Department of Computer Science

Compositional Strategy Synthesis for Stochastic Games with Multiple Objectives

Nicolas Basset, Marta Kwiatkowska and Clemens Wiltsche

CS-RR-16-03

Department of Computer Science, University of Oxford
Wolfson Building, Parks Road, Oxford, OX1 3QD

Compositional Strategy Synthesis for Stochastic Games with Multiple Objectives

N. Basset[a], M. Kwiatkowska[a,*], C. Wiltsche[a]

[a]*Department of Computer Science, University of Oxford, United Kingdom*

Abstract

Design of autonomous systems is facilitated by automatic synthesis of controllers from formal models and specifications. We focus on stochastic games, which can model interaction with an adverse environment, as well as probabilistic behaviour arising from uncertainties. Our contribution is twofold. First, we study long-run specifications expressed as quantitative multi-dimensional mean-payoff and ratio objectives. We then develop an algorithm to synthesise ε-optimal strategies for conjunctions of almost sure satisfaction for mean payoffs and ratio rewards (in general games) and Boolean combinations of expected mean-payoffs (in controllable multi-chain games). Second, we propose a compositional framework, together with assume-guarantee rules, which enables winning strategies synthesised for individual components to be composed to a winning strategy for the composed game. The framework applies to a broad class of properties, which also include expected total rewards, and has been implemented in the software tool PRISM-games.

Contents

*Corresponding author
Email addresses: nicolas.basset@cs.ox.ac.uk (N. Basset), Marta.Kwiatkowska@cs.ox.ac.uk (M. Kwiatkowska), clemens.wiltsche@cs.ox.ac.uk (C. Wiltsche)

1. Introduction

Game theory has found versatile applications in the past decades, in areas ranging from artificial intelligence, through modelling and analysis of financial markets, to control system design and verification. The game model consists of an arena with a number of positions and two or more players that move a token between positions, sometimes called *games on graphs* [26]. The rules of the game determine the allowed moves between positions, and a player's winning condition captures which positions or sequences of positions are desirable for the player. When a player decides on a move, but the next position is determined by a probability distribution, we speak of a *stochastic game* [53]. Since stochastic games can model probabilistic behaviour, they are particularly attractive for the analysis of systems that naturally exhibit uncertainty.

In this article we focus our attention on the development of correct-by-construction controllers for autonomous systems via the synthesis of strategies that are *winning* for turn-based zero-sum stochastic games. When designing autonomous systems, often a critical element is the presence of an uncertain and adverse environment. The controllable parts are modelled as Player $\Diamond$, for which we want to find a strategy, while the non-cooperative behaviour of the environment is modelled as Player $\Box$. Modelling that Player $\Box$ tries to spoil winning for Player $\Diamond$ expresses that we do not make any assumptions on the environment, and hence a winning strategy for Player $\Diamond$ has to be winning against all possible behaviours of the environment. We take the view that stochasticity models uncertain behaviour where we know the prior distribution, while nondeterminism models the situation where all options are available to the other player.

In addition to probabilities, one can also annotate the model with *rewards* to evaluate various quantities, for example profit or energy usage, by means of expectations. Often, not just a single objective is under consideration, but several, potentially conflicting, objectives must be satisfied, for example maximising both throughput and latency of a network. In our previous work [19, 20], we studied *multi-objective* expected total reward properties for stochastic games with certain terminating conditions. Expected total rewards, however, are unable to express *long-run average* (also called mean-payoff) properties. Another important class of properties are *ratio* rewards [58], with which one can state, e.g., speed (distance per time unit) or fuel efficiency (distance per unit of fuel). In this paper we extend the repertoire of reward properties for stochastic games by considering winning conditions based on long-run average and ratio rewards, both for expectation and almost sure satisfaction semantics. These can be expressed as single or multi-objective properties with upper or lower *thresholds* on the expected target reward to be achieved, for example "the average energy consumption does not exceed 100 units per hour almost surely", or "the expected number of passengers transported is at least 100 per hour, while simultaneously ensuring that the expected fuel consumption is at most 50 units per hour". Multi-objective properties allow us to explore trade-offs between objectives by analysing the Pareto curve. The difficulty with multi-objective strategy synthesis compared to verification is that the objectives cannot be considered in isolation, but the synthesised Player $\Diamond$ strategy has to satisfy all simultaneously. Another issue is that monolithic strategy synthesis may be computationally infeasible, as a consequence of algorithmic complexity bounds [16, 20].

We thus formulate a *compositional* framework for strategy synthesis, which allows us to derive a strategy for the composed system by synthesising only for the (smaller) individual components; see e.g. [13] for an approach for non-stochastic systems. To this end we introduce a game composition operation ($\|$), which is closely related to that of probabilistic automata (PAs) in the sense of [52]. PAs correspond to stochastic games with only one player present, and can be used (i) for *verification*, to check whether *all* behaviours satisfy a specification (when only Player $\Box$ is present), and (ii) for *strategy synthesis*, to check whether there *exists* a strategy giving rise to behaviors satisfying a specification (when only Player $\Diamond$ is present) [37]. In verification, the nondeterminism that is present in the PA models an adverse, uncontrollable, environment. By applying a Player $\Diamond$ strategy to a game to resolve the controllable nondeterminism, we are left with a PA where only uncontrollable nondeterminism for Player $\Box$ remains. This observation allows us to reuse rules for compositional PA verification, such as those in [36], to derive synthesis rules for games. Similarly to [36], which employs multi-objective property specifications to achieve compositional verification of PAs, multi-objective properties are crucial for compositional strategy synthesis, as elaborated below.

In our framework, we assume that the designer provides games $\mathcal{G}^1, \mathcal{G}^2, \ldots$ representing components of a larger system, which is modelled as their composition $\mathcal{G} = \mathcal{G}^1 \parallel \mathcal{G}^2 \parallel \cdots$. By giving a local specification φ^i for each component game $\mathcal{G}^i$, we deduce global specifications φ for the composed game $\mathcal{G}$, so that, given local strategies π^i achieving the respective specifications φ^i, the global specification φ is satisfied in $\mathcal{G}$ by applying the local strategies. We deduce the global specifications independently of the synthesised strategies, by instead deducing the global specification φ from the local specifications φ^i using compositional verification rules, that is, rules for systems without controllable nondeterminism (such as PAs) to determine whether φ holds for all strategies given that, for each component $\mathcal{G}^i$, φ^i holds for all strategies. In Theorem 15 we show that, whenever there is a PA verification rule deducing φ from φ^i, then there is a corresponding synthesis rule for games, justifying the use of local strategies for φ^i in the composed game $\mathcal{G}$ to achieve φ.

