Journal of Computer and System Sciences 84 (2017) 144–170

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

Trading performance for stability in Markov decision processes

Tomáš Brázdil [a,*], Krishnendu Chatterjee [b], Vojtěch Forejt [c], Antonín Kučera [a]

[a] Faculty of Informatics, Masaryk University, Czechia
[b] IST Austria, Austria
[c] Department of Computer Science, University of Oxford, United Kingdom

ARTICLE INFO

Article history:
Received 9 February 2016
Received in revised form 16 July 2016
Accepted 21 September 2016
Available online 19 October 2016

Keywords:
Markov decision processes
Mean payoff
Stability
Stochastic systems
Controller synthesis

ABSTRACT

We study controller synthesis problems for finite-state Markov decision processes, where the objective is to optimize the expected mean-payoff performance and stability (also known as variability in the literature). We argue that the basic notion of expressing the stability using the statistical variance of the mean payoff is sometimes insufficient, and propose an alternative definition. We show that a strategy ensuring both the expected mean payoff and the variance below given bounds requires randomization and memory, under both the above definitions. We then show that the problem of finding such a strategy can be expressed as a set of constraints.

1. Introduction

Markov decision processes (MDPs) are a standard model for stochastic dynamic optimization. Roughly speaking, an MDP consists of a finite set of states, where in each state, one of the finitely many actions can be chosen by a controller. For every action, there is a fixed probability distribution over the states. The execution begins in some initial state where the controller selects an outgoing action, and the system evolves into another state according to the distribution associated with the chosen action. Then, another action is chosen by the controller, and so on. A *strategy* is a recipe for choosing actions. In general, a strategy may depend on the execution history (i.e., actions may be chosen differently when revisiting the same state) and the choice of actions can be randomized (i.e., the strategy specifies a probability distribution over the available actions). Fixing a strategy for the controller makes the behaviour of a given MDP fully probabilistic and determines the usual probability space over its *runs*, i.e., infinite sequences of states and actions.

A fundamental concept of performance and dependability analysis based on MDP models is *mean payoff*. Let us assume that every action is assigned some rational *reward*, which corresponds to some costs (or gains) caused by the action. The mean payoff of a given run is then defined as the long-run average reward per executed action, i.e., the limit of partial averages computed for longer and longer prefixes of a given run. For every strategy σ, the overall performance (or throughput) of the system controlled by σ then corresponds to the expected value of mean payoff, i.e., the *expected mean payoff*. It is well known (see, e.g., [23]) that optimal strategies for minimizing/maximizing the expected mean payoff are positional (i.e., deterministic and independent of execution history), and can be computed in polynomial time. However, the quality of ser-

* Corresponding author.
E-mail addresses: xbrazdil@fi.muni.cz (T. Brázdil), krish.chat@gmail.com (K. Chatterjee), vojfor@cs.ox.ac.uk (V. Forejt), tony@fi.muni.cz (A. Kučera).

http://dx.doi.org/10.1016/j.jcss.2016.09.009

vices provided by a given system often depends not only on its overall performance, but also on its *stability* (sometimes also called *variability*). For example, an optimal controller for a live video streaming system may achieve the expected throughput of approximately 2 MBits/sec. That is, if a user connects to the server many times, he gets 2 Mbits/sec connection *on average*. If an acceptable video quality requires at least 1.8 Mbits/sec, the user is also interested in the likelihood that he gets at least 1.8 Mbits/sec. That is, he requires a certain level of *overall stability* in service quality, which can be measured by the *variance* of the mean payoff, called *global variance* in this paper. The basic computational question is *"given rationals u and v, is there a strategy that achieves the expected mean payoff u (or better) and variance v (or better)?"*. Since the expected mean payoff can be "traded" for smaller global variance, we are also interested in approximating the associated *Pareto curve* consisting of all points (u, v) such that (1) there is a strategy achieving the expected mean payoff u and global variance v; and (2) no strategy can improve u or v without worsening the other parameter.

The global variance says how much the actual mean payoff of a run tends to deviate from the expected mean payoff. However, it does not say *anything* about the stability of individual runs. To see this, consider again the video streaming system example, where we now assume that although the connection is guaranteed to be fast on average, the amount of data delivered per second may change substantially along the executed run for example due to a faulty network infrastructure. For simplicity, let us suppose that performing one action in the underlying MDP model takes one second, and the reward assigned to a given action corresponds to the amount of transferred data. The above scenario can be modelled by saying that 6 Mbits are downloaded every third action, and 0 Mbits are downloaded in other time frames. Then the user gets 2 Mbits/sec connection almost surely, but since the individual runs are apparently "unstable", he may still see a lot of stuttering in the video stream. As an appropriate measure for the stability of individual runs, we propose *local variance*, which is defined as the long-run average of $(r_i(\omega) - mp(\omega))^2$, where $r_i(\omega)$ is the reward of the i-th action executed in a run ω and $mp(\omega)$ is the mean payoff of ω. Hence, local variance says how much the rewards of the actions executed along a given run deviate from the mean payoff of the run on average. For example, if the mean payoff of a run is 2 Mbits/sec and all of the executed actions deliver 2 Mbits, then the run is "absolutely smooth" and its local variance is zero. The level of "local stability" of the whole system (under a given strategy) then corresponds to the *expected local variance*. The basic algorithmic problem for local variance is similar to the one for global variance, i.e., *"given rationals u and v, is there a strategy that achieves the expected mean payoff u (or better) and the expected local variance v (or better)?"*. We are also interested in the underlying Pareto curve.

Observe that the global variance and the expected local variance capture different and to a large extent *independent* forms of systems' (in)stability. Even if the global variance is small, the expected local variance may be large, and vice versa.

1.1. The results

Our results are as follows:

1. *(Global variance).* The global variance problem was considered before in [26], but only under the restriction of memoryless strategies. We first show that in general, randomized memoryless strategies are not sufficient for Pareto optimal points for global variance (Example 1). We then establish that 2-memory strategies are sufficient, and that the problem of existence of a strategy can be reduced to the problem of finding a solution of a set of non-linear constraints. We show that the basic algorithmic problem for global variance is in PSPACE, and the approximate version can be solved in pseudo-polynomial time.
2. *(Local variance).* The local variance problem comes with new conceptual challenges. For example, for unichain MDPs, deterministic memoryless strategies are sufficient for global variance, whereas we show (Example 2) that even for unichain MDPs both randomization and memory are required for local variance. We establish that 3-memory strategies are sufficient for Pareto optimality for local variance, and again give a set of non-linear constraints describing the existence of a strategy. We show that the basic algorithmic problem (and hence also the approximate version) is in NP.
3. *(Zero variance).* Finally, we consider the problem where the variance is optimized to zero (as opposed to a given non-negative number in the general case). In this case, we present polynomial-time algorithms to compute the optimal mean-payoff that can be ensured with zero variance (if zero variance can be ensured) for both the cases. The polynomial-time algorithms for zero variance for mean-payoff objectives is in sharp contrast to the NP-hardness for cumulative reward MDPs [19].

To prove the above results, one has to overcome various obstacles. For example, although at multiple places we build on the techniques of [13] and [2] which allow us to deal with maximal end components (sometimes called strongly communicating sets) of an MDP separately, we often need to extend these techniques. Unlike the works [13] and [2] which study multiple "independent" objectives, in the case of the global variance any change of value in the expected mean payoff implies a change of value of the variance. Also, since we do not impose any restrictions on the structure of the strategies, we cannot even assume that the limits defining the mean payoff and the respective variances exist; this becomes most apparent in the case of the local variance, where we need to rely on delicate techniques of selecting runs from which the limits can be extracted. Another complication is that while most of the work on multi-objective controller synthesis for MDPs deals with linear objective functions, our objective functions are inherently quadratic due to the definition of variance. Finally,

Table 1
Summary of the results, where LB and UB denotes lower- and upper-bound, respectively.

	Memory size	Non-linear constr. in encoding	Complexity	Approx. complexity
Global	2-memory LB: Example 1 UB: Theorem 1	1 quadratic, 1 cubic (Theorem 1)	PSPACE (Corollary 1)	Pseudo-poly. (Corollary 2)
Local	LB: 2-memory (Example 2) UB: 3-memory (Corollary 5)	Several cubic (Theorem 2)	NP (Corollary 4)	NP

mean-payoff objectives with global variance was considered in [26], but only for the special class of memoryless strategies. The solution for general strategies is significantly more involved for the following reasons: first, for general strategies it is not clear that limits defining the mean-payoff objectives exist; second, upper bounds on memory for general strategies is also not clear; and finally, encoding strategies with memory as optimization problem is far more non-trivial than memoryless strategies.

The summary of our results is presented in Table 1. A simple consequence of our results is that the Pareto curves can be approximated in pseudo-polynomial time in the case of the global variance, and in exponential time for the local variance.

1.2. Related work

Studying the trade-off between multiple objectives in an MDP has attracted significant attention in the recent years (see [1] for an overview). In the formal verification area, MDPs with multiple mean-payoff objectives [2], discounted objectives [9], cumulative reward objectives [17], and multiple ω-regular objectives [13] have been studied. As for the stability of a system, the variance-penalized mean-payoff problem (where the mean payoff is penalized by a constant times the variance) under memoryless (stationary) strategies was studied in [14]. The mean-payoff variance trade-off problem for unichain MDPs was considered in [10], where a solution using quadratic programming was designed; under memoryless strategies the problem was considered in [26]. All the above works for mean-payoff variance trade-off consider the global variance, and are restricted to memoryless strategies. The problem for general strategies and global variance was not solved before. Although restrictions to unichains or memoryless strategies are feasible in some areas, many systems modelled as MDPs might require more general approach. For example, a decision of a strategy to shut the system down might make it impossible to return the running state again, yielding a non-unichain MDP. Similarly, it is natural to synthesise strategies that change their decisions over time.

As regards other types of objectives, no work considers the local variance problem. The (global) variance problem for *discounted* reward MDPs was studied in [25]. The trade-off of expected value and variance of *cumulative* reward in MDPs was studied in [19], showing NP-hardness already for the case where the goal is to achieve zero variance.

A preliminary version of this paper was presented as a conference publication [3]. The present version contains complete proofs, and also presents a direct encoding for solving the local variance problem. The conference publication also contained a notion of *hybrid variance*, which for focused presentation is omitted from this article.

2. Preliminaries

We use $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$ to denote the sets of positive integers, integers, rational numbers, and real numbers, respectively. Given a set X of elements and $x \in X$, we define $I_x : X \to \{0,1\}$ to be the *indicator function* for x, i.e., the function satisfying $I_x(x') = 1$ if $x = x'$, and $I_x(x') = 0$ otherwise.

We assume familiarity with basic notions of probability theory, e.g., *probability space*, *random variable*, or *expected value*. As usual, a *probability distribution* over a finite or countable set X is a function $f : X \to [0,1]$ such that $\sum_{x \in X} f(x) = 1$. We call f *Dirac* if $f(x) = 1$ for some $x \in X$. The set of all distributions over X is denoted by $dist(X)$.

For our purposes, a *Markov chain* is a triple $M = (L, \to, \mu)$ where L is a finite or countably infinite set of *locations*, $\to \subseteq L \times (0,1] \times L$ is a *transition relation* such that for each fixed $\ell \in L$, $\sum_{\ell \xrightarrow{x} \ell'} x = 1$, and μ is the *initial probability distribution* on L. A *run* in M is an infinite sequence $\omega = \ell_1 \ell_2 \ldots$ of locations such that $\ell_i \xrightarrow{x_i} \ell_{i+1}$ for every $i \in \mathbb{N}$. A *finite path* in M is a finite prefix of a run. Each finite path w in M determines the set $Cone(w)$ consisting of all runs that start with w. To M we associate the probability space $(Runs_M, \mathcal{F}, \mathbb{P})$, where $Runs_M$ is the set of all runs in M, $\mathcal{F}$ is the σ-field generated by all $Cone(w)$ for finite paths w, and $\mathbb{P}$ is the unique probability measure such that $\mathbb{P}(Cone(\ell_1, \ldots, \ell_k)) = \mu(\ell_1) \cdot \prod_{i=1}^{k-1} x_i$, where $\ell_i \xrightarrow{x_i} \ell_{i+1}$ for all $1 \leq i < k$ (the empty product is equal to 1).

2.1. Markov decision processes

A *Markov decision process* (MDP) is a tuple $G = (S, A, Act, \delta)$ where S is a *finite* set of states, A is a *finite* set of actions, $Act : S \to 2^A \setminus \{\emptyset\}$ is an action enabledness function that assigns to each state s the set $Act(s)$ of actions enabled at s, and

$\delta : S \times A \to dist(S)$ is a probabilistic transition function that given a state s and an action $a \in Act(s)$ enabled at s gives a probability distribution over the successor states. For notational simplicity, we assume that every action is enabled in exactly one state, and we denote this state $Src(a)$; such assumption can be imposed without loss of generality, since any MDP can be transformed to this form in polynomial time. Thus, henceforth we will assume that $\delta : A \to dist(S)$.

A *run* in G is an infinite alternating sequence of states and actions $\omega = s_1 a_1 s_2 a_2 \dots$ such that for all $i \geq 1$, $Src(a_i) = s_i$ and $\delta(a_i)(s_{i+1}) > 0$. We denote by $Runs_G$ the set of all runs in G. A *finite path* of length k in G is a finite prefix $w = s_1 a_1 \dots a_{k-1} s_k$ of a run, and we use $last(w) = s_k$ for the last state of w. Given a run $\omega \in Runs_G$, we denote by $A_i(\omega)$ the i-th action a_i of ω. Given a set F of states (resp. actions), we define $Reach(F)$ to be the set of all runs that contain a state (resp. action) from F.

A pair (T, B) with $\emptyset \neq T \subseteq S$ and $B \subseteq \bigcup_{t \in T} Act(t)$ is an *end component* (or strongly communicating set) of G if (1) for all $a \in B$, if $\delta(a)(s') > 0$ then $s' \in T$; and (2) for all $s, t \in T$ there is a finite path $w = s_1 a_1 \dots a_{k-1} s_k$ such that $s_1 = s$, $s_k = t$, and all states and actions that appear in w belong to T and B, respectively. An end component (T, B) is a *maximal end component (MEC)* if it is maximal wrt. pointwise subset ordering. The set of all MECs of G is denoted by $\mathcal{M}(G)$. Given an end component $C = (T, B)$, we sometimes abuse notation by considering C as the disjoint union of T and B (e.g., we write $S \cap C$ to denote the set T). For a given $C \in \mathcal{M}(G)$, we use R_C to denote the set of all runs $\omega = s_1 a_1 s_2 a_2 \dots$ that eventually *stay* in C, i.e., there is $k \in \mathbb{N}$ such that for all $k' \geq k$ we have that $s_{k'}, a_{k'} \in C$.

An MDP is *strongly connected* if all its states form a single (maximal) end component. A strongly connected MDP is a *unichain* if for all end components (T, B) we have $T = S$.

Sometimes we also unify a MEC C with a MDP obtained from G by restricting the set of states and actions to those in C, and by restricting Act and δ accordingly.

2.2. Strategies and plays

Intuitively, a *strategy* (sometimes called *policy*) in an MDP G is a "recipe" to choose actions. Usually, a strategy is formally defined as a function $\sigma : (SA)^* S \to dist(A)$ that given a finite path w, representing the execution history, gives a probability distribution over the actions enabled in $last(w)$. In this paper we adopt a definition which is equivalent to the standard one, but more convenient for our purpose. Let M be a finite or countably infinite set of *memory elements*. A *strategy* is a triple $\sigma = (\mathsf{upd}_\sigma, \mathsf{next}_\sigma, \mathsf{init}_\sigma)$, where $\mathsf{upd}_\sigma : A \times S \times \mathsf{M} \to dist(\mathsf{M})$ and $\mathsf{next}_\sigma : S \times \mathsf{M} \to dist(A)$ are *memory update* and *next move* functions, respectively, and init_σ is an initial distribution on memory elements. We require that for all $(s, m) \in S \times \mathsf{M}$, the distribution $\mathsf{next}_\sigma(s, m)$ assigns a positive value only to actions enabled at s. The set of all strategies is denoted by Σ (the underlying MDP G will be always clear from the context).

A *play* of G determined by an *initial state* $s \in S$ and a strategy σ is a Markov chain G_s^σ (or G^σ if s is clear from the context) where the set of locations is $S \times \mathsf{M} \times A$, the initial distribution μ is positive only on (some) elements of $\{s\} \times \mathsf{M} \times A$ where $\mu(s, m, a) = \mathsf{init}_\sigma(m) \cdot \mathsf{next}_\sigma(s, m)(a)$, and $(t, m, a) \xrightarrow{x} (t', m', a')$ iff $x = \delta(a)(t') \cdot \mathsf{upd}_\sigma(a, t', m)(m') \cdot \mathsf{next}_\sigma(t', m')(a') > 0$. Hence, G_s^σ starts in a location chosen randomly according to init_σ and next_σ. In a current location (t, m, a), the next action to be performed is a, hence the probability of entering t' is $\delta(a)(t')$. The probability of updating the memory to m' is $\mathsf{upd}_\sigma(a, t', m)(m')$, and the probability of selecting a' as the next action is $\mathsf{next}_\sigma(t', m')(a')$. Since these choices are independent (in the probability theory sense), we obtain the product above. We use $\mathbb{P}_s^\sigma$ for the probability measure induced by G_s^σ.

Note that every run in G_s^σ determines a unique run in G. Hence, every notion originally defined for the runs in G can also be used for the runs in G_s^σ, and we use this fact implicitly at many places in this paper. For example, we use the symbol R_C to denote the set of all runs in G_s^σ that eventually stay in C, certain functions originally defined over $Runs_G$ are interpreted as random variables over the runs in G_s^σ, etc.

