $\mathcal{M}'$ with a node for each state and register valuation possible on paths of length at most k, starting from (q,ν). Note that there are at most $2k\delta_M + 1$ values possible for each register. Therefore, total number of states in $\mathcal{M}'$ is at most $O(|Q|(k\delta_M)^{|X|})$. We use dynamic programming to compute $f^k(q,\nu)$ by computing the minimum expected cost for $i+1$ steps using the cost for i steps. We can see that time to compute $f^k(q,\nu)$ is $(|\Gamma||Q|(|\frac{\log\epsilon}{\log p_M}|\delta_M)^{O(|X|)})$.

▶ **Theorem 16.** *In an MDPAO $\mathcal{M}$, the minimum expected cost $f(q, \nu)$ can be computed up to an additive error ϵ in $|\Gamma||Q|(|\frac{\log \epsilon}{\log p_M}|\delta_M)^{O(|X|)}$ time.*

8 Conclusions

We have introduced the model of Markov decision processes with alternative objectives to analyze situations where there are a number of alternative cost/reward objectives of which only a single one is actualized upon termination. We believe that the formalization and our results will find practical applications to planning scenarios with uncertain future rewards.

From a theoretical viewpoint, compared with the existing literature on MDPs, the optimization problem we have considered has an unusual structure worthy of further research: the underlying process is finite-state but the optimal choice depends on the infinite set of cumulative costs. Finding an exact solution for the general case remains an open problem, and as a next step, we would like to investigate whether it is always the case that two-register MDPAOs admit optimal strategies that are eventually periodic (see Section 6).

References

1 R. Alur, L. D'Antoni, J. Deshmukh, M. Raghothaman, and Y. Yuan. Regular functions and cost register automata. In *Proceedings of the 2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 13–22, 2013.

2 R. Alur and M. Raghothaman. Decision problems for additive regular functions. In *Automata, Languages, and Programming – 40th International Colloquium, ICALP, Part II*, pages 37–48, 2013.

3 R. Bellman. A Markovian decision process. *Journal of Mathematics and Mechanics*, 6:679–684, 1957.

4 D. P. Bertsekas and J. N. Tsitsiklis. An analysis of stochastic shortest path problems. *Math. Oper. Res.*, 16(3):580–595, August 1991.

5 T. Brázdil, V. Brožek, K. Etessami, and A. Kučera. Approximating the termination value of one-counter MDPs and stochastic games. In *International Colloquium on Automata, Languages, and Programming*, pages 332–343, 2011.

6 T. Brázdil, V. Brožek, K. Etessami, A. Kučera, and D. Wojtczak. One-counter Markov decision processes. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 863–874, 2010.

7 T. Brázdil, J. Esparza, S. Kiefer, and A. Kučera. Analyzing probabilistic pushdown automata. *Form. Methods Syst. Des.*, 43(2):124–163, October 2013.

8 T. Brázdil, A. Kučera, P. Novotný, and D. Wojtczak. Minimizing expected termination time in one-counter Markov decision processes. In *Automata, Languages, and Programming – 38th ICALP, Part II*, pages 141–152, 2012.

9 J. Esparza, A. Kučera, and R. Mayr. Quantitative analysis of probabilistic pushdown automata: expectations and variances. In *Proceedings of the 2005 20th Annual IEEE Symposium on Logic in Computer Science*, pages 117–126, 2005.

10 E. A. Feinberg and A. Shwartz. *Handbook of Markov decision processes: methods and applications*, volume 40. Springer Science & Business Media, 2012.

11 M. Kwiatkowska. Quantitative verification: Models, techniques and tools. In *Proc. ACM SIGSOFT Symp. on Foundations of Software Engineering*, pages 449–458, 2007.

12 D. M. Roijers, P. Vamplew, S. Whiteson, and R. Dazeley. A survey of multi-objective sequential decision-making. *J. Artif. Int. Res.*, 48(1):67–113, October 2013.