The compositional synthesis problem is thus reduced to finding the local strategies π^i achieving φ^i, which is the classical monolithic strategy synthesis question from (quantitative) objectives that are compatible with the composition rules. By allowing general Boolean combinations of objectives, we can, for example, synthesise for one component a strategy satisfying an objective φ^A, and for a second component a strategy that satisfies an objective φ^G under the assumption φ^A, that is, the implication $\varphi^A \to \varphi^G$, so that the global specification that these strategies satisfy is φ^G.

Contributions. The paper makes the following main contributions.

- **Section 3:** We show that the strategy synthesis problem for conjunctions of almost sure mean payoffs, which maintain several mean payoffs almost surely above the corresponding thresholds, is in co-NP (Corollary 2) and present a synthesis algorithm for ε-optimal strategies (Theorem 7).
- **Section 4:** For expectation objectives, we show how to reduce synthesis problems for Boolean combinations to those for conjunctions (Theorem 8), which allows us to obtain ε-optimal strategies for Boolean combinations of expected mean-payoff objectives (Theorem 14) in a general class of *controllable multi-chain (CM) games* that we introduce.
- **Section 5:** We develop a composition of stochastic games that synchronises on actions, together with composition rules that allow winning strategies synthesised for individual components to be composed to a winning strategy for the composed game (Theorem 15).

Previous Work. Preliminary versions of this work appeared as [4] for synthesis of ε-optimal strategies for multi-objective mean payoff, and as [5] for the compositional framework. We additionally draw inspiration for Boolean combinations from [19]. By introducing controllable multichain games, we can synthesise Boolean combinations of long-run objectives, which allows more general assume-guarantee rules than in [5, 4]. Further, due to our decision procedure, we can present the semi-algorithm of [4] as an algorithm.

The techniques presented here have been implemented in the tool PRISM-games 2.0 [38], a new release of PRISM-games [17]. The implementation supports compositional assume-guarantee synthesis for long-run properties studied here, as well as total expected reward properties of [19, 20]. PRISM-games has been employed to analyse several case studies in autonomous transport and energy management, see e.g. [60, 55, 39] and references therein.

1.1. Related Work

Multi-objective strategy synthesis. Our work generalises multi-objective strategy synthesis for MDPs by introducing nondeterminism arising from an adversarial environment. Previous research on multi-objective synthesis for MDPs discusses PCTL [3], total discounted and undiscounted expected rewards [59, 15, 29], ω-regular properties [27], expected and threshold satisfaction of mean-payoffs [6, 14], percentile satisfaction of mean-payoffs [48, 14], as well as conditional expectations for total rewards [1]; recent work on mixing types of objectives appeared in [10, 47].

In contrast to the case for MDPs, synthesis for games needs to take into account the uncontrollable Player $\Box$. For non-stochastic games, multi-objective synthesis has recently been discussed in the context of mean-payoff and energy games [8, 11], for mean-payoffs and parity conditions [16, 9], and robust mean-payoffs [57]. Non-zero-sum games in the context of assume-guarantee synthesis arise in [13]. For stochastic games, PCTL objectives are the subject of [7]. The special case of precisely achieving a total expected reward is discussed in [18], which is extended to Boolean combinations and LTL specifications for stopping games in [19, 20, 55]. Under stationary strategies and recurrence assumptions on the game, [54] approximate mean-payoff conjunctions. Non-zero-sum stochastic games for more than two players, where each player has a single discounted expected total reward objective, are discussed in [41].

Stochastic games with shift-invariant objectives. To formulate our decision problem for almost-sure satisfaction of conjunctions of mean-payoff objectives (Corollary 2), we rephrase this multi-objective property in terms of shift-invariant winning condition studied in [32] and [33]. These papers state general properties about qualitative determinacy (there is always a winner) and half-positionality (one player needs only memoryless deterministic strategies) for a general class of games, in which the winning condition (possibly multi-objective) is shift-invariant. [32] also consider the problem of satisfaction probability being above an arbitrary given threshold, which is more general than the problem of almost-sure satisfaction considered here. In fact, [32] explain how to solve the former problem using an oracle for the latter, but were not concerned with synthesis nor ε-optimal winning strategies. We believe that ideas could be borrowed from [32] to extend our synthesis algorithm from almost sure satisfaction to arbitrary threshold satisfaction.

Compositional modelling and synthesis. Our compositional framework requires a notion of parallel composition of components, so that composing winning strategies of the components yields a winning strategy for the composition. Several notions of parallel composition of non-stochastic games have been proposed, for example [31], but player identity is not preserved in the composition. In [30] the strategies of the components have to agree in order for the composed game not to deadlock. Similarly, the synchronised compositions in [42] and [43] require the local strategies to ensure that the composition never deadlocks.

Composition of probabilistic systems is studied for PAs in [52], where, however, no notion of players exists. Compositional approaches that distinguish between controllable and uncontrollable events include [25] and probabilistic input/output automata (PIOA) [21]. However, when synthesising strategies concurrent games have to be considered, as there is no partitioning of states between players. In contrast, we work with turn-based games and define a composition that synchronises on actions, similarly to that for PAs [52]. This is reminiscent of single-threaded interface automata (STIA) [24] that enforce a partition between *running* and *waiting* states, which we here interpret as Player $\Diamond$ and Player $\Box$ respectively.

The problem of synthesising systems from components whose composition according to a fixed architecture satisfies a given global LTL specification is undecidable [44]. Strategies in the components need to accumulate sufficient knowledge in order to make choices that are consistent globally, while only being able to view the local history, as discussed in [35]. In our setting, each strategy is synthesised on a single component, considering all other components as black boxes, and hence adversarial. Assume-guarantee synthesis is a convenient way of encoding assumptions on other components and the overall environment in the local specifications; see [13] for a formulation as non-zero-sum non-stochastic games.

2. Preliminaries

In this section we introduce notations and definitions for stochastic games, their strategies and winning conditions. We work with two representations of strategies, (standard) deterministic update and stochastic update of [6], and prove that they are equally powerful if their memory size is not restricted. We then define strategy application and discuss behaviour of stochastic games under strategies. In particular, we define the induced probabilistic automata and Markov chains obtained through strategy application. First, we give general notation used in the article and refer to [49, 50] for basic concepts of topology and probability theory.