2.3. Strategy types

In general, a strategy may use infinite memory, and both upd_σ and next_σ may use randomization. A strategy σ is *deterministic* if init_σ is Dirac and both the memory update and the next move functions give a Dirac distribution for every argument. A *randomized* strategy is a strategy which is not necessarily deterministic. We also classify the strategies according to the size of memory they use. Important subclasses are *memoryless* strategies, in which M is a singleton, *n-memory* strategies, in which M has exactly n elements, and *finite-memory* strategies, in which M is finite.

For a finite-memory strategy σ, a *bottom strongly connected component* (BSCC) of G_s^σ is a subset of locations $W \subseteq S \times \mathsf{M} \times A$ such that for all $\ell_1 \in W$ and $\ell_2 \in S \times \mathsf{M} \times A$ we have that $\ell_2 \in W$ if and only if ℓ_2 is reachable from ℓ_1. We use $\mathcal{B}(G_s^\sigma)$ for the set of all BSCCs of G_s^σ. Every BSCC W determines a unique end component $(\{s \mid (s, m, a) \in W\}, \{a \mid (s, m, a) \in W\})$, and we sometimes do not distinguish between W and its associated end component.

Let ν be a memoryless randomized strategy on a MEC C and let K be a BSCC of C^ν. We say that a strategy μ_K is *induced* by K if

1. $\mu_K(s)(a) = \nu(s)(a)$ for all $s \in K \cap S$ and $a \in K \cap A$
2. in all $s \in S \setminus (K \cap S)$ the strategy μ_K corresponds to a memoryless deterministic strategy which reaches a state of K with probability one

Note that the above definition of induced strategy is independent of the strategy ν: it only depends on the BSCC K, and on the way K is reached.

2.4. Global and local variance

Let $G = (S, A, Act, \delta)$ be an MDP. A *reward function* is a function $r : A \to \mathbb{Q}$, and we define the *mean payoff* of a run $\omega \in Runs_G$ with respect to r by

$$mp(\omega) = \limsup_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} r(A_i(\omega)) .$$

The expected value and variance of mp in G_s^σ are denoted by $\mathbb{E}_s^\sigma[mp]$ and $\mathbb{V}_s^\sigma[mp]$, respectively (recall that $\mathbb{V}_s^\sigma[mp] = \mathbb{E}_s^\sigma\big[(mp - \mathbb{E}_s^\sigma[mp])^2\big] = \mathbb{E}_s^\sigma\big[mp^2\big] - \mathbb{E}_s^\sigma[mp]^2$). Intuitively, $\mathbb{E}_s^\sigma[mp]$ corresponds to the "overall performance" of G_s^σ, and $\mathbb{V}_s^\sigma[mp]$ is a measure of "global stability" of G_s^σ indicating how much the mean payoff of runs in G_s^σ tends to deviate from $\mathbb{E}_s^\sigma[mp]$ (see Section 1). In the rest of this paper, we refer to $\mathbb{V}_s^\sigma[mp]$ as *global variance*.

The stability of a given run $\omega \in Runs_G$ (see Section 1) is measured by its *local variance* defined as follows:

$$lv(\omega) = \limsup_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \big(r(A_i(\omega)) - mp(\omega)\big)^2$$

Note that $lv(\omega)$ is not really a "variance" in the usual sense of probability theory.[1] We call the function $lv(\omega)$ "local variance" because we find this name suggestive; $lv(\omega)$ is the long-run average square of the distance from $mp(\omega)$. The expected value of lv in G_s^σ is denoted by $\mathbb{E}_s^\sigma[lv]$.

If we need to stress which reward function is being considered, we will write it in the superscript, for example mp^r; in particular, by mp^{1_a} we will denote the mean payoff with respect to the reward function that returns 1 for a and 0 for other actions.

From now on, we restrict ourselves to reward functions which only assign non-negative rewards. This is w.l.o.g., since for a reward function r we can define r' given by $r'(a) = r(a) + \min_{a' \in A} r(a')$ for all a, and under any strategy σ the expected mean payoff wrt. r' is by $\min_{a' \in A} r(a')$ greater than the one wrt. r, and the variances remain unchanged.

2.5. Pareto optimality

A *Pareto curve* for an initial state s wrt. global variance is the set of points (u, v) such that for all $\varepsilon > 0$ there is a strategy σ with $(\mathbb{E}_s^\sigma[mp], \mathbb{V}_s^\sigma[mp]) \leq (u, v) + (\varepsilon, \varepsilon)$, and there is no strategy ζ with $(\mathbb{E}_s^\zeta[mp], \mathbb{V}_s^\zeta[mp]) < (u, v)$, where $\leq$ is the standard component-wise ordering. A point (u, v) is a *Pareto point* if it lies on the Pareto curve. A strategy σ is *Pareto optimal* in s wrt. global variance if $(\mathbb{E}_s^\sigma[mp], \mathbb{V}_s^\sigma[mp])$ is a Pareto point. We say that σ *achieves* (u, v) whenever $(\mathbb{E}_s^\sigma[mp], \mathbb{V}_s^\sigma[mp]) \leq (u, v)$, and (u, v) is then called *achievable*.

Similarly, we define a Pareto curve and Pareto optimality of σ wrt. the local variance by replacing $\mathbb{V}_s^\alpha[mp]$ with $\mathbb{E}_s^\alpha[lv]$.

2.6. Frequency functions

Let C be a MEC. We say that $f : C \cap A \to [0, 1]$ is a *frequency function on* C if

- $\sum_{a \in C \cap A} f(a) = 1$
- $\sum_{a \in C \cap A} f(a) \cdot \delta(a)(s) = \sum_{a \in C \cap Act(s)} f(a)$ for every $s \in C \cap S$

Define $mp[f] := \sum_{a \in C \cap A} f(a) \cdot r(a)$ and $lv[f] := \sum_{a \in C \cap A} f(a) \cdot (r(a) - mp[f])^2$.

2.7. The studied problems

In this paper, we study the following basic problems connected to the two stability measures introduced above (below V_s^σ is either $\mathbb{V}_s^\sigma[mp]$ or $\mathbb{E}_s^\sigma[lv]$):

- *Pareto optimal strategies and their memory*. Do Pareto optimal strategies exist for all points on the Pareto curve? Do Pareto optimal strategies require memory and randomization in general? Do strategies achieving non-Pareto points require memory and randomization in general?

[1] By investing some effort, one could perhaps find a random variable X such that $lv(\omega)$ is the variance of X, but this question is not really relevant—we only use lv as a *random variable which measures the level of local stability of runs*. One could perhaps study the variance of lv, but this is beyond the scope of this paper.

- *Deciding strategy existence*. For a given MDP G, an initial state s, a rational reward function r, and a point $(u, v) \in \mathbb{Q}^2$, we ask whether there exists a strategy σ such that $(\mathbb{E}_s^\sigma[mp], V_s^\sigma) \leq (u, v)$.
- *Approximation of strategy existence*. For a given MDP G, an initial state s, a rational reward function r, a number ε and a point $(u, v) \in \mathbb{Q}^2$, we want to get an algorithm which (a) outputs "yes" if there is a strategy σ such that $(\mathbb{E}_s^\sigma[mp], V_s^\sigma) \leq (u - \varepsilon, v - \varepsilon)$; (b) outputs "no" if there is no strategy such that $(\mathbb{E}_s^\sigma[mp], V_s^\sigma) \leq (u, v)$.
- *Strategy synthesis.* For a given MDP G, an initial state s, a rational reward function r, and a point $(u, v) \in \mathbb{Q}^2$, if there exists a strategy σ such that $(\mathbb{E}_s^\sigma[mp], V_s^\sigma) \leq (u, v)$, we wish to *compute* such strategy. Note that it is not *a priori* clear that σ is finitely representable, and hence we also need to answer the question what *type* of strategies is needed to achieve Pareto optimal points.

Remark 1. If the approximation of strategy existence problem is decidable, we design the following algorithm to approximate the Pareto curve up to an arbitrarily small given $\varepsilon > 0$. We compute a finite set of points $P \subseteq \mathbb{Q}^2$ such that (1) for every Pareto point (u, v) there is $(u', v') \in P$ with $(|u - u'|, |v - v'|) \leq (\varepsilon, \varepsilon)$, and (2) for every $(u', v') \in P$ there is a Pareto point (u, v) such that $(|u - u'|, |v - v'|) \leq (\varepsilon, \varepsilon)$. Let $R = \max_{a \in A} |r(a)|$. Note that $|\mathbb{E}_s^\sigma[mp]| \leq R$ and $V_s^\sigma \leq R^2$ for an arbitrary strategy σ. Hence, the set P is computable by a naive algorithm which decides the approximation of strategy existence for $\mathcal{O}(|R|^3/\varepsilon^2)$ points in the corresponding ε-grid and puts $\mathcal{O}(|R|^2/\varepsilon)$ points into P. The question whether the Pareto curves can be approximated more efficiently by sophisticated methods based on deeper analysis of their properties is left for future work.

In the rest of this paper, unless specified otherwise, we suppose we work with a fixed MDP $G = (S, A, Act, \delta)$, an initial state s_{in}, and a reward function $r : A \to \mathbb{Q}$.

2.8. Basic properties of MECs and strategies

At several places of this paper we will proceed by analysing MECs separately and then devising results by combining the sub-results for each respective MEC. For this purpose, we will use several technical results, which we state in the three lemmas below.

Lemma 1 *([11]). Almost all runs eventually end in a MEC, i.e., for all σ and s we have $\mathbb{P}_s^\sigma\left[\bigcup_{C \in \mathcal{M}(G)} R_C\right] = 1$.*

Let G be an MDP, and let G' be an auxiliary MDP obtained from G by adding a state d_s for every state $s \in S$, an action a_s that leads to d_s from s, and a self-loop on d_s. The following lemma is a direct adaptation of [2, Lemma 3].

Lemma 2 *([2]). Let σ be a strategy for G. Then there is a memoryless strategy $\bar{\sigma}$ in G' such that $\mathbb{P}_{s_{in}}^\sigma[R_C] = \mathbb{P}_{s_{in}}^{\bar{\sigma}}[Reach(\{d_s \mid s \in C\})]$ for all MECs C.*

The following lemma will allow us to combine a "transient" strategy for reaching MECs with "recurrent" strategies that describe the behaviour in MECs.

Lemma 3. *Let Δ be a distribution on $\{1, \ldots, n\} \times \mathcal{M}(G)$. Further, let σ and $\pi_1, \ldots, \pi_n$ be strategies such that $\mathbb{E}_t^{\pi_i}[mp] = \mathbb{E}_{t'}^{\pi_i}[mp]$ and $\mathbb{E}_t^{\pi_i}[lv] = \mathbb{E}_{t'}^{\pi_i}[lv]$ for all $1 \leq i \leq n$, all $C \in \mathcal{M}(G)$, and all $t, t' \in C \cap S$. Then there is a strategy σ' such that*

$$\mathbb{E}_{s_{in}}^{\sigma'}[X] = \sum_{C \in \mathcal{M}(G)} \mathbb{E}_{s_{in}}^{\sigma}[R_C] \cdot \sum_{i=1}^{n} \Delta(i)(C) \cdot \mathbb{E}_t^{\pi_i}[X]$$

for $X \in \{mp, lv\}$, where t is any state of C (note that the corresponding values are equal for all states of C); and if for all $1 \leq i \leq n$ almost all runs under π_i have the same mean payoff $\mathbb{E}_t^{\pi_i}[mp]$, then also

$$\mathbb{V}_{s_{in}}^{\sigma'}[mp] = \Big(\sum_{C \in \mathcal{M}(G)} \mathbb{E}_{s_{in}}^{\sigma}[R_C] \cdot \sum_{i=1}^{n} \Delta(i)(C) \cdot \mathbb{E}_t^{\pi_i}[mp]^2 \Big) - \Big(\sum_{C \in \mathcal{M}(G)} \mathbb{E}_{s_{in}}^{\sigma}[R_C] \cdot \sum_{i=1}^{n} \Delta(i)(C) \cdot \mathbb{E}_t^{\pi_i}[mp] \Big)^2$$

where t is any state of C. Moreover, σ' can be constructed so that its memory is the sum of memory sizes of π_i for all $1 \leq i \leq n$ plus one.

Proof. We first apply Lemma 2 to σ and obtain a memoryless strategy $\hat{\sigma}$ for G' above such that $\mathbb{P}_{s_{in}}^\sigma[R_C] = \mathbb{P}_{s_{in}}^{\hat{\sigma}}[Reach(\{d_s \mid s \in C\})]$ for all MECs C. The strategy σ' plays according to $\hat{\sigma}$ until just before reaching d_s for some s

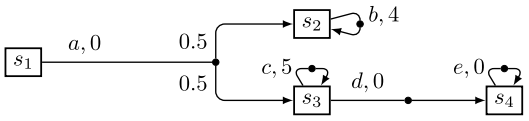

Fig. 1. An MDP witnessing the need for memory and randomization in Pareto optimal strategies for global variance.

contained in a MEC C. Instead of transitioning to d_s, the strategy σ' starts playing as π_i, with probability $\Delta(i, C)$ for the MEC C containing s. The required properties follow easily by the law of total expectation and the definition of variance.

Formally, the set of memory elements of σ' is the union of the sets of memory elements of all $\hat{\sigma}, \pi_1, \ldots, \pi_n$. We use m_{in} for the single memory element of $\hat{\sigma}$, and M_i for the set of memory elements of π_i. The strategy σ' is defined by letting $\mathsf{init}_{\sigma'} = \mathsf{init}_{\hat{\sigma}} = m_{in}$, and for all for all $s \in S$ and $a \in A$ and memory elements m:

$$\mathsf{next}_{\sigma'}(s, m)(a) = \begin{cases} \mathsf{next}_{\hat{\sigma}}(s, m)(a) + \mathsf{next}_{\hat{\sigma}}(s, m)(a_s) \cdot \sum_{i=1}^{n} \Delta(i, C) \cdot \pi_i(s)(a) & \text{if } m = m_{in} \\ \mathsf{next}_{\pi_i}(s, m)(a) & \text{if } m \in \mathsf{M}_i \text{ for some } 1 \leq i \leq n \end{cases}$$

and

$$\mathsf{upd}_{\sigma'}(a, s, m)(m') = \begin{cases} \mathsf{upd}_{\hat{\sigma}}(a, s, m)(m') \cdot \frac{\mathsf{next}_{\hat{\sigma}}(s,m)(a)}{\mathsf{next}_{\sigma'}(s,m)(a)} & \text{if } m = m' = m_{in} \\ \mathsf{upd}_{\pi_i}(a, s, m)(m') \cdot \frac{\mathsf{next}_{\hat{\sigma}}(s,m)(a_s) \cdot \Delta(i,C) \cdot \pi_i(s)(a)}{\mathsf{next}_{\sigma'}(s,m)(a)} & \text{if } m = m_{in} \text{ and } m' \in \mathsf{M}_i \\ \mathsf{upd}_{\pi_i}(a, s, m)(m') & \text{if } m \in \mathsf{M}_i \text{ for some } 1 \leq i \leq n \end{cases}$$

where $\pi_i(s)(a) = \sum_{m' \in \mathsf{M}_i} \mathsf{init}_{\pi_i}(m') \cdot \mathsf{next}_{\pi_i}(s, m')(a)$ and C is the MEC containing s. □

3. Global variance

In this section we study the global variance problem, which was considered in [26] for memoryless strategies.

Basic open questions. Given the previous results of [26] the following basic questions remained open for the global variance problem:

1. Are memoryless strategies sufficient, or are strategies with memory more powerful?
2. If memoryless strategies are not sufficient, then can an upper bound on the memory of strategies be established for sufficiency?
3. Is the problem decidable for general strategies?

We start by proving that both memory and randomization are needed even for achieving non-Pareto points; this implies that memory and randomization is needed even to approximate the value of Pareto points. Then we show that 2-memory strategies are sufficient, which gives a tight bound. We will also establish decidability in PSPACE. Thus our results answer all the basic open questions.

Example 1. Consider the MDP of Fig. 1, with the rewards of actions as given next to the action names. Observe that the point $(4, 2)$ is achievable by a strategy σ which selects c with probability $\frac{4}{5}$ and d with probability $\frac{1}{5}$ upon the *first* visit to s_3; in every subsequent visit to s_3, the strategy σ selects c with probability 1. Hence, σ is a 2-memory randomized strategy which stays in MEC $C = (\{s_3\}, \{c\})$ with probability

$$\mathbb{P}^{\sigma}_{s_1}[R_C] = \mathbb{P}^{\sigma}_{s_1}[Cone(s_1 a s_3)] = \frac{1}{2} \cdot \frac{4}{5} = \frac{2}{5}.$$

Clearly,

$$\begin{aligned} \mathbb{E}^{\sigma}_{s_1}[mp] &= \mathbb{P}^{\sigma}_{s_1}[Cone(s_1 a s_2)] \cdot \mathbb{E}^{\sigma}_{s_1}[mp \mid Cone(s_1 a s_2)] \\ &\quad + \mathbb{P}^{\sigma}_{s_1}[Cone(s_1 a s_3 c s_3)] \cdot \mathbb{E}^{\sigma}_{s_1}[mp \mid Cone(s_1 a s_3 c s_3)] \\ &\quad + \mathbb{P}^{\sigma}_{s_1}[Cone(s_1 a s_3 d s_4)] \cdot \mathbb{E}^{\sigma}_{s_1}[mp \mid Cone(s_1 a s_3 d s_4)] \\ &= \frac{1}{2} \cdot 4 + \frac{1}{2} \cdot \frac{4}{5} \cdot 5 + \frac{1}{2} \cdot \frac{1}{5} \cdot 0 = 4 \end{aligned}$$

$$I_{s_{in}}(t) + \sum_{a \in A} y_a \cdot \delta(a)(t) = \sum_{a \in Act(t)} y_a + y_t \qquad \text{for all } t \in S \tag{1}$$

$$\sum_{C \in \mathcal{M}(G)} \sum_{t \in S \cap C} y_t = 1 \tag{2}$$

$$\alpha_C \leq x_C \leq \beta_C \qquad \text{for all } C \in \mathcal{M}(G) \tag{3}$$

$$u \geq \sum_{C \in \mathcal{M}(G)} x_C \cdot \sum_{t \in S \cap C} y_t \tag{4}$$

$$v \geq \Big(\sum_{C \in \mathcal{M}(G)} x_C^2 \cdot \sum_{t \in S \cap C} y_t \Big) - \Big(\sum_{C \in \mathcal{M}(G)} x_C \cdot \sum_{t \in S \cap C} y_t \Big)^2 \tag{5}$$

Fig. 2. The system L_{glob}.

and similarly

$$\mathbb{V}^{\sigma}_{s_1}[mp] = \mathbb{E}^{\sigma}_{s_1}\Big[mp^2\Big] - \mathbb{E}^{\sigma}_{s_1}[mp]^2 = \frac{1}{2} \cdot 4^2 + \frac{1}{2} \cdot \frac{4}{5} \cdot 5^2 + \frac{1}{2} \cdot \frac{1}{5} \cdot 0^2 - 4^2 = 2$$

Further, note that every strategy $\bar{\sigma}$ which stays in C with probability x satisfies $\mathbb{E}^{\bar{\sigma}}_{s_1}[mp] = \frac{1}{2} \cdot 4 + x \cdot 5$ and $\mathbb{V}^{\bar{\sigma}}_{s_1}[mp] = \frac{1}{2} \cdot 4^2 + x \cdot 5^2 - (2 + x \cdot 5)^2$. For $x > \frac{2}{5}$ we get $\mathbb{E}^{\bar{\sigma}}_{s_1}[mp] > 4$, and for $x < \frac{2}{5}$ we get $\mathbb{V}^{\bar{\sigma}}_{s_1}[mp] > 2$, so $(4, 2)$ is indeed a Pareto point. Every deterministic (resp. memoryless) strategy can end in C with probability either $\frac{1}{2}$ or 0, giving $\mathbb{E}^{\bar{\sigma}}_{s_1}[mp] = \frac{9}{2}$ or $\mathbb{V}^{\bar{\sigma}}_{s_1}[mp] = 4$. So, both memory and randomization are needed to achieve the Pareto point $(4, 2)$ or a non-Pareto point $(4.1, 2.1)$.

Interestingly, if the MDP is strongly connected, memoryless deterministic strategies always suffice, because in this case a memoryless strategy that minimizes the expected mean payoff immediately gets zero variance. This is in contrast with the local variance, where we will show that memory and randomization is required in general already for unichain MDPs. For the general case of global variance, the sufficiency of 2-memory strategies is captured by Theorem 1 below.

By using standard linear programming methods (see, e.g., [23]), for every $C \in \mathcal{M}(G)$ we can compute the *minimal* and the *maximal* expected mean payoff achievable in C, denoted by α_C and β_C, in polynomial time (since C is strongly connected, the choice of the initial state is irrelevant). Thus, we can also compute the system L_{glob} of Fig. 2 in polynomial time. We show the following:

Theorem 1. *Let $u, v \in \mathbb{R}$. The following two statements hold true.*

1. *If there is a strategy ζ with $(\mathbb{E}^{\zeta}_{s_{in}}[mp], \mathbb{V}^{\zeta}_{s_{in}}[mp]) \leq (u, v)$ then the system L_{glob} of Fig. 2 has a non-negative solution.*
2. *If the system L_{glob} of Fig. 2 has a non-negative solution, then there is a 2-memory strategy σ with $(\mathbb{E}^{\sigma}_{s_{in}}[mp], \mathbb{V}^{\sigma}_{s_{in}}[mp]) \leq (u, v)$. In addition, it is possible to construct σ so that there is a number z such that for all $C \in \mathcal{M}(G)$ we have $\mathbb{V}^{\sigma}_{s_{in}}[mp \mid R_C] = 0$, and also have the following: If $\alpha_C > z$, then $\mathbb{E}^{\sigma}_{s_{in}}[mp \mid R_C] = \alpha_C$; if $\beta_C < z$, then $\mathbb{E}^{\sigma}_{s_{in}}[mp \mid R_C] = \beta_C$; otherwise $\mathbb{E}^{\sigma}_{s_{in}}[mp \mid R_C] = z$.*

Observe that the existence of Pareto optimal strategies follows from the above theorem, since we define points (u, v) that strategies can achieve by a continuous function from values x_C and $\sum_{t \in S \cap C} y_t$ for $C \in \mathcal{M}(G)$ to $\mathbb{R}^2$. Because the domain is bounded (all x_C and $\sum_{t \in S \cap C} y_t$ have minimal and maximal values) and closed (the points of the domain are expressible as a projection of feasible solutions of a linear program), it is also compact, and a continuous map of a compact set is compact (see, e.g., [24]), and hence closed.

3.1. Proof of Item 1 of Theorem 1 (from strategy to solution of constraints)

Our proof of Theorem 1 combines new techniques with results of [2] and [13]. We start with Item 1. Let ζ be a strategy satisfying $(\mathbb{E}^{\zeta}_{s_{in}}[mp], \mathbb{V}^{\zeta}_{s_{in}}[mp]) \leq (u, v)$. The way how ζ determines the values of all y_κ, where $\kappa \in S \cup A$, is exactly the same as in [2], but for clarity we outline the proof here.

Consider the MDP G' introduced before Lemma 2. By Lemma 2 there is a strategy ζ' for G' such that $\mathbb{P}^{\zeta}_{s_{in}}[R_C] = \mathbb{P}^{\zeta'}_{s_{in}}[Reach(\{d_s \mid s \in C\})]$. Since G' satisfies the conditions of [13, Theorem 3.2], we get a solution $\bar{y}$ to the linear program of [13, Figure 3] where for all C we have $\sum_{s \in C \cap S} \bar{y}_{d_s} = \mathbb{P}^{\zeta}_{s_{in}}[R_C]$. This solution gives us a solution to (1)–(2) by $y_t := \bar{y}_{d_t}$ for all $t \in S$, and $y_a := \bar{y}_{(s,a)}$ for all a (note that the state s is given uniquely as the state in which a is enabled). Because $\bar{y}_{d_t} = y_t$, we get that $\sum_{t \in C \cap S} y_t = \sum_{t \in C \cap S} \bar{y}_{d_t} = \mathbb{P}^{\zeta}_{s_{in}}[R_C]$.

The value of x_C is the conditional expected mean payoff under the condition that a run stays in C, i.e., $x_C = \mathbb{E}^{\zeta}_{s_{in}}[mp \mid R_C]$. Hence, $\alpha_C \leq x_C \leq \beta_C$, which means that (3) is satisfied. Further, $\mathbb{E}^{\zeta}_{s_{in}}[mp] = \sum_{C \in \mathcal{M}(G)} x_C \cdot \sum_{t \in S \cap C} y_t$ by the law of total

expectation and by Lemma 1, and hence (4) holds. Note that $\mathbb{V}_s^{\zeta}[mp]$ is *not* necessarily equal to the right-hand side of (5), and hence it is not immediately clear why (5) should hold. Here we need the following lemma.

Lemma 4. *Let $C \in \mathcal{M}(G)$, and let $z_C \in [\alpha_C, \beta_C]$. Then there exists a memoryless randomized strategy σ_{z_C} such that for every state $t \in C \cap S$ we have $\mathbb{P}_t^{\sigma_{z_C}}[mp{=}z_C] = 1$.*

Proof. Given a memoryless strategy σ and an action a, we use $f_\sigma(a) = \mathbb{E}_s^\sigma\left[\lim_{i\to\infty} \frac{1}{i} I_a(A_i)\right]$ for the frequency of action a.

Let σ_1 and σ_2 be memoryless deterministic strategies that minimize and maximize the expectation, respectively, and only yield one BSCC for any initial state. Let σ' be arbitrary memoryless randomized strategy that visits every action in C with nonzero frequency (such strategy clearly exists). We define the strategy σ_{z_C} as follows. If $z_C = \sum_{a\in C\cap A} f_{\sigma_2}(a)\cdot r(a)$, then $\sigma_{z_C} = \sigma_2$, and the result follows by the Ergodic theorem for Markov chains (see, e.g., [21, Theorem 1.10.2]). If $z_C > \sum_{a\in C\cap A} f_{\sigma'}(a)\cdot r(a)$, then, because also $z_C < \sum_{a\in C\cap A} f_{\sigma_2}(a)\cdot r(a)$, there must be a number $p \in (0,1]$ such that

$$z_C = p\cdot\Big(\sum_{a\in C\cap A} f_{\sigma'}(a)\cdot r(a)\Big) + (1-p)\cdot\Big(\sum_{a\in C\cap A} f_{\sigma_2}(a)\cdot r(a)\Big) \tag{6}$$

We define numbers $z_a = p\cdot f_{\sigma'}(a) + (1-p)\cdot f_{\sigma_2}(a)$ for all $a \in C\cap A$. Observe that we have, for any $s \in C$

$$\begin{aligned}
\sum_{a\in C\cap A} z_a\cdot\delta(a)(s) &= \sum_{a\in C\cap A}\Big(p\cdot f_{\sigma'}(a)\cdot\delta(a)(s) + (1-p)\cdot f_{\sigma_2}(a)\cdot\delta(a)(s)\Big) && \text{(def of } z_a\text{)}\\
&= p\cdot\Big(\sum_{a\in C\cap A} f_{\sigma'}(a)\cdot\delta(a)(s)\Big) + (1-p)\cdot\Big(\sum_{a\in C\cap A} f_{\sigma_2}(a)\cdot\delta(a)(s)\Big) && \text{(rearranging)}\\
&= p\cdot\Big(\sum_{a\in Act(s)} f_{\sigma'}(a)\Big) + (1-p)\cdot\Big(\sum_{a\in Act(s)} f_{\sigma_2}(a)\Big) && \text{(prop. of frequency functions)}\\
&= \sum_{a\in Act(s)}\Big(p\cdot f_{\sigma'}(a) + (1-p)\cdot f_{\sigma_2}(a)\Big) && \text{(rearranging)}
\end{aligned}$$

Hence, using [23, Section 9.3] we get a memoryless randomized strategy σ_{z_C} which for any starting state in C gets the expected mean payoff

$$\begin{aligned}
&\Big(\sum_{a\in C\cap A} p\cdot f_{\sigma'}(a)\cdot r(a)\Big) + \Big(\sum_{a\in C\cap A}(1-p)\cdot f_{\sigma_2}(a)\cdot r(a)\Big)\\
&\quad= p\cdot\Big(\sum_{a\in C\cap A} f_{\sigma'}(a)\cdot r(a)\Big) + (1-p)\cdot\Big(\sum_{a\in C\cap A} f_{\sigma_2}(a)\cdot r(a)\Big) && \text{(rearranging)}\\
&\quad= z_C && \text{(by (6))}
\end{aligned}$$

and because every action has non-zero frequency, by the Ergodic theorem for Markov chains almost every run has the same mean payoff. For the case $z_C < \sum_{a\in C\cap A} f_{\sigma'}(a)\cdot r(a)$ we proceed similarly, this time combining σ' with σ_1 instead of σ_2. □

Using Lemma 3 and Lemma 4, we can define another strategy ζ' from ζ such that for every $C \in \mathcal{M}(G)$ we have the following: (1) the probability of R_C in G_s^ζ and in $G_s^{\zeta'}$ is the same; (2) almost all runs $\omega \in R_C$ satisfy $mp(\omega) = x_C$. This means that $\mathbb{E}_s^\zeta[mp] = \mathbb{E}_s^{\zeta'}[mp]$, and we show:

Lemma 5. $\mathbb{V}_s^\zeta[mp] \geq \mathbb{V}_s^{\zeta'}[mp]$.

Proof. Since by law of total variance $\mathbb{V}(Z) = \mathbb{E}(\mathbb{V}(Z \mid Y)) + \mathbb{V}(\mathbb{E}(Z \mid Y))$ for all random variables Y and Z, we have for $\sigma \in \{\zeta, \zeta'\}$:

$$\mathbb{V}_s^\sigma[mp] = \Big(\sum_{C\in\mathcal{M}(G)} \mathbb{P}_s^\sigma[R_C]\cdot\mathbb{V}_s^\sigma[mp \mid R_C]\Big) + \mathbb{V}(X)$$

where X is the random variable which to every run of R_C assigns $\mathbb{E}_s^\sigma[mp \mid R_C]$. Note that the random variables X are equal for both ζ and ζ', and so also the second summands in the equation above are equal for ζ and ζ'. In the first summand, all the values $\mathbb{V}_s^\zeta[mp \mid R_C]$ are nonnegative, while $\mathbb{V}_s^{\zeta'}[mp \mid R_C]$ are zero. Consequently the variance can only decrease when we take ζ instead of ζ'. □

Hence, $(\mathbb{E}_s^{\zeta'}[mp], \mathbb{V}_s^{\zeta'}[mp]) \leq (u, v)$, and therefore (1)–(4) also hold if we use ζ' instead of ζ to determine the values of all variables. Further, the right-hand side of (5) is equal to $\mathbb{V}_s^{\zeta'}[mp]$, and hence (5) holds. This completes the proof of Item 1.

3.2. Proof of Item 2 of Theorem 1 (from solution of constraints to strategy)

Item 2 of Theorem 1 is proved as follows. Let y_κ, where $\kappa \in S \cup A$, and x_C, where $C \in \mathcal{M}(G)$, be a non-negative solution of L_{glob}. For every $C \in \mathcal{M}(G)$, we put $y_C = \sum_{t \in S \cap C} y_t$. By using [2, Lemma 4.4] (note that (1) and (2) are exactly the corresponding constraints in [2]), we construct a finite-memory strategy ϱ such that $\mathbb{P}^{\varrho}_{s_{in}}[R_C] = y_C$. Further, we consider a memoryless randomized strategy π which for every MEC C and its state t satisfies $\mathbb{P}^{\pi}_{t}[mp{=}x_C] = 1$; such strategy exists by Lemma 4. Then using Lemma 3 we obtain a 2-memory strategy $\hat{\sigma}$ with $(\mathbb{E}^{\hat{\sigma}}_{s_{in}}[mp], \mathbb{V}^{\hat{\sigma}}_{s_{in}}[mp]) \leq (u, v)$.

Finally, we transform $\hat{\sigma}$ into another 2-memory strategy σ which satisfies the additional conditions of Item 2 for a suitable z. This is achieved by modifying the behaviour of $\hat{\sigma}$ in some MECs so that the probability of staying in every MEC is preserved, the expected mean payoff is also preserved, and the global variance can only decrease. Here we use the following technical lemma.

Lemma 6. *Let B be a finite set with distinguished elements $b_1, \ldots b_n, b'_1 \ldots b'_m \in B$, let $X, Y : B \to \mathbb{R}$ be random variables, and let $d_1, \ldots d_n, d'_1, \ldots d'_m \geq 0$ be numbers satisfying the following:*

a. *For all $b \notin \{b_1, \ldots b_n, b'_1 \ldots b'_m\}$ we have $X(b) = Y(b)$.*
b. *There is x such that for all $1 \leq i \leq n$ and $1 \leq j \leq m$ we have $Y(b_i) \leq x$ and $Y(b'_j) \geq x$.*
c. *$X(b_i) + d_i = Y(b_i)$ for all $1 \leq i \leq n$.*
d. *$X(b_j) - d'_j = Y(b_j)$ for all $1 \leq j \leq m$.*
e. *$\mathbb{E}(X) = \mathbb{E}(Y)$.*

Then $\mathbb{V}(X) \geq \mathbb{V}(Y)$.