Probability distributions. A distribution on a countable set Q is a function $\mu : Q \to [0, 1]$ such that $\sum_{q \in Q} \mu(q) = 1$; its *support* is the set $\mathsf{supp}(\mu) \stackrel{\text{def}}{=} \{q \in$

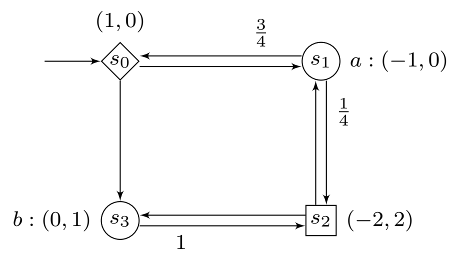


Figure 1: An example game. Moves and states for Player $\Diamond$ and Player $\Box$ are respectively shown as $\bigcirc$, $\Diamond$ and $\Box$. States are annotated with a two-dimensional reward structure. Moves (also called stochastic states) are labelled with actions.

$Q \mid \mu(q) > 0\}$. We denote by $\mathsf{D}(Q)$ the set of all distributions over Q with finite support. A distribution $\mu \in \mathsf{D}(Q)$ is *Dirac* if $\mu(q) = 1$ for some $q \in Q$, and if the context is clear we just write q to denote such a distribution μ.

The vector space $\mathbb{R}^n$. When dealing with multi-objective queries comprising n objectives, we operate in the vector space $\mathbb{R}^n$ of dimension n over the field of reals $\mathbb{R}$, one for each objective, and consider optimisation along n dimensions. We use the standard vector dot product $(\cdot)$ and matrix multiplication. We use the uniform norm $\|\vec{x}\|_\infty \stackrel{\text{def}}{=} \max_{i=1..n} |x_i|$ and the corresponding notion of distance between vectors. For a set $X \subseteq \mathbb{R}^n$, we denote by $\mathsf{conv}(X)$ its convex hull, that is, the smallest convex set containing X. We use the partial order on $\mathbb{R}^n$ defined for every $\vec{x}, \vec{y} \in \mathbb{R}^n$ by $\vec{x} \leq \vec{y}$ if, for every $1 \leq i \leq n$, $x_i \leq y_i$. The *downward closure* of a set X is defined as $\mathsf{dwc}(X) \stackrel{\text{def}}{=} \{\vec{y} \in \mathbb{R}^n \mid \exists \vec{x} \in X \,.\, \vec{y} \leq \vec{x}\}$. Its upward closure is $\mathsf{upc}(X) \stackrel{\text{def}}{=} \{\vec{y} \in \mathbb{R}^n \mid \exists \vec{x} \in X \,.\, \vec{x} \leq \vec{y}\}$. We denote by $\mathsf{C}(X)$ the set of *extreme points* of $\mathsf{dwc}(X)$ for a closed convex set X.

2.1. Stochastic Models

We give the definition of stochastic games and discuss their relationship to probabilistic automata in the sense of [52].

2.1.1. Stochastic games

Primarily, we consider turn-based action-labelled stochastic two-player games (henceforth simply called *games*), which distinguish two types of nondeterminism, each controlled by a separate player. Player $\Diamond$ represents the controllable part for which we want to synthesise a strategy, while Player $\Box$ represents the uncontrollable environment.

Definition 1. *A* game $\mathcal{G}$ *is a tuple* $\langle S, (S_\Diamond, S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta\rangle$*, where* S *is a nonempty, countable set of* states *partitioned into* Player $\Diamond$ states $S_\Diamond$*,* Player $\Box$ states $S_\Box$*, and* stochastic states $S_\bigcirc$*;* $\varsigma \in D(S_\Diamond \cup S_\Box)$ *is an* initial distribution*;* $\mathcal{A}$ *is a set of* actions*;* $\chi : S_\bigcirc \to \mathcal{A} \cup \{\tau\}$ *is a (total)* labelling function*; and* $\Delta : S \times S \to [0, 1]$ *is a* transition function*, such that* $\Delta(s, t) = 0$ *for all* $s, t \in S_\Diamond \cup S_\Box$*,* $\Delta(s, t) \in \{0, 1\}$ *for all* $s \in S_\Diamond \cup S_\Box$ *and* $t \in S_\bigcirc$*, and* $\sum_{t \in S_\Diamond \cup S_\Box} \Delta(s, t) = 1$ *for all* $s \in S_\bigcirc$*.*

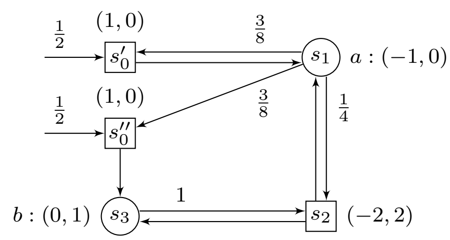


Figure 2: An example PA.

We call $S_\Diamond \cup S_\Box$ the *player states*. Define the *successors* of $s \in S$ as $\Delta(s) \stackrel{\text{def}}{=} \{t \in S \,|\, \Delta(s,t) > 0\}$. For stochastic states s, we sometimes write $s = (a, \mu)$ (called *move*), where $a \stackrel{\text{def}}{=} \chi(s)$, and $\mu(t) = \Delta(s,t)$ for all $t \in S$. If $\Delta(s,(a,\mu)) > 0$, we write $s \xrightarrow{a} \mu$ for the *transition* labelled by $a \in \mathcal{A} \cup \{\tau\}$, called an a-transition. The action labels $a \in \mathcal{A}$ on transitions model observable behaviours, whereas τ can be seen as internal: it is not synchronised in the composition that we formulate in this paper. A move (a, μ) is *incoming* to a state s if $\mu(s) > 0$, and is *outgoing* from a state s if $s \xrightarrow{a} \mu$.

We remark that we work with *finite* games; more precisely, all our statements are for finite stochastic games, except for the induced PAs which may be infinite. In the rest of this paper, if not explicitly stated otherwise, we assume that games have no deadlocks, that is, $|\Delta(s)| \geq 1$ for every $s \in S$.

A *path* $\lambda = s_0 s_1 s_2 \ldots$ is a (possibly infinite) sequence of states such that, for all $i \geq 0$, $\Delta(s_i, s_{i+1}) > 0$. Note that paths of games alternate between player and stochastic states. Given a finite path $\lambda = s_0 s_1 \ldots s_N$, we write $\mathsf{last}(\lambda) = s_N$, and write $|\lambda| = N + 1$ for the length of λ. We denote the set of finite (infinite) paths of a game $\mathcal{G}$ by $\Omega^{\mathsf{fin}}_{\mathcal{G}}$ ($\Omega_{\mathcal{G}}$), and by $\Omega^{\mathsf{fin}}_{\mathcal{G},\Diamond}$ ($\Omega^{\mathsf{fin}}_{\mathcal{G},\Box}$) the set of finite paths ending in a Player $\Diamond$ (Player $\Box$) state.