Proof. We need to show that $\mathbb{E}(X^2) - \mathbb{E}(X)^2 \geq \mathbb{E}(Y^2) - \mathbb{E}(Y)^2$ and because the expectations are equal by e. above, it suffices to prove that $\mathbb{E}(Y^2) - \mathbb{E}(X^2)$ is non-positive. We have

$$\begin{aligned}
\mathbb{E}(Y^2) - \mathbb{E}(X^2) &= \sum_{b \in B} Y(b)^2 \cdot \mathbb{P}(b) - \sum_{b \in B} X(b)^2 \cdot \mathbb{P}(b) && \text{(by def. of expectation)} \\
&= \sum_{i=1}^{n} Y(b_i)^2 \cdot \mathbb{P}(b_i) + \sum_{j=1}^{m} Y(b'_j)^2 \cdot \mathbb{P}(b_j) \\
&\quad - \sum_{i=1}^{n} X(b_i)^2 \cdot \mathbb{P}(b_i) - \sum_{j=1}^{m} X(b'_j)^2 \cdot \mathbb{P}(b_j) && \text{(by a.)} \\
&= \sum_{i=1}^{n} (X(b_i) + d_i)^2 \cdot \mathbb{P}(b_i) + \sum_{j=1}^{m} (X(b'_j) - d'_j)^2 \cdot \mathbb{P}(b'_j) \\
&\quad - \sum_{i=1}^{n} X(b_i)^2 \cdot \mathbb{P}(b_i) - \sum_{j=1}^{m} X(b'_j)^2 \cdot \mathbb{P}(b'_j) && \text{(by c and d.)} \\
&= \sum_{i=1}^{n} (2 \cdot d_i \cdot X(b_i) + (d_i)^2) \cdot \mathbb{P}(b_i) + \sum_{j=1}^{m} (-2 \cdot d'_j \cdot X(b'_j) + (d'_j)^2) \cdot \mathbb{P}(b'_j) && \text{(by arithmetic operations)} \\
&= \sum_{i=1}^{n} (2 \cdot X(b_i) + d_i) \cdot d_i \cdot \mathbb{P}(b_i) - \sum_{j=1}^{m} (2 \cdot X(b'_j) - d'_j) \cdot d'_j \cdot \mathbb{P}(b'_j) && \text{(by arithmetic operations)} \\
&= \sum_{i=1}^{n} (X(b_i) + Y(b_i)) \cdot d_i \cdot \mathbb{P}(b_i) - \sum_{j=1}^{m} (X(b'_j) + Y(b'_j)) \cdot d'_j \cdot \mathbb{P}(b'_j) && \text{(by c. and d.)} \\
&\leq \sum_{i=1}^{n} (2 \cdot x) \cdot d_i \cdot \mathbb{P}(b_i) - \sum_{j=1}^{m} (2 \cdot x) \cdot d'_j \cdot \mathbb{P}(b'_j) && \text{(by b., c. and d.)} \\
&= 2 \cdot x \cdot \Big(\sum_{i=1}^{n} d_i \cdot \mathbb{P}(b_i) - \sum_{j=1}^{m} d'_j \cdot \mathbb{P}(b'_j) \Big) && \text{(by arithmetic operations)}
\end{aligned}$$

To finish the proof, we show that $\sum_{i=1}^{n} d_i \cdot \mathbb{P}(b_i) - \sum_{j=1}^{m} d'_j \cdot \mathbb{P}(b'_j) = 0$. Indeed, we have:

$$
\begin{aligned}
0 &= \mathbb{E}(Y) - \mathbb{E}(X) && \text{(by e.)}\\
&= \sum_{b\in B} Y(b)\cdot\mathbb{P}(b) - \sum_{b\in B} X(b)\cdot\mathbb{P}(b) && \text{(by definition of expectation)}\\
&= \sum_{i=1}^{n}(X(b_i)+d_i)\cdot\mathbb{P}(b_i) + \sum_{j=1}^{m}(X(b'_j)-d'_j)\cdot\mathbb{P}(b'_j)\\
&\quad - \sum_{i=1}^{n} X(b_i)\cdot\mathbb{P}(b_i) - \sum_{j=1}^{m} X(b'_j)\cdot\mathbb{P}(b'_j) && \text{(by a., c and d.)}\\
&= \sum_{i=1}^{n} d_i\cdot\mathbb{P}(b_i) - \sum_{j=1}^{m} d'_j\cdot\mathbb{P}(b'_j) && \text{(by arithmetic operations)}
\end{aligned}
$$

This finishes the proof of the lemma. □

For a number z, we define $f(z) := \sum_{C\in\mathcal{M}(G)} y_C \cdot \gamma_C(z)$ where

$$
\gamma_C(z) = \begin{cases} \beta_C(z) & \text{if } z > \beta_C \\ \alpha_C(z) & \text{if } z < \alpha_C \\ z & \text{otherwise} \end{cases}
$$

Note that f is a continuous function, and that there is z with $f(z) = \mathbb{E}^{\hat{\sigma}}_{s_{in}}[mp]$. Lemma 6 shows that the strategy σ defined in the same way as $\hat{\sigma}$, but using γ_C where x_C was used, satisfies the required properties.

3.3. Complexity

We can solve the strategy existence problem by encoding the existence of a solution to L_{glob} as a closed formula Φ of the existential fragment of $(\mathbb{R}, +, *, \leq)$. Since Φ is computable in polynomial time and the existential fragment of $(\mathbb{R}, +, *, \leq)$ is decidable in polynomial space by [4], we obtain the following corollary.

Corollary 1. *The problem whether there is a strategy achieving a point (u, v) is in PSPACE.*

3.4. Approximation algorithm

In this subsection we show how to obtain a pseudo-polynomial-time approximation algorithm. First note that if we had the number z above, we could simplify the system L_{glob} of Fig. 2 by substituting all x_C variables with constants $\gamma_C(z)$. Then, (3) can be eliminated, (4) becomes a linear constraint, and (5) the only quadratic constraint. Thus, the system L_{glob} can be transformed into a quadratic program $L_{glob}(z)$ in which the quadratic constraint is negative semi-definite with rank 1, as the following lemma proves.

Lemma 7. *Let $n \in \mathbb{N}$ and $m_i \in \mathbb{N}$ for every $1 \leq i \leq n$. For all $1 \leq i \leq n$ and $1 \leq j \leq m_i$, we use $\langle i, j\rangle$ to denote the index $j + \sum_{\ell=1}^{i-1} m_\ell$. Consider a function $f : \mathbb{R}^k \to \mathbb{R}$, where $k = \sum_{i=1}^{n} m_i$, of the form*

$$
f(\vec{v}) \quad = \quad \left(\sum_{i=1}^{n}\left(\vec{c}_i^{\,2}\cdot\sum_{j=1}^{m_i}\vec{v}_{\langle i,j\rangle}\right)\right) \quad - \quad \left(\sum_{i=1}^{n}\left(\vec{c}_i\cdot\sum_{j=1}^{m_i}\vec{v}_{\langle i,j\rangle}\right)\right)^2
$$

where $\vec{c} \in \mathbb{R}^n$. Then $f(\vec{v})$ can be written as $f(\vec{v}) = \vec{v}^T Q\,\vec{v} + \vec{d}^T\vec{v}$ where Q is a negative semi-definite matrix of rank 1 and $\vec{d} \in \mathbb{R}^k$. Consequently, $f(\vec{v})$ is concave and Q has exactly one eigenvalue.

Proof. Observe that every $\vec{u} \in \mathbb{R}^k$ can be written as

$$
\vec{u}^T = (\vec{u}_{\langle 1,1\rangle}, \ldots, \vec{u}_{\langle 1,m_1\rangle}, \cdots, \vec{u}_{\langle n,1\rangle}, \ldots, \vec{u}_{\langle 1,m_n\rangle}).
$$

Let Q be $k \times k$ matrix where $Q_{\langle i,j\rangle,\langle i',j'\rangle} = -(\vec{c}_{i'}\cdot\vec{c}_i)$. Then

$$
(Q\,\vec{v})_{\langle i,j\rangle} \quad = \quad \sum_{i'=1}^{n}\sum_{j'=1}^{m_{i'}} Q_{\langle i,j\rangle,\langle i',j'\rangle}\cdot\vec{v}_{\langle i',j'\rangle} \quad = \quad -\sum_{i'=1}^{n}\sum_{j'=1}^{m_{i'}}(\vec{c}_{i'}\cdot\vec{c}_i)\cdot\vec{v}_{\langle i',j'\rangle} \tag{7}
$$

and consequently

$$\begin{aligned}
\vec{v}^T Q\, \vec{v} &= -\sum_{i=1}^{n}\sum_{j=1}^{m_i} \vec{v}_{\langle i,j\rangle} \cdot \Big(\sum_{i'=1}^{n}\sum_{j'=1}^{m_{i'}} (\vec{c}_{i'} \cdot \vec{c}_i)\vec{v}_{\langle i',j'\rangle}\Big) && \text{(by (7))}\\
&= -\sum_{i=1}^{n}\sum_{i'=1}^{n} (\vec{c}_i \cdot \vec{c}_{i'}) \cdot \sum_{j=1}^{m_i} \vec{v}_{\langle i,j\rangle} \cdot \sum_{j'=1}^{m_{i'}} \vec{v}_{\langle i',j'\rangle} && \text{(rearranging)}\\
&= -\Big(\sum_{i=1}^{n} \Big(\vec{c}_i \cdot \sum_{j=1}^{m_i} \vec{v}_{i,j}\Big)\Big)^2 && \text{(rearranging)}
\end{aligned}$$

Hence, $f(\vec{v}) = \vec{v}^T Q\, \vec{v} + \vec{d}^T \vec{v}$, where $\vec{d}_{\langle i,j\rangle} = \vec{c}_i^{\,2}$. Let $\vec{u} \in \mathbb{R}^k$ be a (fixed) vector such that $\vec{u}_{\langle i,j\rangle} = -\vec{c}_i$. Then the $\langle i', j'\rangle$-th column of Q is equal to $\vec{c}_{i'} \cdot \vec{u}$, which means that the rank of Q is 1. The matrix Q is negative semi-definite because $\vec{v}^T Q\, \vec{v} \le 0$ for every $\vec{v} \in \mathbb{R}^k$. □

Because (5) is of the form given in the statement in Lemma 7, we can approximate $L_{glob}(z)$ for any given z in polynomial time, using results of [28], up to a given $0 < \varepsilon < 1$. Since we do not know the precise number z, we try different candidates $\bar{z}$, namely we approximate the value (to the precision $\frac{\varepsilon}{2}$) of $L_{glob}(\bar{z})$ for all numbers $\bar{z}$ between $\min_{a\in A} r(a)$ and $\max_{a\in A} r(a)$ that are a multiple of $\tau = \frac{\varepsilon}{8\max\{N,1\}}$ where $N = \max_{a\in A} r(a)$. If any $L_{glob}(\bar{z})$ has a solution lower than $u - \frac{\varepsilon}{2}$, we output "yes", otherwise we output "no". The correctness of the algorithm is proved as follows.

Assume there is a strategy σ such that $(\mathbb{E}_s^\sigma[mp], \mathbb{V}_s^\sigma[mp]) \le (u-\varepsilon, v-\varepsilon)$, and let z be the number from Item 2, and let us fix a valuation for the variables y_κ where $\kappa \in S \cup A$ from equations of the system L_{glob} (see Fig. 2). Let $\bar{z}$ be a number between the minimal and the maximal assigned reward that is a multiple of τ, and which satisfies $|z - \bar{z}| < \tau$. Such a number must exist. We show that the system $L_{glob}(\bar{z})$ has a solution. The valuation fixed above can be applied to the system $L_{glob}(\bar{z})$, and we get

$$\begin{aligned}
&\Big|\sum_{C\in\mathcal{M}(G)} \gamma_C(\bar{z}) \cdot \sum_{t\in S\cap C} y_t - \sum_{C\in\mathcal{M}(G)} \gamma_C(z) \cdot \sum_{t\in S\cap C} y_t\Big|\\
&\quad\le \sum_{C\in\mathcal{M}(G)} |\gamma_C(\bar{z}) - \gamma_C(z)| \cdot \sum_{t\in S\cap C} y_t && \text{(by } y_t \ge 0\text{)}\\
&\quad\le \sum_{C\in\mathcal{M}(G)} \tau \cdot \sum_{t\in S\cap C} y_t && \text{(by } |\gamma_C(\bar{z}) - \gamma_C(z)| < \tau\text{)}\\
&\quad= \tau && \text{(by } \textstyle\sum_{C\in\mathcal{M}(G)}\sum_{t\in S\cap C} y_t = 1\text{)}
\end{aligned}$$

and so $\sum_{C\in\mathcal{M}(G)} \gamma_C(\bar{z}) \le (u-\varepsilon) + \tau \le u$.

For variance, let $M_{\bar{z}} = \sum_{C\in\mathcal{M}(G)} \gamma_C(\bar{z}) \cdot \sum_{t\in S\cap C} y_t$ and $M_z = \sum_{C\in\mathcal{M}(G)} \gamma_C(z) \cdot \sum_{t\in S\cap C} y_t$, from the above we have $|M_z - M_{\bar{z}}| \le \tau$ and so

$$\begin{aligned}
&\Big(\sum_{C\in\mathcal{M}(G)} \gamma_C(\bar{z})^2 \cdot \sum_{t\in S\cap C} y_t\Big) - \Big(\sum_{C\in\mathcal{M}(G)} \gamma_C(\bar{z}) \cdot \sum_{t\in S\cap C} y_t\Big)^2\\
&= \sum_{C\in\mathcal{M}(G)} (\gamma_C(\bar{z}) - M_{\bar{z}})^2 \cdot \sum_{t\in S\cap C} y_t && \text{(definition of variance)}\\
&\le \sum_{C\in\mathcal{M}(G)} \big(|\gamma_C(z) - M_z| + |\gamma_C(z) - \gamma_C(\bar{z})| + |M_z - M_{\bar{z}}|\big)^2 \cdot \sum_{t\in S\cap C} y_t && \text{(reformulating)}\\
&\le \sum_{C\in\mathcal{M}(G)} \big(|\gamma_C(z) - M_z| + 2\cdot\tau\big)^2 \cdot \sum_{t\in S\cap C} y_t && \text{(by } |\gamma_C(z) - \gamma_C(\bar{z})| \le \tau \text{ and } |M_z - M_{\bar{z}}| \le \tau\text{)}\\
&= \Big(\sum_{C\in\mathcal{M}(G)} \big((\gamma_C(z) - M_z)^2 \cdot \sum_{t\in S\cap C} y_t\big)\Big)\\
&\qquad + \big(4\cdot|\gamma_C(z) - M_z|\cdot\tau + 4\cdot\tau^2\big) \cdot \sum_{C\in\mathcal{M}(G)}\sum_{t\in S\cap C} y_t && \text{(rearranging)}\\
&= (v-\varepsilon) + \big(4\cdot N\cdot\tau + 4\cdot\tau^2\big) \cdot \sum_{t\in S\cap C} y_t && \text{(by } \mathbb{V}_s^\sigma[mp] \le v-\varepsilon \text{ and } \textstyle\sum_{C\in\mathcal{M}(G)}\sum_{t\in S\cap C} y_t = 1\text{)}
\end{aligned}$$

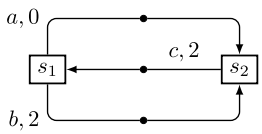

Fig. 3. An MDP showing that Pareto optimal strategies need randomization/memory for the local variance.

$$\leq (v-\varepsilon)+(\frac{\varepsilon}{2}+\frac{\varepsilon^2}{2}) \qquad \text{(def. of } \tau\text{)}$$

$$\leq v \qquad \text{(arithmetic operations, } \varepsilon < 1\text{)}$$

Hence we have shown that there is a solution for $L_{glob}(\bar{z})$, and so the algorithm returns "yes".

On the other hand, if there is no strategy such that $(\mathbb{E}^{\sigma}_{s}[mp], \mathbb{V}^{\sigma}_{s}[mp]) \leq (u, v)$, then the algorithm clearly returns "no". We obtain the following corollary.

Corollary 2. *The answer to the problem whether there is a strategy achieving a point (u, v) can be approximated in pseudo-polynomial time.*

Remark 1 and Corollary 2 immediately yield the following result.

Corollary 3. *The approximate Pareto curve for global variance can be computed in pseudo-polynomial time.*

Note that if we *knew* the constant z, we would even get that the approximation problem for a point (u, v) can be solved in polynomial time (assuming that the number of digits in z is polynomial in the size of the problem instance). Unfortunately, our proof of Item 2 does not give a procedure for computing z, and we cannot even conclude that z is rational. We conjecture that the constant z *can* actually be chosen as a rational number with small number of digits (which would immediately lower the complexity of strategy existence to **NP** using the results of [27] for solving negative semi-definite quadratic programs).

4. Local variance

In this section we analyse the problem for local variance. As before, we start by showing the lower bounds for memory needed by strategies, and then provide an upper bound together with an algorithm computing a Pareto optimal strategy. As in the case of global variance, Pareto optimal strategies require both randomization and memory, however, in contrast with global variance where for unichain MDPs deterministic memoryless strategies are sufficient we show (in the following example) that for local variance both memory and randomization are required even for unichain MDPs.

Example 2. Consider the MDP from Fig. 3 and consider a strategy σ that in the first step in s_1 makes a random choice uniformly between a and b, and then, whenever the state s_1 is revisited, it chooses the action that was chosen in the first step. The expected mean payoff under such strategy is

$$\begin{aligned}\mathbb{E}^{\sigma'}_{s_1}[mp] &= \mathbb{P}^{\sigma'}_{s_1}[Cone(s_1as_2)]\cdot\mathbb{E}^{\sigma'}_{s_1}[mp \mid Cone(s_1as_2)] \\ &\quad + \mathbb{P}^{\sigma'}_{s_1}[Cone(s_1bs_2)]\cdot\mathbb{E}^{\sigma'}_{s_1}[mp \mid Cone(s_1bs_2)] \\ &= 0.5\cdot 1 + 0.5\cdot 2 = 1.5\end{aligned}$$

and the expected local variance is

$$\begin{aligned}\mathbb{E}^{\sigma'}_{s_1}[lv] &= \mathbb{P}^{\sigma'}_{s_1}[Cone(s_1as_2)]\cdot\mathbb{E}^{\sigma'}_{s_1}[lv \mid Cone(s_1as_2)] \\ &\quad + \mathbb{P}^{\sigma'}_{s_1}[Cone(s_1bs_2)]\cdot\mathbb{E}^{\sigma'}_{s_1}[lv \mid Cone(s_1bs_2)] \\ &= 0.5\cdot\mathbb{E}^{\sigma'}_{s_1}[lv \mid Cone(s_1as_2)] + 0.5\cdot\mathbb{E}^{\sigma'}_{s_1}[lv \mid Cone(s_1bs_2)] \\ &= 0.5\cdot\big(0.5\cdot(0-\mathbb{E}^{\sigma'}_{s_1}[mp \mid Cone(s_1as_2)])^2 \\ &\quad + 0.5\cdot(2-\mathbb{E}^{\sigma'}_{s_1}[mp \mid Cone(s_1as_2)])^2\big) \\ &\quad + 0.5\cdot(2-\mathbb{E}^{\sigma'}_{s_1}[mp \mid Cone(s_1bs_2)])^2 \\ &= 0.5\cdot\big(0.5\cdot(0-1)^2 + 0.5\cdot(2-1)^2\big) + 0.5\cdot(2-2)^2 \\ &= 0.5\end{aligned}$$

$$I_{s_{in}}(s) + \sum_{a \in A} y_a \cdot \delta(a)(s) = \sum_{a \in Act(s)} y_a + y_{s,1} + y_{s,2} \quad \text{for all } s \in S \tag{8}$$

$$\sum_{C \in \mathcal{M}(G)} \sum_{s \in C \cap S} y_{s,1} + y_{s,2} = 1 \tag{9}$$

$$\sum_{a \in C \cap A} x_{a,i} = 1 \quad \text{for all } C \in \mathcal{M}(G) \tag{10}$$

$$\sum_{a \in A} x_{a,i} \cdot \delta(a)(s) = \sum_{a \in Act(s)} x_{a,i} \quad \text{for all } s \in S \text{ and } i \in \{1,2\} \tag{11}$$

$$z_{mp,C,i} = \sum_{a \in C \cap A} x_{a,i} \cdot r(a) \quad \text{for all } C \in \mathcal{M}(G) \text{ and } i \in \{1,2\} \tag{12}$$

$$z_{lv,C,i} = \sum_{a \in C \cap A} (x_{a,i} \cdot r(a) - z_{mp,C,i})^2 \quad \text{for all } C \in \mathcal{M}(G) \text{ and } i \in \{1,2\} \tag{13}$$

$$u \geq \sum_{C \in \mathcal{M}(G)} \sum_{i \in \{1,2\}} \Big(\sum_{s \in C \cap S} y_{s,i} \Big) \cdot z_{mp,C,i} \tag{14}$$

$$v \geq \sum_{C \in \mathcal{M}(G)} \sum_{i \in \{1,2\}} \Big(\sum_{s \in C \cap S} y_{s,i} \Big) \cdot z_{lv,C,i} \tag{15}$$

Fig. 4. The set of constraints L_{loc} for local variance.

We show that the point $(1.5, 0.5)$ cannot be achieved by any memoryless randomized strategy σ'. Given $x \in \{a, b, c\}$, denote by $f(x)$ the frequency of the action x under σ', i.e. $f(x) = \mathbb{E}^{\sigma}_{s_1}\left[\lim_{i \to \infty} \frac{1}{i} I_x(A_i)\right]$. Clearly, $f(c) = 0.5$ and $f(b) = 0.5 - f(a)$. If $f(a) < 0.2$, then the mean payoff $\mathbb{E}^{\sigma'}_{s_1}[mp] = 2 \cdot (f(c) + f(b)) = 2 - 2f(a)$ is greater than 1.6. Assume that $0.2 \leq f(a) \leq 0.5$. Then $\mathbb{E}^{\sigma'}_{s_1}[mp] \leq 1.6$ but the expected local variance is at least 0.64:

$$\begin{aligned}
\mathbb{E}^{\sigma'}_{s_1}[lv] &= f(a)(0 - \mathbb{E}^{\sigma'}_{s_1}[mp])^2 + (f(b) + f(c))(2 - \mathbb{E}^{\sigma'}_{s_1}[mp])^2 \\
&= f(a)(-2 + 2f(a)))^2 + (1 - f(a))(2f(a))^2 \\
&= 4f(a) - 8f(a)^2 + 4f(a)^3 + 4f(a)^2 - 4f(a)^3 \\
&= 4f(a) - 4f(a)^2 \geq 0.64
\end{aligned}$$

Insufficiency of deterministic history-dependent strategies is proved using the same equations and the fact that there is only one run under such a strategy.

Thus we have shown that memory and randomization are needed to achieve also a non-Pareto point $(1.51, 0.51)$. The need of memory and randomization to achieve Pareto points will follow later from the fact that there always exist Pareto optimal strategies.

The main result of this section is described in the following theorem.

Theorem 2. *There is a strategy ζ satisfying $(\mathbb{E}^{\zeta}_{s_{in}}[mp], \mathbb{E}^{\zeta}_{s_{in}}[lv]) \leq (u, v)$ if and only if the set of constraints from Fig. 4 has a non-negative solution.*

We will prove the theorem in the following two subsections.

4.1. Proof of direction ⇒ of Theorem 2 (from strategy to solution of constraints)

Our proof relies on the fact that any achievable mean payoff and local variance can be extracted as a combination of two frequency functions. The idea is formalised in Proposition 1 below, but before proceeding, we prove the following easy lemma.

Lemma 8. *Let $(a_1, b_1), (a_2, b_2), \ldots, (a_m, b_m)$ be a sequence of points in $\mathbb{R}^2$ and let $c_1, c_2, \ldots, c_m \in (0, 1]$ be numbers satisfying $\sum_{i=1}^{m} c_i = 1$. Then there are two points (a_k, b_k) and (a_ℓ, b_ℓ) and a number $p \in [0, 1]$ such that*

$$\sum_{i=1}^{m} c_i(a_i, b_i) \quad \geq \quad p(a_k, b_k) + (1 - p)(a_\ell, b_\ell)$$

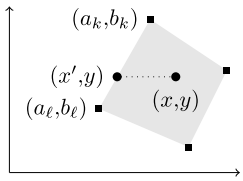

Fig. 5. An illustration for the proof of Lemma 8, with the points of H shown as squares, and $\mathcal{C}(H)$ shown as a greyed area.

Proof. Denote by (x, y) the point $\sum_{i=1}^{m} c_i(a_i, b_i)$ and by H the set $\{(a_i, b_i) \mid 1 \leq i \leq m\}$. The convex hull $\mathcal{C}(H)$ of H is a convex polygon whose vertices are some of the points of H. Consider a point (x', y) where $x' = \min\{z \mid z \leq x, (z, y) \in \mathcal{C}(H)\}$. The point (x', y) lies on the boundary of $\mathcal{C}(H)$ and thus, as $\mathcal{C}(H)$ is a convex polygon, (x', y) lies on the line segment between two vertices, say $(a_k, b_k), (a_\ell, b_\ell)$, of $\mathcal{C}(H)$. Thus there is $p \in [0, 1]$ such that

$$(x', y) = p(a_k, b_k) + (1 - p)(a_\ell, b_\ell) \leq (x, y) = \sum_{i=1}^{m} c_i(a_i, b_i)\,.$$

This finishes the proof. For reader's convenience we illustrate the proof in Fig. 5. □

Let us now proceed with formalising the main essence of the proof.

Proposition 1. *Let us fix a MEC C and let $\varepsilon \geq 0$. There are two frequency functions $f_\varepsilon : C \cap A \to [0, 1]$ and $f'_\varepsilon : C \cap A \to [0, 1]$, and a number $p_\varepsilon \in [0, 1]$ such that:*

$$p_\varepsilon \cdot (mp[f_\varepsilon], lv[f_\varepsilon]) + (1 - p_\varepsilon) \cdot (mp[f'_\varepsilon], lv[f'_\varepsilon]) \leq (\mathbb{E}^{\zeta}_{s_{in}}[mp|R_C]\,, \mathbb{E}^{\zeta}_{s_{in}}[lv|R_C]) + (\varepsilon, \varepsilon)$$

The proposition is proved in two steps, for the first and simpler step, we show that if the proposition holds for every $\varepsilon > 0$, then it holds for $\varepsilon = 0$. There is a sequence $\varepsilon_1, \varepsilon_2, \ldots$, two functions f_C and f'_C, and $p_C \in [0, 1]$ such that $\lim_{n\to\infty} \varepsilon_n = 0$, $\lim_{n\to\infty} p_{\varepsilon_n} = p_C$, and as $n \to \infty$:

- f_{ε_n} converges pointwise to f_C
- f'_{ε_n} converges pointwise to f'_C

It is easy to show that f_C as well as f'_C are frequency functions. Moreover, as

$$\lim_{n\to\infty} (\mathbb{E}^{\zeta}_{s_{in}}[mp \mid R_C]\,, \mathbb{E}^{\zeta}_{s_{in}}[lv \mid R_C]) + (\varepsilon_n, \varepsilon_n) = (\mathbb{E}^{\zeta}_{s_{in}}[mp \mid R_C]\,, \mathbb{E}^{\zeta}_{s_{in}}[lv \mid R_C])$$

and

$$\lim_{n\to\infty} p_{\varepsilon_n} \cdot (mp[f_{\varepsilon_n}], lv[f_{\varepsilon_n}]) + (1 - p_{\varepsilon_n}) \cdot (mp[f'_{\varepsilon_n}], lv[f'_{\varepsilon_n}]) = p_C \cdot (mp[f_C], lv[f_C]) + (1 - p_C) \cdot (mp[f'_C], lv[f'_C])$$

we obtain

$$p_C \cdot (mp[f_C], lv[f_C]) + (1 - p_C) \cdot (mp[f'_C], lv[f'_C]) \leq (\mathbb{E}^{\zeta}_{s_{in}}[mp \mid R_C]\,, \mathbb{E}^{\zeta}_{s_{in}}[lv \mid R_C])$$

The more involved step of the proof of Proposition 1 is to show that it holds for every $\varepsilon > 0$. We prove this by showing that there are runs ω from which we can extract the frequency functions f_ε and f'_ε. The selection of runs is rather involved, since it is not clear a priori which runs to pick or even how to extract the frequencies from them (note that the naive approach of considering the average ratio of taking a given action a does not work, since the averages might not be defined).

Given $\ell, k \in \mathbb{Z}$ we denote by $A^{\ell,k}$ the set of all runs $\omega \in R_C$ such that

$$(\ell \cdot \varepsilon, k \cdot \varepsilon) \quad \leq \quad (mp(\omega), lv(\omega)) \quad < \quad (\ell \cdot \varepsilon, k \cdot \varepsilon) + (\varepsilon, \varepsilon)$$

Note that

$$\sum_{\ell,k\in\mathbb{Z}} \mathbb{P}^{\zeta}_{s_{in}}(A^{\ell,k} \mid R_C) \cdot (\ell \cdot \varepsilon, k \cdot \varepsilon) \quad \leq \quad (\mathbb{E}^{\zeta}_{s_{in}}[mp \mid R_C]\,, \mathbb{E}^{\zeta}_{s_{in}}[lv \mid R_C])$$

By Lemma 8, there are $\ell, k, \ell', k' \in \mathbb{Z}$ and $p \in [0, 1]$ such that $\mathbb{P}^{\zeta}_{s_{in}}(A^{\ell,k} \mid R_C) > 0$ and $\mathbb{P}^{\zeta}_{s_{in}}(A^{\ell',k'} \mid R_C) > 0$ and

$$\begin{aligned} p \cdot (\ell \cdot \varepsilon, k \cdot \varepsilon) + (1 - p) \cdot (\ell' \cdot \varepsilon, k' \cdot \varepsilon) &\leq \sum_{\ell,k\in\mathbb{Z}} \mathbb{P}^{\zeta}_{s_{in}}(A^{\ell,k} \mid R_C) \cdot (\ell \cdot \varepsilon, k \cdot \varepsilon) \\ &\leq (\mathbb{E}^{\zeta}_{s_{in}}[mp \mid R_C]\,, \mathbb{E}^{\zeta}_{s_{in}}[lv \mid R_C]) \end{aligned} \tag{16}$$

Let us concentrate on $(\ell \cdot \varepsilon, k \cdot \varepsilon)$ and construct a frequency function f on C such that

$$(mp[f], lv[f]) \quad \leq \quad (\ell \cdot \varepsilon, k \cdot \varepsilon) + (\varepsilon, \varepsilon)$$

Intuitively, we obtain f as a vector of frequencies of individual actions on an appropriately chosen run of R_C. Such frequencies determine the average and variance close to $\ell \cdot \varepsilon$ and $k \cdot \varepsilon$, respectively. We have to deal with some technical issues, mainly with the fact that the frequencies might not be well-defined for almost all runs (i.e., the corresponding limits might not exist). This is solved by a careful choice of subsequences as follows.

Claim 1. *For every run $\omega \in R_C$ there is a sequence of numbers $T_1[\omega], T_2[\omega], \ldots$ such that all the following limits are defined:*

$$\lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} r(A_j(\omega)) \quad = \quad mp(\omega) \tag{17}$$

$$\lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} (r(A_j(\omega)) - mp(\omega))^2 \quad \leq \quad lv(\omega) \tag{18}$$

and for every action $a \in A$ there is a number $f_\omega(a)$ such that

$$\lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega)) \quad = \quad f_\omega(a)$$

Moreover, for almost all runs ω of R_C we have that f_ω is a frequency function on C and that f_ω determines $(mp(\omega), lv(\omega))$, i.e., $mp(\omega) = mp[f_\omega]$ and $lv(\omega) \geq lv[f_\omega]$.

Proof. We start by taking a sequence $T'_1[\omega], T'_2[\omega], \ldots$ such that

$$\lim_{i\to\infty} \frac{1}{T'_i[\omega]} \sum_{j=1}^{T'_i[\omega]} r(A_j(\omega)) \quad = \quad mp(\omega)$$

Existence of such a sequence follows from the fact that every sequence of real numbers has a subsequence which converges to the lim sup of the original sequence.

Now we extract a subsequence $T''_1[\omega], T''_2[\omega], \ldots$ of $T'_1[\omega], T'_2[\omega], \ldots$ such that

$$\lim_{i\to\infty} \frac{1}{T''_i[\omega]} \sum_{j=1}^{T''_i[\omega]} (r(A_j(\omega)) - mp(\omega))^2 \quad \leq \quad lv(\omega) \tag{19}$$

using the same argument.

Now assuming an order on actions, $a_1, \ldots, a_m$, we define $T^k_1[\omega], T^k_2[\omega], \ldots$ for $0 \leq k \leq m$ so that $T^0_1[\omega], T^0_2[\omega], \ldots$ is the sequence $T''_1[\omega], T''_2[\omega], \ldots$, and every $T^{k+1}_1[\omega], T^{k+1}_2[\omega], \ldots$ is a subsequence of $T^k_1[\omega], T^k_2[\omega], \ldots$ such that the following limit exists (and is equal to a number $f_\omega(a_{k+1})$)

$$\lim_{i\to\infty} \frac{1}{T^{k+1}_i[\omega]} \sum_{j=1}^{T^{k+1}_i[\omega]} I_{a_{k+1}}(A_j(\omega))$$

We take $T^m_1[\omega], T^m_2[\omega], \ldots$ to be the desired sequence $T_1[\omega], T_2[\omega], \ldots$.

Now we have to prove that f_ω is a frequency function on C for almost all runs of R_C. Clearly, $0 \leq f_\omega(a) \leq 1$ for all $a \in C \cap A$. Also,

$$\begin{aligned}
\sum_{a\in C\cap A} f_\omega(a) &= \sum_{a\in C\cap A} \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega)) \\
&= \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} \sum_{a\in C\cap A} I_a(A_j(\omega)) \\
&= \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} 1 = 1
\end{aligned}$$

To prove the third condition from the definition of frequency functions, we use a variant of strong law of large numbers. Given a run ω, an action a, a state s and $k \geq 1$, define

$$N_k^{a,s}(\omega) = \begin{cases} 1 - \delta(a)(s) & \text{if } a \text{ is executed at least } i \text{ times, and } s \text{ is visited} \\ & \text{immediately after the } i\text{-th execution of } a; \\ -\delta(a)(s) & \text{if } a \text{ is executed at least } i \text{ times, and } s \text{ is not} \\ & \text{visited immediately after the } i\text{-th execution of } a; \\ 0 & \text{otherwise (i.e., } a \text{ is executed at most } i-1 \text{ times).} \end{cases}$$

Fix a and s. We show that $\mathbb{E}(N_i^{a,s} \cdot N_j^{a,s}) = 0$ for $i \neq j$. W.l.o.g. suppose $i > j$. We have

$$\begin{aligned} \mathbb{E}(N_i^{a,s} \cdot N_j^{a,s}) &= \sum_n \mathbb{E}(N_i^{a,s} \cdot N_j^{a,s} \mid N_j^{a,s} = n) \cdot \mathbb{P}(N_j^{a,s} = n) \\ &= \sum_n n \cdot \mathbb{E}(N_i^{a,s} \mid N_j^{a,s} = n) \cdot \mathbb{P}(N_j^{a,s} = n) \end{aligned}$$

which equals 0 because $\mathbb{E}(N_i^{a,s} \mid N_j^{a,s} = n) = 0$ for all $i > j$ and n. Hence, we can use [18, Corollary 4] (where we substitute $\Phi_1(0) = 1$ and $\Phi(i) = 0$ for $i > 0$) and obtain that almost surely the following equality holds:

$$\lim_{j\to\infty} \frac{\sum_{k=1}^{j} N_k^{a,s}(\omega)}{j} = 0 \tag{20}$$

We let $\bar{N}_k^{a,s} = N_k^{a,s} + \delta(a)(s)$, and obtain

$$\begin{aligned} &\sum_{a \in C \cap A} f_\omega(a) \cdot \delta(a)(s) \\ &= \sum_{a \in C \cap A} \Big(\lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega)) \Big) \cdot \lim_{i\to\infty} \frac{1}{i} \sum_{k=1}^{i} \bar{N}_k^{a,s}(\omega) && \text{(def. of } f_\omega, (20) \text{ and def. of } \bar{N}_k^{a,s}) \\ &= \sum_{a \in C \cap A} \Big(\lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega)) \Big) \\ &\qquad \cdot \lim_{i\to\infty} \frac{1}{\sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega))} \sum_{k=1}^{\sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega))} \bar{N}_k^{a,s}(\omega) && \text{(taking subsequence)} \\ &= \sum_{a \in C \cap A} \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{k=1}^{\sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega))} \bar{N}_k^{a,s}(\omega) && \text{(property of lim, and arithmetic opers.)} \\ &= \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{a \in C \cap A} \sum_{k=1}^{\sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega))} \bar{N}_k^{a,s}(\omega) && \text{(splitting by } a) \\ &= \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} I_s(S_j(\omega)) && \text{(def. of } N_k^{a,s} \text{ and } \bar{N}_k^{a,s}) \\ &= \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} \sum_{a \in Act(s)} I_a(A_j(\omega)) && \text{(splitting by } a) \\ &= \sum_{a \in Act(s)} \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega)) && \text{(linearity of lim)} \\ &= \sum_{a \in Act(s)} f_\omega(a) && \text{(def of } f_\omega) \end{aligned}$$

Here $S_j(\omega)$ is the j-th state of ω.

$$\begin{aligned} mp(\omega) &= \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} r(A_j(\omega)) \\ &= \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} \sum_{a\in C\cap A} I_a(A_j(\omega)) \cdot r(a) \\ &= \sum_{a\in C\cap A} r(a) \cdot \lim_{i\to\infty} \frac{1}{T_i[\omega]} \sum_{j=1}^{T_i[\omega]} I_a(A_j(\omega)) \\ &= \sum_{a\in C\cap A} r(a) \cdot f_\omega(a) \\ &= mp[f_\omega] \end{aligned}$$

and replacing $r(x)$ with $(r(x) - mp(\omega))^2$ in the derivation, we also get $lv(\omega) = lv[f_\omega]$. □

Now pick an arbitrary run ω of $A^{k,\ell}$ such that f_ω is a frequency function. Then

$$(mp[f_\omega], lv[f_\omega]) \le (mp(\omega), lv(\omega)) \le (\ell\cdot\varepsilon, k\cdot\varepsilon) + (\varepsilon,\varepsilon)$$

Similarly, for ℓ', k' we obtain f'_ω such that

$$(mp[f'_\omega], lv[f'_\omega]) \le (mp(\omega), lv(\omega)) \le (\ell'\cdot\varepsilon, k'\cdot\varepsilon) + (\varepsilon,\varepsilon)$$

This together with (16) from page 158 gives

$$\begin{aligned} &p\cdot(mp[f_\omega], lv[f_\omega]) + (1-p)\cdot(mp[f'_\omega], lv[f'_\omega]) \\ &\qquad \le p\cdot((\ell\cdot\varepsilon, k\cdot\varepsilon) + (\varepsilon,\varepsilon)) + (1-p)\cdot\big((\ell'\cdot\varepsilon, k'\cdot\varepsilon) + (\varepsilon,\varepsilon)\big) \\ &\qquad \le (\mathbb{E}^\zeta_{s_{in}}[mp \mid R_C], \mathbb{E}^\zeta_{s_{in}}[lv \mid R_C]) + (\varepsilon,\varepsilon) \end{aligned}$$

and thus finishes the proof of Proposition 1.