A finite (infinite) *trace* is a finite (infinite) sequence of actions. Given a path λ, its trace $\mathsf{trace}(\lambda)$ is the sequence of actions that label moves along λ, where we elide τ. We write $\mathcal{A}^*$ (resp. $\mathcal{A}^\omega$) for the set of finite (resp. infinite) sequences over $\mathcal{A}$.

Example 1. *Figure 1 shows a stochastic game, where Player $\Diamond$ and Player $\Box$ states are respectively shown as $\Diamond$ and $\Box$, and moves as $\bigcirc$. A path of the game is $s_0 s_1 s_0 s_3 s_2 s_3$ and its trace is abb.*

2.1.2. Probabilistic automata

If $S_\Diamond = \emptyset$ then the game is a *probabilistic automaton* (PA) [52], which we sometimes write as $\langle S, (S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta \rangle$. The model considered here is due to Segala [52], and should not be confused with Rabin's probabilistic automata [46]. Segala's PAs have strong compositionality properties, as discussed in [56].

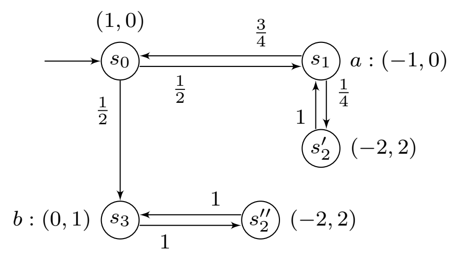


Figure 3: An example DTMC. The labelling function is partial: only s_1 and s_3 have labels.

Note that, in contrast to Markov decision processes[1], we allow PAs to have several moves associated to each action, and similarly for games. An example PA is shown in Figure 2.

An *end component* (EC) is a sub-PA that is closed under the transition relation and strongly connected. Formally, an EC $\mathcal{E}$ of a PA $\mathcal{M}$ is a pair $(S_{\mathcal{E}}, \Delta_{\mathcal{E}})$ with $\emptyset \neq S_{\mathcal{E}} \subseteq S$ and $\emptyset \neq \Delta_{\mathcal{E}} \subseteq \Delta$, such that (i) for all $s \in S_{\mathcal{E}} \cap S_{\bigcirc}$, $\sum_{t \in S_{\Box}} \Delta_{\mathcal{E}}(s,t) = 1$; (ii) for all $s \in S_{\mathcal{E}}$, $\Delta_{\mathcal{E}}(s,t) > 0$ only if $t \in S_{\mathcal{E}}$; and (iii) for all $s,t \in S_{\mathcal{E}}$, there is a finite path $s_0 s_1 \dots s_l \in \Omega^{\mathsf{fin}}_{\mathcal{M}}$ within $\mathcal{E}$ (that is, $s_i \in S_{\mathcal{E}}$ for all $0 \leq i \leq l$), such that $s_0 = s$ and $s_l = t$. An end component is a *maximal end component* (MEC) if it is maximal with respect to the pointwise subset ordering.

2.1.3. Discrete-time Markov chains

In contrast to games and PAs, the discrete-time Markov chain model contains no nondeterminism.

Definition 2. *A* discrete-time Markov chain *(DTMC)* $\mathcal{D}$ *is a tuple* $\langle S, \varsigma, \mathcal{A}, \chi, \Delta \rangle$, *where* S *is a nonempty, countable set of* states, $\varsigma \in D(S)$ *is an* initial distribution *on states,* $\mathcal{A}$ *is a finite alphabet of action,* $\chi : S \to \mathcal{A}$ *is a partial labelling function and* Δ *is a* transition function *such that* $\sum_{t \in S} \Delta(s,t) = 1$.

An example DTMC is shown in Figure 3.

Note that, as opposed to games and PAs, there are no player states in DTMCs but only stochastic states. The labelling function is partial to allow for states that correspond to stochastic states of a game, which are labelled, as opposed to states that correspond to player states of a game, which are unlabelled (see Example 2 below). Note also that DTMCs cannot have deadlocks.

Paths and traces of DTMCs are defined as for games, where the set of finite (infinite) paths is denoted by $\Omega^{\mathsf{fin}}_{\mathcal{D}}$ (resp. $\Omega_{\mathcal{D}}$).

[1]A Markov decision process (MDP) is a PA where all successors of any player state have pairwise distinct labels.

2.2. Strategies

Nondeterminism for each player is resolved by a strategy, which keeps internal memory that can be updated stochastically. For the remainder of this section, fix a game $\mathcal{G} = \langle S, (S_\Diamond, S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta \rangle$.

Definition 3. *A* strategy π *of* Player $\Diamond$ *is a tuple* $\langle \mathfrak{M}, \pi_c, \pi_u, \pi_d \rangle$*, where* $\mathfrak{M}$ *is a countable set of* memory elements*;* $\pi_c : S_\Diamond \times \mathfrak{M} \to D(S_\bigcirc)$ *is a* choice function *s.t.* $\pi_c(s, \mathfrak{m})(a, \mu) > 0$ *only if* $s \xrightarrow{a} \mu$*;* $\pi_u : \mathfrak{M} \times S \to D(\mathfrak{M})$ *is a* memory update function*; and* $\pi_d : S_\Diamond \cup S_\Box \to D(\mathfrak{M})$ *is an* initial distribution *on* $\mathfrak{M}$*. A strategy* σ *of* Player $\Box$ *is defined analogously.*

We will sometimes refer to Player $\Diamond$ strategy as a *controller*. For a given strategy, the game proceeds as follows. It starts in a player state with memory sampled according to the initial distribution. Every time a (stochastic) state s is entered, both players update their current memory $\mathfrak{m}$ and $\mathfrak{n}$ according to these states; the updated memory $\mathfrak{m}'$ and $\mathfrak{n}'$ are $\pi_u(\mathfrak{m}, s)(\mathfrak{m}')$ and $\pi_u(\mathfrak{n}, s)(\mathfrak{n}')$. Once the memory is updated, if s is a stochastic state then the next state is picked randomly according to the probability $t \mapsto \Delta(s, t)$; otherwise, s is a player state and the next stochastic state t is chosen according to the distribution $\pi_c(s, \mathfrak{m}')$ when $s \in S_\Diamond$, and according to $\sigma_c(s, \mathfrak{n}')$ if $s \in S_\Box$.