We are now ready to finish the proof of direction $\Rightarrow$ of Theorem 2. For every MEC C we use Proposition 1 to obtain frequency functions f^C_1 and f^C_2 and a number p^C satisfying the conditions of the proposition for $\varepsilon = 0$, and we set the value of $x_{a,1}$ and $x_{a,2}$ to $f^C_1(a)$ and $f^C_2(a)$, respectively, where C is the MEC containing a. This ensures satisfaction of (10) and (11).

The values to y_a and $y_{s,i}$ are determined by applying the construction very similar to the one from the proof in Section 3.1 (page 151). Consider the MDP G' introduced before Lemma 2. By Lemma 2 there is a strategy ζ' for G' such that $\mathbb{P}^\zeta_{s_{in}}[R_C] = \mathbb{P}^{\zeta'}_{s_{in}}[Reach(\{d_s \mid s\in C\})]$. Since G' satisfies the conditions of [13, Theorem 3.2], we get a solution $\bar{y}$ to the linear program of [13, Figure 3] where for all C we have $\sum_{s\in C\cap S}\bar{y}_{d_s} = \mathbb{P}^\zeta_{s_{in}}[R_C]$. This solution gives us a solution to (8)–(9) by $y_a := \bar{y}_{(s,a)}$ for all a (note that the state s is given uniquely as the state in which a is enabled), and by $y_{t,1} := \bar{y}_{d_t}\cdot p^C$ and $y_{t,1} := \bar{y}_{d_t}\cdot(1-p^C)$ for all $t\in S$, where p^C is the number obtained above for MEC C. We further set $z_{mp,C,i}$ and $z_{lv,C,i}$ to $mp[f^C_i]$ and $lv[f^C_i]$, and get that (12) and (13) hold true, and consequently, we get that (14) and (15) are also satisfied, because

$$\begin{aligned} (u,v) &\ge (\mathbb{E}^\zeta_{s_{in}}[mp], \mathbb{E}^\zeta_{s_{in}}[lv]) && \text{(property of } \zeta\text{)} \\ &= \sum_{C\in\mathcal{M}(G)} \mathbb{P}^\zeta_{s_{in}}[R_C]\cdot(\mathbb{E}^\zeta_{s_{in}}[mp \mid R_C], \mathbb{E}^\zeta_{s_{in}}[lv \mid R_C]) && \text{(law of total expect.)} \\ &\ge \sum_{C\in\mathcal{M}(G)} \mathbb{P}^\zeta_{s_{in}}[R_C]\cdot\big(p^C\cdot(mp[f^C_1], lv[f^C_1]) + (1-p^C)\cdot(mp[f^C_2], lv[f^C_2])\big) && \text{(def. of } f^C_i\text{)} \\ &= \sum_{C\in\mathcal{M}(G)} \Big(\sum_{s\in C\cap S}\bar{y}_{d_s}\Big)\cdot\big(p^C\cdot(mp[f^C_1], lv[f^C_1]) + (1-p^C)\cdot(mp[f^C_2], lv[f^C_2])\big) && \text{(def. of } \bar{y}_{d_s}\text{)} \\ &= \sum_{C\in\mathcal{M}(G)}\sum_{i\in\{1,2\}}\sum_{s\in C\cap S} y_{s,i}\cdot(mp[f^C_i], lv[f^C_i]) && \text{(def. of } y_{t,i} \text{ and rearranging)} \end{aligned}$$

4.2. Proof of direction ⇐ of Theorem 2 (from solution of constraints to strategy)

In order to prove the direction ⇐ of Theorem 2, we first introduce two auxiliary lemmas. The following lemma shows how to minimize the mean square deviation (to which our notion of local variance is a special case).

Lemma 9. *Let $a_1, \ldots, a_m \in \mathbb{R}$ such that $\sum_{i=0}^{m} a_i = 1$, let $r_1, \ldots, r_m \in \mathbb{R}$ and let us consider the following function of one real variable:*

$$V(x) = \sum_{i=1}^{m} a_i \, (r_i - x)^2$$

Then the function V has a unique minimum in $\sum_{i=1}^{m} a_i r_i$.

Proof. By taking the first derivative of V we obtain

$$\frac{\delta V}{\delta x} = -2 \cdot \sum_{i=1}^{m} a_i \, (r_i - x) = -2 \cdot \left(\sum_{i=1}^{m} a_i r_i \right) + 2x$$

Thus $\frac{\delta V}{\delta x}(x) = 0$ iff $x = \sum_{i=1}^{m} a_i r_i$. Moreover, by taking the second derivative we obtain $\frac{\delta^2 V}{\delta x^2} = 2 > 0$, and thus $\sum_{i=1}^{m} a_i r_i$ is a minimum. □

The following lemma shows that frequencies of actions determine (in some cases) the expected mean payoff as well as the expected local variance.

Lemma 10. *Let μ be a memoryless strategy and let D be a BSCC of G^μ, and let s be an arbitrary state of D. The following equalities hold:*

$$\mathbb{E}_s^\mu[mp] = \sum_{a \in A} r(a) \cdot \mathbb{E}_s^\mu\left[mp^{I_a}\right] \quad \text{and} \quad \mathbb{E}_s^\mu[lv] = \sum_{a \in A} (r(a) - \mathbb{E}_s^\mu[mp])^2 \cdot \mathbb{E}_s^\mu\left[mp^{I_a}\right]$$

Proof. We have

$$\begin{aligned}
\mathbb{E}_s^\mu[mp] &= \mathbb{E}_s^\mu\left[\lim_{i\to\infty} \frac{1}{i} \cdot \sum_{j=1}^{i} r(A_j)\right] \\
&= \mathbb{E}_s^\mu\left[\lim_{i\to\infty} \frac{1}{i} \cdot \sum_{j=1}^{i} \sum_{a \in A} r(a) \cdot I_a(A_j)\right] \\
&= \sum_{a \in A} r(a) \cdot \mathbb{E}_s^\mu\left[\lim_{i\to\infty} \frac{1}{i} \cdot \sum_{j=1}^{i} I_a(A_j)\right] \\
&= \sum_{a \in A} r(a) \cdot \mathbb{E}_s^\mu\left[mp^{I_a}\right]
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{E}_s^\mu[lv] &= \mathbb{E}_s^\mu\left[\lim_{i\to\infty} \frac{1}{i} \cdot \sum_{j=1}^{i} (r(A_j) - \mathbb{E}_s^\mu[mp])^2\right] \\
&= \mathbb{E}_s^\mu\left[\lim_{i\to\infty} \frac{1}{i} \cdot \sum_{j=1}^{i} \sum_{a \in A} (r(a) - \mathbb{E}_s^\mu[mp])^2 \cdot I_a(A_j)\right] \\
&= \sum_{a \in A} (r(a) - \mathbb{E}_s^\mu[mp])^2 \cdot \mathbb{E}_s^\mu\left[\lim_{i\to\infty} \frac{1}{i} \cdot \sum_{j=1}^{i} I_a(A_j)\right] \\
&= \sum_{a \in A} (r(a) - \mathbb{E}_s^\mu[mp])^2 \cdot \mathbb{E}_s^\mu\left[mp^{I_a}\right]
\end{aligned}$$

where the first equality above holds true because in BSCCs almost all runs have the same frequencies of actions, and so we can replace mp with $\mathbb{E}_s^\mu[mp]$. □

We now show that any two frequency functions can be "mimicked" by two memoryless randomized strategies to achieve the required expected mean payoff and expected local variance.

Lemma 11. *Let C be a MEC, f and f' be two frequency functions, and $0 \leq p \leq 1$ a number. Then there are two memoryless randomized strategies π and π', each yielding a single BSCC, and a number $0 \leq d \leq 1$ such that*

$$d \cdot (\mathbb{E}_s^{\pi}[mp], \mathbb{E}_s^{\pi}[lv]) + (1-d) \cdot (\mathbb{E}_s^{\pi'}[mp], \mathbb{E}_s^{\pi'}[lv]) \\ \leq p \cdot (mp[f], lv[f]) + (1-p) \cdot (mp[f'], lv[f'])$$

for all states s from C.

Proof. We start with defining memoryless strategies κ and κ' (which possibly yield several BSCCs) in C as follows: Given $s \in C \cap S$ such that $\sum_{b \in A(s)} f(b) > 0$ and $a \in A(s)$, we put

$$\kappa(s)(a) = f(a) \,/\, \sum_{b \in A(s)} f(b) \qquad \text{and} \qquad \kappa'(s)(a) = f'(a) \,/\, \sum_{b \in A(s)} f'(b)$$

In the remaining states s the strategy κ (or κ') behaves as a memoryless deterministic strategy reaching $\{s \in C \cap S \mid \sum_{b \in Act(s)} f(b) > 0\}$ (or $\{s \in C \cap S \mid \sum_{b \in Act(s)} f'(b) > 0\}$, resp.) with probability one.

Given a BSCC D of C^{κ} (or D' of $C^{\kappa'}$), we write $f(D)$ for $\sum_{a \in D \cap A} f(a)$ (or $f'(D')$ for $\sum_{a \in D' \cap A} f'(a)$). We have

$$(mp[f], lv[f]) = \sum_{D \in \mathcal{B}(C^{\kappa})} f(D) \cdot \Big(\sum_{a \in D \cap A} \frac{f(a)}{f(D)} \cdot r(a), \sum_{a \in D \cap A} \frac{f(a)}{f(D)} \cdot (r(a) - mp[f])^2 \Big) \qquad (f(a) > 0 \text{ iff } a \in D \cap A)$$

$$\geq \sum_{D \in \mathcal{B}(C^{\kappa})} f(D) \cdot \Big(\sum_{a \in D \cap A} \frac{f(a)}{f(D)} \cdot r(a), \sum_{a \in D \cap A} \frac{f(a)}{f(D)} \cdot (r(a) - \sum_{b \in D \cap A} \frac{f(b)}{f(D)} \cdot r(b))^2 \Big) \qquad \text{(Lemma 9)}$$

$$= \sum_{D \in \mathcal{B}(C^{\kappa})} f(D) \cdot (\mathbb{E}_D(mp), \mathbb{E}_D(lv)) \qquad \text{(Lemma 10, and } \tfrac{f(a)}{f(D)} = \lim_{i \to \infty} \tfrac{1}{i} I_a(A_i) \text{ almost surely)}$$

Here $\mathbb{E}_D(mp)$ and $\mathbb{E}_D(lv)$ denote the expected mean payoff and the expected local variance, resp., on almost all runs of either C^{κ} or $C^{\kappa'}$ initiated in any state of D; note that almost all such runs have the same mean payoff and the local variance due to the Ergodic theorem for Markov chains (see, e.g., [21, Theorem 1.10.2]). The last equality follows from Lemma 10 and the fact that $f(a)/f(D) = \lim_{i \to \infty} \frac{1}{i} I_a(A_i)$ on almost all runs initiated in D. We obtain that $p \cdot (mp[f], lv[f]) + (1-p) \cdot (mp[f'], lv[f'])$ is equal to

$$\sum_{D \in \mathcal{B}(C^{\kappa})} p \cdot f(D) \cdot (\mathbb{E}_D(mp), \mathbb{E}_D(lv)) + \sum_{D \in \mathcal{B}(C^{\kappa'})} (1-p) \cdot f'(D) \cdot (\mathbb{E}_D(mp), \mathbb{E}_D(lv)) \tag{21}$$

and by Lemma 8, there are two components $D, D' \in \mathcal{B}(C^{\kappa}) \cup \mathcal{B}(C^{\kappa'})$ and $0 \leq d \leq 1$ such that

$$(21) \quad \geq \quad d \cdot (\mathbb{E}_D(mp), \mathbb{E}_D(lv)) + (1-d) \cdot (\mathbb{E}_{D'}(mp), \mathbb{E}_{D'}(lv))$$

It is now straightforward to take as π_1 and π_2 the strategies whose only BSCCs are D and D', respectively. □

Now we can finish the proof of Theorem 2 by showing how a non-negative solution yields a required strategy. First, because (8) and (9) are satisfied, we can construct a finite-memory strategy σ such that $\mathbb{P}^{\sigma}_{s_{in}}[R_C] = \sum_{s \in C} y_{a,1} + y_{a,2}$; this can be done using the same steps as the construction of the strategy ϱ in the proof of Item 2 of Theorem 1 (note in particular that (8) and (9) are the same as (1) and (2), except for the variables y_t for $t \in S$ being "split" in two variables). Further, the solutions to (10) and (11) immediately give us frequency functions $f_{C,i}$ for all MECs C and $i \in \{1,2\}$, and these functions satisfy $mp[f_{C,i}] = z_{mp,C,i}$ and $lv[f_{C,i}] = z_{lv,C,i}$ by (12) and (13). We use Lemma 11 to obtain strategies π_C and π'_C, and a number d_C for each MEC C from $f_{C,1}$, $f_{C,2}$ and the number $\beta_C := (\sum_{s \in C \cap S} y_{s,1}) / (\sum_{s \in C \cap S} \sum_{i \in \{1,2\}} y_{s,i})$, and then Lemma 3 to combine σ and all π_C and π'_C into the resulting strategy ζ. We get

$$\mathbb{E}^{\zeta}_s[mp] = \sum_{C \in \mathcal{M}(G)} \mathbb{P}^{\zeta}_{s_{in}}[R_C] \cdot (\mathbb{E}^{\zeta}_{s_{in}}[mp|R_C], \mathbb{E}^{\zeta}_{s_{in}}[lv|R_C]) \qquad \text{(law of tot. expect.)}$$

$$= \sum_{C \in \mathcal{M}(G)} \mathbb{P}^{\sigma}_{s_{in}}[R_C] \cdot \big(d_C \cdot (\mathbb{E}^{\pi_C}_{s_{in}}[mp], \mathbb{E}^{\pi_C}_{s_{in}}[lv]) + (1-d_C) \cdot (\mathbb{E}^{\pi'_C}_{s_{in}}[mp], \mathbb{E}^{\pi'_C}_{s_{in}}[lv]) \big) \qquad \text{(Lemma 3 and def. of } \zeta)$$

$$\leq \sum_{C \in \mathcal{M}(G)} \Big(\sum_{\substack{s \in C \cap S \\ i \in \{1,2\}}} y_{s,i} \Big) \cdot \big(\beta_C \cdot (mp[f_{C,1}], lv[f_{C,1}]) + (1-\beta_C) \cdot (mp[f_{C,2}], lv[f_{C,2}]) \big) \qquad \text{(def. of } \pi_C, \pi'_C \text{ and } d_C)$$

$$= \sum_{C \in \mathcal{M}(G)} \sum_{i \in \{1,2\}} \Big(\sum_{s \in C \cap S} y_{s,i} \Big) \cdot (mp[f_{C,i}], lv[f_{C,i}]) \qquad \text{(def. of } \beta_C \text{ and arithmetic opers.)}$$

$$= \sum_{C \in \mathcal{M}(G)} \sum_{i \in \{1,2\}} \Big(\sum_{s \in C \cap S} y_{s,i} \Big) \cdot (z_{mp,C,i}, z_{lv,C,i}) \qquad \text{(def. of } f_{C,i})$$

$$\leq (u, v) \qquad \text{(by (14) and (15))}$$

4.3. Complexity

Let us now turn to complexity-theoretic questions. We will show that in fact the local-variance problem is in NP. We prove this by showing that any combination of two frequency functions can be achieved as a combination of two memoryless deterministic strategies, each yielding one BSCC. We first prove the following claim.

Claim 2. *Let C be a MEC and μ a memoryless randomized strategy generating a single BSCC. There are* memoryless deterministic *strategies χ_1, χ_2 in C, each generating a single BSCC, and a number $0 \leq q \leq 1$ such that for all $s \in C \cap S$*

$$(\mathbb{E}_s^{\mu}[mp], \mathbb{E}_s^{\mu}[lv]) \geq q \cdot (\mathbb{E}_s^{\chi_1}[mp], \mathbb{E}_s^{\chi_1}[lv]) + (1-q) \cdot (\mathbb{E}_s^{\chi_2}[mp], \mathbb{E}_s^{\chi_2}[lv])$$

Proof. By [12, Chapter 7, Theorem 2], $\mathbb{E}_s^{\mu}\left[mp^{l_a}\right]$ is equal to a convex combination of the values $\mathbb{E}_s^{\iota_i}\left[mp^{l_a}\right]$ for some memoryless deterministic strategies $\iota_1, \ldots, \iota_m$, i.e., there are $\gamma_1, \ldots, \gamma_m > 0$ such that $\sum_{i=1}^{m} \gamma_i = 1$ and $\sum_{i=1}^{m} \gamma_i \cdot \mathbb{E}_s^{\iota_i}\left[mp^{l_a}\right] = \mathbb{E}_s^{\mu}\left[mp^{l_a}\right]$. For all $1 \leq i \leq m$ and $D \in \mathcal{B}(C^{\iota_i})$ denote $\iota_{i,D}$ a memoryless deterministic strategy such that $\iota_{i,D}(s) = \iota_i(s)$ on all $s \in D \cap S$, and on other states $\iota_{i,D}$ is defined so that $D \cap S$ is reached with probability 1, independent of the starting state. For all $a \in D \cap A$ we have $\mathbb{E}_s^{\iota_{i,D}}\left[mp^{l_a}\right] = \mathbb{P}_s^{\iota_i}[Reach(D \cap S)] \cdot \mathbb{E}_s^{\iota_i}\left[mp^{l_a}\right]$, while for $a \notin D \cap A$ we have $\mathbb{E}_s^{\iota_{i,D}}\left[mp^{l_a}\right] = 0$. Hence

$$\sum_{i=1}^{m} \sum_{D \in \mathcal{B}(C^{\iota_i})} \gamma_i \cdot \mathbb{P}_s^{\iota_i}[Reach(D \cap S)] \cdot \mathbb{E}_s^{\iota_{i,D}}\left[mp^{l_a}\right] = \mathbb{E}_s^{\mu}\left[mp^{l_a}\right]$$

Since $\sum_{i=1}^{m} \sum_{D \in \mathcal{B}(C^{\iota_i})} \gamma_i \cdot \mathbb{P}_s^{\iota_i}[Reach(D \cap S)] = 1$, we apply Lemma 8 and get two memoryless deterministic single-BSCC strategies χ_1, χ_2 and $0 \leq q \leq 1$ such that

$$\mathbb{E}_s^{\mu}\left[mp^{l_a}\right] = q \cdot \mathbb{E}_s^{\chi_1}\left[mp^{l_a}\right] + (1-q) \cdot \mathbb{E}_s^{\chi_2}\left[mp^{l_a}\right]$$

which together with Lemma 10 implies that

$$\begin{aligned}
\mathbb{E}_s^{\mu}[mp] &= \sum_{a \in A} r(a) \cdot \mathbb{E}_s^{\mu}\left[mp^{l_a}\right] \\
&= \sum_{a \in A} r(a) \cdot \left(q \cdot \mathbb{E}_s^{\chi_1}\left[mp^{l_a}\right] + (1-q) \cdot \mathbb{E}_s^{\chi_2}\left[mp^{l_a}\right] \right) \\
&= q \cdot \sum_{a \in A} r(a) \cdot \mathbb{E}_s^{\chi_1}\left[mp^{l_a}\right] + (1-q) \cdot \sum_{a \in A} r(a) \cdot \mathbb{E}_s^{\chi_2}\left[mp^{l_a}\right] \\
&= q \cdot \mathbb{E}_s^{\chi_1}[mp] + (1-q) \cdot \mathbb{E}_s^{\chi_2}[mp]
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{E}_s^{\mu}[lv] &= \sum_{a \in A} (r(a) - \mathbb{E}_s^{\mu}[mp])^2 \cdot \mathbb{E}_s^{\mu}\left[mp^{l_a}\right] \\
&= \sum_{a \in A} (r(a) - \mathbb{E}_s^{\mu}[mp])^2 \cdot (q \cdot \mathbb{E}_s^{\chi_1}\left[mp^{l_a}\right] + (1-q) \cdot \mathbb{E}_s^{\chi_2}\left[mp^{l_a}\right]) \\
&= q \cdot \sum_{a \in A} (r(a) - \mathbb{E}_s^{\mu}[mp])^2 \cdot \mathbb{E}_s^{\chi_1}\left[mp^{l_a}\right] \\
&\quad + (1-q) \cdot \sum_{a \in A} (r(a) - \mathbb{E}_s^{\mu}[mp])^2 \cdot \mathbb{E}_s^{\chi_2}\left[mp^{l_a}\right] \\
&\geq q \cdot \sum_{a \in A} (r(a) - \mathbb{E}_s^{\chi_1}[mp])^2 \cdot \mathbb{E}_s^{\chi_1}\left[mp^{l_a}\right] \\
&\quad + (1-q) \cdot \sum_{a \in A} (r(a) - \mathbb{E}_s^{\chi_2}[mp])^2 \cdot \mathbb{E}_s^{\chi_2}\left[mp^{l_a}\right] \\
&= q \cdot \mathbb{E}_s^{\chi_1}[lv] + (1-q) \cdot \mathbb{E}_s^{\chi_2}[lv]
\end{aligned}$$

Here, the first and the last equality follow since μ, χ_1, and χ_2 have a single BSCC and so almost all runs have the same mean payoff. The inequality follows from Lemma 9. □

The following lemma strengthens Lemma 11 by showing that any two frequency functions can be "mimicked" by two memoryless deterministic strategies to achieve the required expected mean payoff and expected local variance.

Lemma 12. *Let C be a MEC, f and f' be two frequency functions, and $p \in [0,1]$ a number. Then there are two memoryless* deterministic *strategies π and π', each yielding a single BSCC, and a number $0 \leq h \leq 1$ such that for all $s \in C \cap S$ we have*

$$h \cdot (\mathbb{E}_s^{\pi}[mp], \mathbb{E}_s^{\pi}[lv]) + (1-h) \cdot (\mathbb{E}_s^{\pi'}[mp], \mathbb{E}_s^{\pi'}[lv]) \leq p \cdot (mp[f], lv[f]) + (1-p) \cdot (mp[f'], lv[f'])$$

Proof. In what follows we use the following definition: Let ν be a memoryless randomized strategy on a MEC C and let K be a BSCC of C^{ν}. We say that a strategy μ_K is *induced* by K if

1. $\mu_K(s)(a) = \nu(s)(a)$ for all $s \in K \cap S$ and $a \in K \cap A$
2. in all $s \in S \smallsetminus (K \cap S)$ the strategy μ_K corresponds to a memoryless deterministic strategy which reaches a state of K with probability one

Note that the above definition is independent of the strategy ν and only depends on the BSCC K.

We first apply Lemma 11 and obtain from f, f' and p two memoryless randomized strategies μ, μ' (each yielding a single BSCC) and a number d such that

$$d \cdot (\mathbb{E}_s^{\mu}[mp], \mathbb{E}_s^{\mu}[lv]) + (1-d) \cdot (\mathbb{E}_s^{\mu'}[mp], \mathbb{E}_s^{\mu'}[lv]) \leq p \cdot (mp[f], lv[f]) + (1-p) \cdot (mp[f'], lv[f'])$$

Now we show that these strategies may be even deterministic. By Claim 2,

$$\begin{aligned} &d \cdot (\mathbb{E}_s^{\mu}[mp], \mathbb{E}_s^{\mu}[lv]) + (1-d) \cdot (\mathbb{E}_s^{\mu'}[mp], \mathbb{E}_s^{\mu'}[lv]) \\ &\quad \geq d \cdot q \cdot (\mathbb{E}_s^{\chi_1}[mp], \mathbb{E}_s^{\chi_1}[lv]) + d \cdot (1-q) \cdot (\mathbb{E}_s^{\chi_2}[mp], \mathbb{E}_s^{\chi_2}[lv]) \\ &\qquad + (1-d) \cdot q' \cdot (\mathbb{E}_s^{\chi_1'}[mp], \mathbb{E}_s^{\chi_1'}[lv]) + (1-d) \cdot (1-q') \cdot (\mathbb{E}_s^{\chi_2'}[mp], \mathbb{E}_s^{\chi_2'}[lv]) \end{aligned}$$

for some memoryless deterministic strategies χ_1, χ_2, χ_1', and χ_2' and numbers $0 \leq q, q' \leq 1$. Subsequently, by Lemma 8, there are $\pi, \pi' \in \{\chi_1, \chi_2, \chi_1', \chi_2'\}$ and a number h such that

$$d \cdot (\mathbb{E}_s^{\mu}[mp], \mathbb{E}_s^{\mu}[lv]) + (1-d) \cdot (\mathbb{E}_s^{\mu'}[mp], \mathbb{E}_s^{\mu'}[lv]) \geq h \cdot (\mathbb{E}_s^{\pi}[mp], \mathbb{E}_s^{\pi}[lv]) + (1-h) \cdot (\mathbb{E}_s^{\pi'}[mp], \mathbb{E}_s^{\pi'}[lv])$$

This finishes the proof. □

The above lemma allows us to design a nondeterministic polynomial time algorithm as follows. Note that for any non-negative solution to the constrained problem in Fig. 4 the solution to variables $x_{a,1}$ and $x_{a,2}$ for any MEC C gives us two frequency functions f_1^C and f_2^C given by $f_i^C(a) = x_{a,i}$ for all a and i. By the above lemma, there are two memoryless deterministic strategies π_1^C and π_2^C and a number q^C for each MEC C such that, for any $s \in C \cap S$:

$$q^C \cdot \mathbb{E}_s^{\pi_1^C}[mp] + (1-q^C) \cdot \mathbb{E}_s^{\pi_2^C}[mp] \leq p^C \cdot mp[f_1^C] + (1-p^C) \cdot mp[f_2^C] \tag{22}$$

$$q^C \cdot \mathbb{E}_s^{\pi_1^C}[lv] + (1-q^C) \cdot \mathbb{E}_s^{\pi_2^C}[lv] \leq p^C \cdot lv[f_1^C] + (1-p^C) \cdot lv[f_2^C] \tag{23}$$

where $p^C = (\sum_{t \in C \cap S} y_{t,1}) / (\sum_{t \in C \cap S} \sum_{i \in \{1,2\}} y_{t,i})$, and because

$$\begin{aligned} &\Big(\sum_{t \in C \cap S} \sum_{i \in \{1,2\}} y_{y,i}\Big) \cdot \big(p^C \cdot (mp[f_1^C], lv[f_1^c]) + (1-p^C) \cdot (mp[f_2^C], lv[f_2^C])\big) \\ &\quad = \Big(\sum_{t \in C \cap S} y_{t,1}\Big) \cdot (mp[f_1^C], lv[f_1^c]) + \Big(\sum_{t \in C \cap S} y_{t,2}\Big) \cdot (mp[f_2^C], lv[f_2^C]) && \text{(by arithmetic operations)} \\ &\quad = \Big(\sum_{t \in C \cap S} y_{a,1}\Big) \cdot (z_{mp,C,1}, z_{lv,C,1}) + \Big(\sum_{t \in C \cap S} y_{t,2}\Big) \cdot (z_{mp,C,2}, z_{lv,C,2}) && \text{(by def. of } f_1^C \text{ and } f_2^C) \\ &\quad = \sum_{t \in C \cap S} \sum_{i \in \{1,2\}} y_{t,i} \cdot (z_{mp,C,i}, z_{lv,C,i}) && \text{(rearranging)} \end{aligned}$$

we get

$$\sum_{C\in\mathcal{M}(G)}\Big(\sum_{\substack{t\in C\cap S\\ i\in\{1,2\}}} y_{t,i}\Big)\cdot\big(q^C\cdot(\mathbb{E}_s^{\pi_1^C}[mp],\mathbb{E}_s^{\pi_1^C}[lv])+(1-q^C)\cdot(\mathbb{E}_s^{\pi_2^C}[mp],\mathbb{E}_s^{\pi_2^C}[lv])\big)$$

$$\leq \sum_{C\in\mathcal{M}(G)}\sum_{t\in C\cap S}\sum_{i\in\{1,2\}} y_{t,i}\cdot(z_{mp,C,i},z_{lv,C,i}) \qquad \text{(by (22) and (23), and above inequality)}$$

$$\leq (u,v) \qquad \text{(by (14) and (15))}$$

Hence, if there is a solution to the constraints from Fig. 4, then from (10) and (11) we get frequency functions f_C and f'_C for each MEC C. Using Lemma 12 we obtain two memoryless deterministic strategies π_1^C and π_2^C and a number q^C satisfying the conditions of the lemma, for each MEC C. The strategies π_1^C and π_2^C induce another non-negative solution, bearing the symbol $\hat{}$, to the constraints from Fig. 4, in which the y_a are preserved from the old solution, $\hat{y}_{s,1}=q^C/(y_{s,1}+y_{s,2})$ and $\hat{y}_{s,2}=(1-q^C)\cdot(y_{s,1}+y_{s,2})$ where C is the MEC containing a, $\hat{x}_{a,i}=f_{\pi_i}(a)$ for all a and $i\in\{1,2\}$, and $\hat{z}_{mp,C,i}$ and $\hat{z}_{lv,C,i}$ are uniquely determined by the values $\hat{x}_{a,i}$. Then (10)–(15) are satisfied by the properties of π_1^C and π_2^C, and the satisfaction of (8) and (9) is preserved because $\hat{y}_{s,1}+\hat{y}_{s,2}=y_{s,1}+y_{s,2}$.

Now note that the values of z and x are only dependent on π_1^C and π_2^C, and once they are fixed, the remaining constraints reduce to a linear program. Hence, to solve the local variance problem in nondeterministic polynomial time it is sufficient to guess the memoryless strategies π_1^C and π_2^C, compute the corresponding values to z and x, construct the linear program by substituting these values into L_{loc}, and by solving the linear program verify that our guess of the memoryless deterministic strategies was correct. Hence, we get the following corollary.

Corollary 4. *The problem of deciding if there is a strategy ζ satisfying the condition $(\mathbb{E}_{s_{in}}^{\zeta}[mp],\mathbb{E}_{s_{in}}^{\zeta}[lv])\leq(u,v)$ is in NP.*

A simple adaptation of our proof also allows us to give an upper bound on the memory needed by a strategy.

Corollary 5. *If there is a strategy ζ satisfying $(\mathbb{E}_{s_{in}}^{\zeta}[mp],\mathbb{E}_{s_{in}}^{\zeta}[lv])\leq(u,v)$ then there is a 3-memory strategy with the same properties.*

Proof. The proof is a straightforward modification of the proof of direction $\Leftarrow$ of Theorem 2 from the end of Subsection 4.2. The only difference is that instead of using Lemma 11 to obtain two memoryless randomized strategies for all MECs, we use Lemma 12 to obtain two memoryless deterministic strategies π_C and π'_C for each MEC C. We then combine all π_C (resp. all π'_C) to obtain a strategy π (resp. π') which in every MEC C behaves as π_C (resp. π'_C), and use Lemma 3 to combine σ, π and π' into the resulting strategy. □

Corollary 4 and Remark 1 give the following corollary.

Corollary 6. *The approximate Pareto curve for local variance can be computed in exponential time.*

5. Zero variance with optimal performance

Now we present polynomial-time algorithms to compute the optimal mean payoff expectation that can be ensured along with zero variance. The results are captured in the following theorem.

Theorem 3. *The minimal expectation that can be ensured with zero variance can be computed in polynomial time, both for global and local variance.*

We prove the theorem in the following two subsections.

5.1. Global variance

The basic intuition for zero global variance is that we need to find the minimal number y such that there is an almost-sure winning strategy to reach the MECs where expectation *exactly* y can be ensured with zero variance. Relying on Lemma 4, we then get that for each MEC such values are described by an interval, which allows us to reduce the problem to a series of (polynomially many) almost-sure reachability problems. The complete algorithm is given in Algorithm 1. The correctness is proved in the following lemma.

Lemma 13. *Given an MDP $G=(S,A,Act,\delta)$, a starting state s_{in}, and a reward function r, the following assertions hold:*

1. *If x is the output of Algorithm 1, then there is a strategy to ensure that the expectation is at most x and the global variance is zero.*
2. *If there is a strategy to ensure that the expectation u and the global variance is zero, then the output x of Algorithm 1 satisfies that $x\leq u$.*

Input: An MDP $G = (S, A, Act, \delta)$, a starting state s_{in}, and a reward function r
Output: An optimal reward value or NO
1 Compute the MEC decomposition $C_1, C_2, \ldots, C_n$ of G;
2 **foreach** C_i **do**
3 $\quad \alpha_{C_i} := \inf_\sigma \min_{s \in C_i} \mathbb{E}_s^\sigma[mp]$; // minimal expectation in C_i
4 $\quad \beta_{C_i} := \sup_\sigma \max_{s \in C_i} \mathbb{E}_s^\sigma[mp]$; // maximal expectation in C_i
5 Sort the values α_{C_i} in a non-decreasing order $\ell_1 \le \ell_2 \le \ldots \le \ell_n$;
6 $i := 1$;
7 **while** $i \le n$ **do**
8 $\quad \mathcal{C}_i := \{C_j \mid \alpha_{C_j} \le \ell_i \le \beta_{C_j}\}$; // MECs whose interval contain ℓ_i
9 $\quad A_i := \bigcup_{C_j \in \mathcal{C}_i} C_j$; // the union of the MECs in $\mathcal{C}_i$
10 $\quad$ **if** $\exists \sigma : \mathbb{P}_{s_{in}}^\sigma[Reach(A_i)] = 1$ **then**
11 $\quad\quad$ **return** ℓ_i;
12 $\quad i := i + 1$;
13 **return** NO

Algorithm 1: Zero global variance.