If the memory update function maps to Dirac distributions, we speak of *deterministic memory update* (DU) strategies, and sometimes use the alternative, equivalent, formulation where $\pi : \Omega^{\mathsf{fin}}_{\mathcal{G},\Diamond} \to \mathsf{D}(S_\bigcirc)$ is a function such that $\pi(\lambda)(a, \mu) > 0$ only if $\mathsf{last}(\lambda) \xrightarrow{a} \mu$ for all $\lambda \in \Omega_{\mathcal{G},\Diamond}$ (and symmetrically for Player $\Box$). If we want to emphasise that memory might not be deterministically updated, we speak of *stochastic memory update* (SU) strategies. The set of Player $\Diamond$ (resp. Player $\Box$) strategies is denoted by Π (resp. Σ), and we use superscripts DU and fin to refer to DU and finite (memory) strategies, respectively. If a DU strategy can be represented with only one memory element and its choice functions maps to a Dirac distribution in every state, it is called *memoryless deterministic* (MD).

2.2.1. Strategy application

Definition 4. *Given a game* $\mathcal{G} = \langle S, (S_\Diamond, S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta \rangle$*,* Player $\Diamond$ *strategy* π *and* Player $\Box$ *strategy* σ*, we define the* induced DTMC $\mathcal{G}^{\pi,\sigma} = \langle S', \varsigma', \mathcal{A}, \chi', \Delta' \rangle$*, where* $S' \subseteq S \times \mathfrak{M} \times \mathfrak{N}$ *is defined as the set of reachable states from* $supp(\varsigma')$ *through* Δ' *defined as follows. For every* $s \in supp(\varsigma)$*,* $\varsigma'(s, \mathfrak{m}, \mathfrak{n}) = \pi_d(s)(\mathfrak{m})\pi_d(s)(\mathfrak{n})$ *and* Δ' *is such that*

$$\Delta'((s, \mathfrak{m}, \mathfrak{n}), (s', \mathfrak{m}', \mathfrak{n}')) = \pi_u(\mathfrak{m}, s')(\mathfrak{m}') \cdot \sigma_u(\mathfrak{n}, s')(\mathfrak{n}') \cdot \begin{cases} \pi_c(s, \mathfrak{m})(s') & \text{if } s \in S_\Diamond \\ \sigma_c(s, \mathfrak{n})(s') & \text{if } s \in S_\Box \\ \Delta(s, s') & \text{if } s \in S_\bigcirc \end{cases} \quad (1)$$

The labelling function χ' *is defined by* $\chi'(s, \mathfrak{m}, \mathfrak{n}) = \chi(s)$ *for every* $s \in S_\bigcirc$*.*

The first two terms of the right-hand side of (1) correspond to the memory updates, while the last term corresponds to the probability of moving from one state to another depending on the type of the current state.

Note that paths of the induced DTMC include memory. We introduce a mapping $\mathsf{path}_{\mathcal{G}}((s_0, \mathfrak{m}_0, \mathfrak{n}_0) \cdots (s_n, \mathfrak{m}_n, \mathfrak{n}_n)) = s_0 \cdots s_n$ to retrieve paths of the game from paths of the induced DTMC.

Example 2. *Figure 3 shows the induced DTMC from the stochastic game of Figure 1 by the two strategies described below. The* $\mathsf{Player}\,\Diamond$ *strategy is memoryless (let* $\mathfrak{m}$ *its single memory element); it randomises amongst the successors of* s_0 *with the same probability* $\frac{1}{2}$*. The* $\mathsf{Player}\,\Box$ *strategy decides in state* s_2 *to go to the state visited just before entering* s_2*. It hence requires only two memory elements,* $\mathfrak{n}$ *and* $\mathfrak{n}'$*. The current memory element is always* $\mathfrak{n}$ *except when* s_2 *is chosen from* s_3*, where it is updated to* $\mathfrak{n}'$*. For the sake of readability we denote by* s_i *the state* $(s_i, \mathfrak{m}, \mathfrak{n})$ *for* $i \neq 2$*, by* $s_2' = (s_2, \mathfrak{m}, \mathfrak{n}')$ *and* $s_2'' = (s_2, \mathfrak{m}, \mathfrak{n})$*. For instance,* $\mathsf{path}_{\mathcal{G}}(s_0 s_1 s_0 s_3 s_2' s_3) = s_0 s_1 s_0 s_3 s_2 s_3$*.*

Similarly, given a PA $\mathcal{M}$ and a $\mathsf{Player}\,\Box$ strategy σ, one can define an induced DTMC $\mathcal{M}^\sigma$ and a mapping $\mathsf{path}_{\mathcal{M}}$, where a generic path of the induced PA is of the form $\kappa = (s_0, \mathfrak{n}_0) \cdots (s_n, \mathfrak{n}_n)$ and is mapped to $\mathsf{path}_{\mathcal{M}}(\kappa) \stackrel{\text{def}}{=} s_0 \cdots s_n$. Note that the maps $\mathsf{path}_{\mathcal{M}}$ and $\mathsf{path}_{\mathcal{G}}$ preserve the lengths of the paths.

We define the (standard) probability measure on paths of a DTMC $\mathcal{D} = \langle S, \varsigma, \mathcal{A}, \chi, \Delta \rangle$ in the following way. The *cylinder set* of a finite path $\lambda \in \Omega_{\mathcal{D}}^{\mathsf{fin}}$ (resp. trace $w \in \mathcal{A}^*$) is the set of infinite paths (resp. traces) with prefix λ (resp. w). For a finite path $\lambda = s_0 s_1 \ldots s_n \in \Omega_{\mathcal{D}}^{\mathsf{fin}}$ and a distribution $\vartheta \in \mathsf{D}(S)$, we define $\mathbb{P}_{\mathcal{D},\vartheta}(\lambda)$, the measure of its cylinder set weighted by the distribution ϑ, by $\mathbb{P}_{\mathcal{D},\vartheta}(\lambda) \stackrel{\text{def}}{=} \vartheta(s_0) \prod_{i=0}^{n-1} \Delta(s_i, s_{i+1})$. If $\vartheta = \varsigma$, i.e. the initial distribution, we omit it and just write $\mathbb{P}_{\mathcal{D}}$. Given a PA $\mathcal{M}$ and a strategy σ, one can define for every path λ the measure of its cylinder set by $\mathbb{P}_{\mathcal{M}}^{\sigma}(\lambda) \stackrel{\text{def}}{=} \sum\{\mathbb{P}_{\mathcal{M}^\sigma}(\lambda') \mid \mathsf{path}_{\mathcal{M}}(\lambda') = \lambda\}$. Similarly, given a game $\mathcal{G}$ and a pair of strategies π, σ, define for every path λ the measure $\mathbb{P}_{\mathcal{G}}^{\pi,\sigma}(\lambda) \stackrel{\text{def}}{=} \sum\{\mathbb{P}_{\mathcal{G}^{\pi,\sigma}}(\lambda') \mid \mathsf{path}_{\mathcal{G}}(\lambda') = \lambda\}$.