Proof. The proof of the first item proceeds as follows. If the output of the algorithm is x, then consider $\mathcal{C}$ to be the set of MECs whose interval contains x. Let $A' = \bigcup_{C_j \in \mathcal{C}} C_j$. By line 10 of the algorithm we have that there exists an almost-sure winning strategy for the objective $Reach(A')$, and by using standard algorithms for MDPs [23,15] there exists a memoryless deterministic almost-sure winning strategy σ_R for the objective. We consider a strategy defined as follows: (i) play σ_R until an end-component in $\mathcal{C}$ is reached; (ii) once A' is reached, consider a MEC C_j that is reached and switch to the memoryless randomized strategy σ_x of Lemma 4 to ensure that every BSCC obtained in C_j by fixing σ_x has expected mean-payoff exactly x (i.e., it ensures expectation x with zero global variance). Since σ is an almost-sure winning strategy for the reachability objective to the MECs in $\mathcal{C}$, and once the MECs are reached the strategy σ_x ensures that every BSCC of the Markov chain has expectation exactly x, it follows that the expectation is x and the global variance is zero.

For the proof of the second item, suppose that there is a strategy to ensure that the expectation is u and the global variance zero. By Theorem 1 there is a finite-memory strategy σ with the same properties. Let $\widehat{\mathcal{C}} = \{\widehat{C} \mid \widehat{C}$ is a BSCC reachable from s in $G_{s_{in}}^\sigma\}$. Since the global variance is zero and the expectation is u, in every BSCC $\widehat{C} \in \widehat{\mathcal{C}}$ the expected mean payoff must be exactly u. Let

$$\mathcal{C} = \{C \mid C \text{ is a MEC and there exists } \widehat{C} \in \widehat{\mathcal{C}} \text{ such that the associated end component of } \widehat{C} \text{ is contained in } C\}.$$

For every $C \in \mathcal{C}$ we have $u \in [\alpha_C, \beta_C]$, where $[\alpha_C, \beta_C]$ is the interval of C. Moreover, the strategy σ is also a witness almost-sure winning strategy for the reachability objective $Reach(A')$, where $A' = \bigcup_{C \in \mathcal{C}} C$. Let $\alpha' = \max\{\alpha_C \mid C \in \mathcal{C}\}$. Since for every $C \in \mathcal{C}$ we have $u \in [\alpha_C, \beta_C]$, it follows that $\alpha' \le u$. Observe that if the algorithm checks the value α' (say $\alpha' = \ell_i$), then the condition at line 10 is true, as $A' \subseteq \bigcup_{C_j \in \mathcal{C}_i} C_j$ and σ will be a witness almost-sure winning strategy to reach $\bigcup_{C_j \in \mathcal{C}_i} C_j$. Thus the algorithm must return a value at most $\alpha' \le u$. □

The complexity analysis of Algorithm 1 is as follows: (i) the MEC decomposition at line 1 can be computed in polynomial time [7,6]; (ii) the minimal and maximal expectation at lines 3 and 4 can be computed in polynomial time, e.g. using linear programming [23]; and (iii) sorting (line 5) can be done in polynomial time, as well as deciding existence of almost-sure winning strategies for reachability objectives (line 10) can be achieved in polynomial time [5,8]. It follows that the algorithm runs in polynomial time, and we obtain Theorem 3 for global variance.

5.2. Local variance

Given a set of actions X, we denote by $Safe(X)$ the set of runs that never take any action outside of X. As the first step, our algorithm for computing optimal expectation under zero variance requires to compute, for all states s, a number $\gamma(s)$. The number $\gamma(s)$ is the minimum number for which there is a strategy σ_s that, when initiated in s, only visits actions with reward $\gamma(s)$. This corresponds to the minimal q for which there is a strategy σ_s such that $\mathbb{P}_s^{\sigma_s}[Safe(\{a \mid r(a) = q\}) = 1]$.

The intuition of the algorithm for zero local variance is that to minimize the expectation with zero local variance, a strategy σ needs to reach states s with low $\gamma(s)$, and then mimic σ_s. Moreover, σ minimizes the expected value of mp among all possible behaviours satisfying the above. For this purpose, we define an MDP $\overline{G}$ from G as follows: For each state s such that $\gamma(s) < \infty$ we add a state $\overline{s}$ with a self-loop on it, and we add a new action a_s that leads from s to $\overline{s}$. Further, we construct a reward function M^γ which assigns $\gamma(s)$ to a_s, and 0 to all other actions. Let $F = \{a_s \mid \gamma(s) < \infty\}$ be the target set of actions. We compute a strategy that minimizes the expected cumulative reward cr_{M^γ} (i.e., $\mathrm{cr}_{M^\gamma}(s_1 a_1 s_2 \ldots) = \sum_{j=1}^{\infty} r(a_j)$), and at the same time ensures almost-sure (probability 1) reachability to F in $\overline{G}$. Our proofs below show that this minimal expected cumulative reward is equal to the minimal expected mean payoff achievable under zero variance in G.

For reader's convenience, the algorithm that we intuitively described above is formally described as Algorithm 2. In the rest of this section we prove correctness of the algorithm, and analyse its complexity. The following lemma is straightforward.

Input: An MDP $G = (S, A, Act, \delta)$, a starting state s_{in}, and a reward function r
Output: An optimal reward value or NO

1 Sort the values $r(a)$ for $a \in A$ in an increasing order $q_1 < q_2 < \ldots < q_n$;
2 **foreach** $s \in S$ **do**
3 $\quad i := 1$;
4 $\quad \gamma(s) := \infty$;
5 $\quad$ **while** $i \leq n$ **do**
6 $\quad\quad A_i := \{a \mid r(a) = q_i\}$; // actions with reward q_i
7 $\quad\quad$ **if** $\exists \sigma$ *with* $\mathbb{P}^{\sigma}_{s}[Safe(A_i)] = 1$ **then**
8 $\quad\quad\quad \gamma(s) := q_i$;
9 $\quad\quad\quad$ **break**; // lowest solution found, exit while-loop
10 $\quad\quad i := i + 1$;
11 Construct $\overline{G}$ and M^{γ};
12 $x := \inf\{x \mid \exists \sigma \text{ in } \overline{G} : \mathbb{P}^{\sigma}_{s_{in}}[Reach(F)]{=}1 \wedge \mathbb{E}^{\sigma}_{s_{in}}[\mathsf{cr}_{M^{\gamma}}]{=}x\}$; // $\inf\emptyset{=}\infty$
13 **if** $x = \infty$ **then return** NO **else return** x;

Algorithm 2: Zero local variance.

Lemma 14. *The values $\gamma(s)$ computed by Algorithm 2 at lines 2–10 are the correct values of γ.*

Let u be the minimal expected mean payoff that can be ensured along with zero local variance in G, and let x be the value returned by Algorithm 2.

Lemma 15. *Given an MDP $G = (S, A, Act, \delta)$, a starting state s_{in}, and a reward function r, the following assertions hold:*

1. *If x is the output of Algorithm 2, then there is a strategy to ensure that the expectation is at most x and the local variance is zero.*
2. *If there is a strategy to ensure that the expectation u and the local variance is zero, then the output x of Algorithm 2 satisfies that $x \leq u$.*

Proof. For the first item, consider a finite-memory strategy $\overline{\sigma}$ in $\overline{G}$ that satisfies $\mathbb{P}^{\sigma}_{s_{in}}[Reach(F)]{=}1$ and $\mathbb{E}^{\sigma}_{s_{in}}[M^{\gamma}]{=}x$ (it exists due to line 7 of the algorithm). We construct a witness strategy σ for zero local variance in G as follows: play as $\overline{\sigma}$ until the set F is reached, and after F is reached, if a state $\overline{s'}$ is reached, switch to the memoryless deterministic strategy $\sigma_{s'}$ to ensure that no reward other than $\gamma(s')$ is visited. The strategy σ ensures that every BSCC of the resulting Markov chain consists of only one reward value. Hence the local variance is zero. We also have $\mathbb{E}^{\sigma}_{s_{in}}[mp] \leq x$, and the desired result follows.

Let us now proceed with the second item of the lemma. By Corollary 5 there is a finite-memory witness strategy σ with $\mathbb{E}^{\sigma}_{s_{in}}[mp] = u$ and $\mathbb{E}^{\sigma}_{s_{in}}[lv] = 0$. Consider the Markov chain $G^{\sigma}_{s_{in}}$, and its BSCC C. We establish the following properties:

1. Rewards of all actions in C must be the same. Otherwise the local variance is positive (by the definition of local variance and by Ergodic theorem for Markov chains [21, Theorem 1.10.2]).
2. For all states s' that appear in the BSCC we have $\gamma(s') \leq r_C$, where r_C is the reward of the actions in C. Otherwise if $\gamma(s') > r_C$, playing according the strategy σ in the BSCC from s' we ensure that only states with reward r_C are visited, contradicting properties of $\gamma(s)$.

It follows that in every BSCC C of the Markov chain the reward r_C of the BSCC satisfies that $r_C \geq \gamma(s')$, for every s' that appears in C. We construct a strategy $\overline{\sigma}$ in $\overline{G}$ as follows: the strategy plays as σ until a BSCC is reached, and as soon as a BSCC C is reached at state s', the strategy $\overline{\sigma}$ chooses the action $a_{s'}$ to proceed to the state $\overline{s'}$ securing reward $\gamma(s')$ on the run. The strategy $\overline{\sigma}$ ensures that the cumulative reward in $\overline{G}$ is at most u, and, because in a finite-state Markov chain a BSCC is almost surely reached, also $\mathbb{P}^{\overline{\sigma}}_{s_{in}}[Reach(F)] = 1$. By Lemma 14 and by examining lines 11–13 of Algorithm 2 we get $u \geq x$. □

Let us now analyse the complexity of Algorithm 2. Since safety properties for MDPs can be decided in polynomial time (see e.g. [15]), we obtain the computation of γ can be executed in polynomial time. Further, the property on line 12 can be checked in polynomial time using [16].

6. Conclusions and future work

We studied two notions of variance for MDPs with mean-payoff objectives: the global (the standard one) and the local variance. We established a strategy complexity (i.e., the memory and randomization required) for Pareto optimal strategies, and established results for complexity of the problems. For global variance, our results yield PSPACE upper bound for the decision problem, and pseudo-polynomial algorithm for approximation. For local variance, we gave an NP upper bound. We further showed that the problems of finding the optimal expected mean payoff achievable with zero variance can be solved in polynomial time for both global and local variance.

The main question which we left open is establishing tighter complexity bounds. There are several possible directions in achieving this. One might try to prove NP-hardness for either the local or global variance, but this appears to be a difficult task requiring novel insights, for the following reasons:

- There are NP-hardness results for variance-restricted cumulative reward [19], but these hold *already for zero variance*. As we have shown in Section 5, the corresponding problems for mean payoff are solvable in polynomial time. This suggest that the problems for cumulative reward and mean-payoff reward are substantially different.
- Hardness results of non-convex programming, such as [22,20], do not extend easily to our setting.
  - The proof of [22] encodes the clique problem in a quadratic program. There are two major obstacles to modifying the proof for our setting.
    Firstly, the encoding of [22] captures relatively complex relation between variables, expressing a combinatorial problem. In our encoding almost every variable is bound to a single state, and combinatorial dependencies is not easily expressible.
    Secondly, in the proof of [22] the intuition of the encoding is that a structure of a given graph is directly encoded in variables. Similar approach will fail in our setting, since, for example, any encoding "based on" strongly connected graph will trivially yield optimal solution with zero variance (see Lemma 4), for which an optimal satisfiable assignment can be found in polynomial time.
  - The proofs of [20] use objective functions and constraints which are clearly more complex than what encodings permit.
- For problems related to MDPs, NP-hardness proofs typically exploit combinatorial nature of the problem. For the global variance, the issue is quite orthogonal, as the complicated computational step is the guess of the number z (see page 156), since if z is given in polynomially many bits, then our algorithm for approximation is polynomial.

An alternative step towards establishing better complexity bounds is lowering the complexity of the global-variance problem. As we have remarked at the end of Section 3, it would be sufficient to obtain polynomial bound on the size of z (see page 156). Nevertheless, this appears to be a non-trivial step requiring new insights into the problem.

There are several interesting directions for future work. The first direction would be to close the computational complexity gaps in the problems we study. In this work, we introduce local variance as a measure of stability, which along side global variance capture different aspects of stability of a system. Investigating different notions of stability is another interesting direction for future work.

Acknowledgments

T. Brázdil and A. Kučera are supported by the Czech Science Foundation, grant No. 15-17564S. K. Chatterjee is supported by the Austrian Science Fund (FWF) Grant No. P 23499-N23; FWF NFN Grant No. S11407-N23 (RiSE); ERC Start grant (279307: Graph Games); Microsoft faculty fellows award. V. Forejt is supported by EPSRC project EP/M023656/1.

References

[1] E. Altman, Constrained Markov Decision Processes, Stoch. Model., Chapman & Hall/CRC, 1999.
[2] T. Brázdil, V. Brožek, K. Chatterjee, V. Forejt, A. Kučera, Markov decision processes with multiple long-run average objectives, Log. Methods Comput. Sci. 10 (4) (2014).
[3] T. Brázdil, K. Chatterjee, V. Forejt, A. Kučera, Trading performance for stability in Markov decision processes, in: 28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013, New Orleans, LA, USA, June 25–28, 2013, IEEE Computer Society, 2013, pp. 331–340.
[4] J. Canny, Some algebraic and geometric computations in PSPACE, in: Proceedings of STOC'88, ACM Press, 1988, pp. 460–467.
[5] K. Chatterjee, M. Henzinger, Faster and dynamic algorithms for maximal end-component decomposition and related graph problems in probabilistic verification, in: SODA, SIAM, 2011, pp. 1318–1336.
[6] K. Chatterjee, M. Henzinger, An $O(n^2)$ time algorithm for alternating Büchi games, in: SODA, SIAM, 2012, pp. 1386–1399.
[7] K. Chatterjee, M. Henzinger, Efficient and dynamic algorithms for alternating Büchi games and maximal end-component decomposition, J. ACM 61 (3) (2014).
[8] K. Chatterjee, M. Henzinger, M. Joglekar, N. Shah, Symbolic algorithms for qualitative analysis of Markov decision processes with Büchi objectives, Form. Methods Syst. Des. 42 (3) (2013) 301–327.
[9] K. Chatterjee, R. Majumdar, T. Henzinger, Markov decision processes with multiple objectives, in: Proceedings of STACS 2006, in: Lect. Notes Comput. Sci., vol. 3884, Springer, 2006, pp. 325–336.
[10] K.-J. Chung, Mean-variance tradeoffs in an undiscounted MDP: the unichain case, Oper. Res. 42 (1994) 184–188.
[11] C. Courcoubetis, M. Yannakakis, Markov decision processes and regular events, IEEE Trans. Autom. Control 43 (10) (1998) 1399–1418.
[12] C. Derman, Finite State Markovian Decision Processes, Math. Sci. Eng., Academic Press, 1970.
[13] K. Etessami, M. Kwiatkowska, M. Vardi, M. Yannakakis, Multi-objective model checking of Markov decision processes, Log. Methods Comput. Sci. 4 (4) (2008) 1–21.
[14] J.A. Filar, L. Kallenberg, H.-M. Lee, Variance-penalize Markov decision processes, Math. Oper. Res. 14 (1989) 147–161.
[15] V. Forejt, M. Kwiatkowska, G. Norman, D. Parker, Automated verification techniques for probabilistic systems, in: M. Bernardo, V. Issarny (Eds.), Formal Methods for Eternal Networked Software Systems (SFM'11), in: Lect. Notes Comput. Sci., vol. 6659, Springer, 2011, pp. 53–113.
[16] V. Forejt, M. Kwiatkowska, G. Norman, D. Parker, H. Qu, Quantitative multi-objective verification for probabilistic systems, in: Proceedings of TACAS 2011, 2011, pp. 112–127.
[17] V. Forejt, M. Kwiatkowska, D. Parker, Pareto curves for probabilistic model checking, in: Proc. of ATVA'12, in: Lect. Notes Comput. Sci., vol. 7561, Springer, 2012, pp. 317–332.

[18] R. Lyons, Strong laws of large numbers for weakly correlated random variables, Mich. Math. J. 35 (3) (1988) 353–359.
[19] S. Mannor, J. Tsitsiklis, Mean-variance optimization in Markov decision processes, in: Proceedings of ICML-11, ACM, New York, NY, USA, June 2011, pp. 177–184.
[20] K.G. Murty, S.N. Kabadi, Some np-complete problems in quadratic and nonlinear programming, Math. Program. 39 (2) (1987) 117–129.
[21] J. Norris, Markov Chains, Cambridge University Press, 1998.
[22] P.M. Pardalos, S.A. Vavasis, Quadratic programming with one negative eigenvalue is np-hard, J. Glob. Optim. 1 (1) (1991) 15–22.
[23] M. Puterman, Markov Decision Processes, Wiley, 1994.
[24] H.L. Royden, Real Analysis, 3rd edition, Macmillan, New York, 1988.
[25] M.J. Sobel, The variance of discounted MDP's, J. Appl. Probab. 19 (1982) 794–802.
[26] M.J. Sobel, Mean-variance tradeoffs in an undiscounted MDP, Oper. Res. 42 (1994) 175–183.
[27] S.A. Vavasis, Quadratic programming is in NP, Inf. Process. Lett. 36 (2) (1990) 73–77.
[28] S.A. Vavasis, Approximation algorithms for indefinite quadratic programming, Math. Program. 57 (2) (Nov. 1992) 279–311.