We introduce the remaining definitions for a generic model (game, PA or DTMC) together with the probability measure $\mathbb{P}$ on its paths. Given a finite trace w, $\mathsf{paths}(w)$ denotes the set of minimal finite paths with trace w, i.e. $\lambda \in \mathsf{paths}(w)$ if $\mathsf{trace}(\lambda) = w$ and there is no path $\lambda' \neq \lambda$ with $\mathsf{trace}(\lambda') = w$ and λ' being a prefix of λ. The measure of the cylinder set of w is $\tilde{\mathbb{P}}(w) \stackrel{\text{def}}{=} \sum_{\lambda \in \mathsf{paths}(w)} \mathbb{P}(\lambda)$, and we call $\tilde{\mathbb{P}}$ the *trace distribution* induced by $\mathbb{P}$. The measures uniquely extend to infinite paths due to Carathéodory's extension theorem. We denote by $\mathbb{E}[\vec{\rho}]$ the expectation wrt $\mathbb{P}$ of a measurable function $\vec{\rho}$ over infinite paths, that is, $\int \vec{\rho}(\lambda) d\mathbb{P}(\lambda)$, and use the same lower- and upper-script notation for $\mathbb{E}$ and $\mathbb{P}$, for instance $\mathbb{E}_{\mathcal{D},\vartheta}$ denotes expectation wrt $\mathbb{P}_{\mathcal{D},\vartheta}$.

Given a subset $T \subseteq S$, let $\mathbb{P}(\mathsf{F}^{=k}\, T) \stackrel{\text{def}}{=} \sum\{\mathbb{P}(\lambda) \mid \lambda = s_0 s_1 \ldots \text{s.t. } s_k \in T \wedge \forall i < k \,.\, s_i \notin T\}$ the probability to reach T in exactly k steps, and by $\mathbb{P}(\mathsf{F}\, T) \stackrel{\text{def}}{=} \sum\{\mathbb{P}(\lambda) \mid \lambda = s_0 s_1 \ldots \text{s.t. } \exists i \,.\, s_i \in T\}$ the probability to eventually reach T.

2.2.2. Determinising strategies

In this section we show that SU and DU strategies are equally powerful if the memory size is not restricted (Proposition 1). The memory elements of the determinised strategies are distributions over memory elements of the original strategy. Such distributions can be interpreted as the *belief* the other player has about the memory element, knowing only the history and the rules to update the memory, while the actual memory based on sampling is kept secret. The term belief is inspired by the study of partially observable Markov decision processes. At any time, the belief attributes to a memory element $\mathfrak{m}$ the probability of $\mathfrak{m}$ under the original strategy given the history.

Definition 5. *Given an SU strategy $\pi = \langle \mathfrak{M}, \pi_c, \pi_u, \pi_d \rangle$, we define its* determinised strategy $\bar{\pi} = \langle \mathfrak{D}(\pi), \bar{\pi}_c, \bar{\pi}_u, \bar{\pi}_d \rangle$, *where $\mathfrak{D}(\pi) \subseteq D(\mathfrak{M})$ is a countable set called the* belief space *defined as the reachable beliefs from the initial beliefs $\bar{\pi}_d(s)$ under belief updates $\bar{\pi}_u$ along paths of the game defined as follows. The initial belief in a state is the initial memory distribution in this state:*

$$\bar{\pi}_d(s) \stackrel{\text{def}}{=} \pi_d(s).$$

Any belief $\mathfrak{d}$ is updated according to a state s' as follows:

$$\bar{\pi}_u(\mathfrak{d}, s')(\mathfrak{m}') \stackrel{\text{def}}{=} \sum_{\mathfrak{m} \in \mathfrak{M}} \mathfrak{d}(\mathfrak{m}) \cdot \pi_u(\mathfrak{m}, s')(\mathfrak{m}').$$

The choice of a state s' is made according to a belief $\mathfrak{d}$ as follows:

$$\bar{\pi}_c(s, \mathfrak{d})(s') \stackrel{\text{def}}{=} \sum_{\mathfrak{m} \in \mathfrak{M}} \mathfrak{d}(\mathfrak{m}) \pi_c(s, \mathfrak{m})(s').$$

Note that determinising a finite memory SU strategy can lead to either a finite or an infinite memory DU strategy.

We can now state the main result of this section, namely, that the original and the determinised strategy give exactly the same semantics. They are indistinguishable from the Player $\Box$ viewpoint.

Proposition 1. *Given a game $\mathcal{G}$ and two strategies π, σ, it holds that $\mathbb{P}_{\mathcal{G}}^{\pi,\sigma} = \mathbb{P}_{\mathcal{G}}^{\bar{\pi},\sigma}$, where $\bar{\pi}$ is the determinisation of π.*

This proposition is proved in Appendix A.1. We do not need to consider the determinisation of Player $\Box$ strategies. Note, however, that it could be defined in the same way, and using Proposition 1 twice (once for each player) yields $\mathbb{P}_{\mathcal{G}}^{\pi,\sigma} = \mathbb{P}_{\mathcal{G}}^{\bar{\pi},\bar{\sigma}}$.

2.3. Winning Conditions

2.3.1. Rewards and long-run behaviours

A *reward structure* of a game $\mathcal{G}$ is a function $r : S \to \mathbb{R}$; it is *defined on actions* $\mathcal{A}_r \subseteq \mathcal{A}$ if $r(a, \mu) = r(a, \mu')$ for all moves $(a, \mu), (a, \mu') \in S_{\bigcirc}$ such that $a \in \mathcal{A}_r$, and $r(s) = 0$ otherwise. r straightforwardly extends to induced DTMCs via $r(s) = r(\mathsf{path}_{\mathcal{G}}(s))$ for $s \in S_{\mathcal{G}^{\pi,\sigma}}$. Given reward structures r and r', define the reward structure $r + r'$ by $(r + r')(s) \stackrel{\text{def}}{=} r(s) + r'(s)$ for all $s \in S$, and, given $v \in \mathbb{R}$, define $r + v$ by $(r + v)(s) \stackrel{\text{def}}{=} r(s) + v$ for all $s \in S$.

For a path $\lambda = s_0 s_1 \ldots$ (of a game or DTMC) and a reward structure r, we define $\mathsf{rew}^N(r)(\lambda) \stackrel{\text{def}}{=} \sum_{i=0}^{N} r(s_i)$, and similarly for traces if r is defined on actions. We use the following types of reward:

- the *average reward* (*mean-payoff*) is $\mathsf{mp}(r)(\lambda) \stackrel{\text{def}}{=} \underline{\lim}_{N\to\infty} \frac{1}{N+1} \mathsf{rew}^N(r)(\lambda)$;
- the *ratio reward* is $\mathsf{ratio}(r/c)(\lambda) \stackrel{\text{def}}{=} \underline{\lim}_{N\to\infty} \mathsf{rew}^N(r)(\lambda)/(1 + \mathsf{rew}^N(c)(\lambda))$, where c is a *weakly positive* reward structure, that is, it is non-negative and there exists $c_{\min} > 0$ such that $\mathbb{P}^{\pi,\sigma}_{\mathcal{G}}(\mathsf{mp}(c) > c_{\min}) = 1$ for all π and σ.

Example 3. *Let r and c be the first and second component of the reward structure of the game shown in Figure 1. The reward structure c is weakly positive because, under every pair of strategies, s_2 or s_3 are visited with positive frequency. In the induced DTMC shown in Figure 3, every path λ that begins with $s_0 s_1 s_0 s_3 s''_2 s_3$ has cumulative rewards after 6 steps equal to $\mathsf{rew}^6(r)(\lambda) = r(s_0) + r(s_1) + r(s_0) + r(s_3) + r(s''_2) + r(s_3) = 1 + (-1) + 1 + 0 + (-2) + 0 = -1$ and $\mathsf{rew}^6(c)(\lambda) = 4$, leading to a ratio of $\mathsf{rew}^6(r)(\lambda)/(1 + \mathsf{rew}^6(c)(\lambda)) = -1/5$ after 6 steps.*

If a DTMC $\mathcal{D}$ has a finite state space, the limit inferior ($\underline{\lim}$) of the average and ratio rewards can be replaced by the true limit, as it is almost surely defined (see Lemma 15 and 16 in Appendix A.2). Ratio rewards $\mathsf{ratio}(r/c)$ generalise average rewards $\mathsf{mp}(r)$ since, to express the latter, we can let $c(s) = 1$ for all states s of $\mathcal{G}$, see [58].

2.3.2. Specifications and objectives

A *specification* φ on a model (game, PA, or DTMC) is a predicate on its path distributions. We call π *winning* for φ in $\mathcal{G}$ if, for every Player $\Box$ strategy σ, $\mathbb{P}^{\pi,\sigma}_{\mathcal{G}}$ satisfies φ. We say that φ is *achievable* if such a winning strategy exists, written $\mathcal{G} \models \varphi$. A specification φ on a PA $\mathcal{M}$ is *satisfied* if, for every Player $\Box$ strategy σ, $\mathbb{P}^{\sigma}_{\mathcal{M}}$ satisfies φ, which we write $\mathcal{M} \models \varphi$. A specification φ on a DTMC $\mathcal{D}$ is *satisfied* if $\mathbb{P}_{\mathcal{D}}$ satisfies φ, which we write $\mathcal{D} \models \varphi$. A specification φ is *defined on traces of* $\mathcal{A}$ if $\varphi(\tilde{\mathbb{P}}) = \varphi(\tilde{\mathbb{P}}')$ for all $\mathbb{P}, \mathbb{P}'$ such that $\tilde{\mathbb{P}}(w) = \tilde{\mathbb{P}}'(w)$ for all traces $w \in \mathcal{A}^*$. We consider the following *objectives*, which are specifications with single-dimensional reward structures.

Semantics	Reward	Syntax	Definition
(a.s.) satisfaction	mean payoff	$\mathsf{Pmp}(r)(v)$	$\mathbb{P}(\mathsf{mp}(r) \geq v) = 1$
(a.s.) satisfaction	ratio	$\mathsf{Pratio}(r/c)(v)$	$\mathbb{P}(\mathsf{ratio}(r/c) \geq v) = 1$
expectation	mean payoff	$\mathsf{Emp}(r)(v)$	$\mathbb{E}[\mathsf{mp}(r)] \geq v$
expectation	ratio	$\mathsf{Eratio}(r/c)(v)$	$\mathbb{E}[\mathsf{ratio}(r/c)] \geq v$

Note that, when inducing a DTMC, the reward structure is carried over and the mean-payoff and ratio reward are not affected; hence, specifications defined for games are also naturally carried over to the induced models. In particular, a $\mathsf{Player}\ \Diamond$ strategy π of a game $\mathcal{G}$ is winning for a specification φ if and only if, for every $\mathsf{Player}\ \Box$ strategy σ, $\mathcal{G}^{\pi,\sigma} \models \varphi$. The same remark holds for induced PAs $\mathcal{G}^{\pi}$ defined in Section 2.4 below.

The objective $\mathsf{Pmp}(r)(v)$ (resp. $\mathsf{Emp}(r)(v)$) is equivalent to $\mathsf{Pmp}(r-v)(0)$ (resp. $\mathsf{Emp}(r-v)(0)$), i.e. with the rewards shifted by $-v$. Hence, if not otherwise stated, we assume without loss of generality that mean-payoff objectives have target 0 and write $\mathsf{Pmp}(r)$ (resp. $\mathsf{Emp}(r)$). An objective with target v is *ε-achievable* if, for all $\varepsilon > 0$, the objective is achievable with target $v - \varepsilon$ by some strategy, which we call *ε-optimal*. We can replace the non-strict inequality in an ε-achievable objective by a strict inequality, and retain ε-achievability. We hence define for expectations $\neg\mathsf{Emp}(r)(v) \stackrel{\text{def}}{=} \mathsf{Emp}(-r)(-v)$ and $\neg\mathsf{Eratio}(r/c)(v) \stackrel{\text{def}}{=} \mathsf{Eratio}(-r/c)(-v)$. We do not consider negating a.s. satisfaction objectives here.

Additionally, we consider *expected energy (EE) objectives*, which we use as an auxiliary tool in strategy synthesis. A DTMC $\mathcal{D}$ satisfies the EE objective $\mathsf{EE}(r)$ if there exists a finite *shortfall* v_0, such that, for every state s of $\mathcal{D}$, $\mathbb{E}_{\mathcal{D},s}[\mathsf{rew}^N(r)] \geq v_0$ for all $N \geq 0$.

We recall known results about strategies in PAs and games.

Lemma 1 (Theorem 9.1.8 in [45]). *In finite PAs, MD strategies suffice to achieve single-dimensional Emp objectives.*

Lemma 2 (see Section 2.1.1 of [34]). *Given a game $\mathcal{G}$, and a set $A \subseteq S$, the set of states A' from which $\mathsf{Player}\ \Diamond$ can reach A almost surely is computable in polynomial time. Moreover, an MD strategy π reaching almost surely A from any state of A' is computable in polynomial time.*

2.3.3. Multi-objective queries and their Pareto sets

A *multi-objective query* (MQ) φ is a Boolean combination of objectives and its truth value is defined inductively on its syntax. Given an MQ with n thresholds $v_1, v_2, \ldots$, call $\vec{v} = (v_1, v_2, \ldots)$ the *target vector*. Denote by $\varphi[\vec{x}]$ the MQ φ, where, for all i, the bound v_i is replaced by x_i. An MQ φ is a *conjunctive query* (CQ) if it is a conjunction of objectives. The notation $\mathsf{Pmp}(\vec{r})(\vec{v})$, $\mathsf{Emp}(\vec{r})(\vec{v})$ stands for the CQ $\bigwedge_{i=1}^{n} \mathsf{Pmp}(r_i)(v_i)$ and $\bigwedge_{i=1}^{n} \mathsf{Emp}(r_i)(v_i)$, respectively. The notation $\mathsf{Pratio}(\vec{r}/\vec{c})(\vec{v})$, $\mathsf{Eratio}(\vec{r}/\vec{c})(\vec{v})$ stands for the CQ $\bigwedge_{i=1}^{n} \mathsf{Pratio}(r_i/c_i)(v_i)$ and $\bigwedge_{i=1}^{n} \mathsf{Eratio}(r_i/c_i)(v_i)$, respectively. We write $\vec{\varepsilon}$ to denote the vector $(\varepsilon, \varepsilon, \ldots, \varepsilon)$, and, if the context is clear, we use ε instead of $\vec{\varepsilon}$.

The Pareto set $\text{Pareto}(\varphi)$ of an MQ φ is the topological closure of the set of achievable vectors. Alternatively, this set can be defined as the set of *approximable* target vectors, namely, the vectors $\vec{v}$ such that, for every ε, the target vector $\varphi[\vec{v} - \varepsilon]$ is achievable. We denote by $\text{Pareto}_{\text{FDU}}(\varphi)$ the subset of $\text{Pareto}(\varphi)$ concerning achievability by a finite DU strategy.

In some of our results, we consider only finite memory adversaries. We denote by $\text{Pareto}_{\text{FDU,FSU}}(\varphi)$ the topological closure of the set of vectors achievable against finite SU strategies by finite DU strategies. Note that a Pareto set is equal to its downward closure. More precisely, we distinguish three regions in

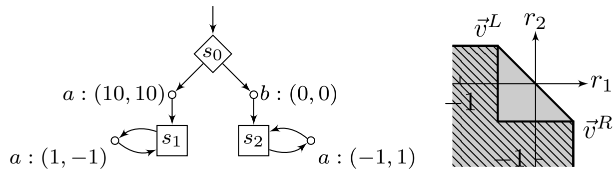


Figure 4: Left: A game. Right: Its Pareto sets for Pmp (hashed) and Emp (grey). Vector $\vec{v}_L \stackrel{\text{def}}{=} (-1/2, 1/2)$ (resp. $\vec{v}_R \stackrel{\text{def}}{=} (1/2, -1/2)$) and its downward closure $\mathsf{dwc}(\{\vec{v}_L\})$ is achieved by the MD strategy that chooses a (resp. b) in s_0. For $p \in (0, 1)$ the strategy that chooses a with probability p and b with probability $1-p$ achieves $\mathsf{Emp}(\vec{r})(p\vec{v}_L + (1-p)\vec{v}_R)$ but it achieves only $\mathsf{Pmp}(\vec{r})(\min(\vec{v}_L, \vec{v}_R))$, where the minimum is taken componentwise.

a Pareto set, the interior of the Pareto set where vectors are achievable; the boundary of a Pareto set, usually called the *Pareto frontier*, where vectors are approximable but may not be achievable; and the complement of the Pareto set, where vectors are not achievable.

Remark 1. *For every game and reward structure r, if a strategy π is winning for $\mathsf{Pmp}(\vec{r})$ then it is winning for $\mathsf{Emp}(\vec{r})$. In particular,* $\mathrm{Pareto}(\mathsf{Pmp}(\vec{r})) \subseteq \mathrm{Pareto}(\mathsf{Emp}(\vec{r}))$ *and* $\mathrm{Pareto}_{\mathrm{FDU}}(\mathsf{Pmp}(\vec{r})(\vec{v})) \subseteq \mathrm{Pareto}_{\mathrm{FDU}}(\mathsf{Emp}(\vec{r}))$ *but the converse inclusions do not hold in general. The same remark holds when replacing mean-payoff by ratio rewards.*

Indeed, given a probability distribution on paths, if the mean payoff is almost surely above a threshold then the expected mean payoff is also above this threshold, leading to the inclusion claimed. Figure 4 provides an example where the inclusion is strict.

2.3.4. Problem statement

We are mainly interested in the following *synthesis problem*: given a quantitative specification φ, an approximable target vector and a positive real ε, synthesise an ε-optimal strategy for this vector. To obtain achievable specifications, we are also interested in (under-approximating) the Pareto set to provide a choice of approximable targets as input to the synthesis problem. Specifically, we seek to compute, for every $\varepsilon > 0$, ε-tight under-approximations of Pareto sets where, given two subsets X, Y of $\mathbb{R}^n$, X is an ε-tight under-approximation of Y if $Y \subseteq X$ and for every $x \in X$ there is $y \in Y$ such that $\|x - y\|_\infty \leq \varepsilon$.

2.3.5. Comparison of ratio rewards and mean payoffs

We now discuss the relationship between different classes of specifications. Firstly, note that ratio rewards are defined on traces, since we can control which actions are counted in the denominator, and are therefore well suited to our compositional synthesis framework described in Section 5 that is tailored to such properties. On the other hand, average rewards $\underline{\lim}_{N\to\infty} \frac{1}{N+1} \mathsf{rew}^N(r)(\lambda)$ are not defined over traces in general, since the divisor $(N+1)$ counts the steps in the game, irrespective of whether the specification takes them into account. Hence, when composing systems by interleaving transitions, the additional actions counted in the denominator of the ratio reward, in between these originally counted, skew the value of the average rewards.

However, Pratio and Pmp are inter-reducible in the following sense, and we will use this fact in Section 4 to obtain a reduction for expected mean payoffs.

Proposition 2. *A strategy π is winning for $\mathsf{Pratio}(\vec{r}/\vec{c})(\vec{v})$ if and only if it is winning for $\mathsf{Pmp}(\vec{r} - \vec{v} \bullet \vec{c})(0)$ where, for every dimension i and state s, $[\vec{r} - \vec{v} \bullet \vec{c}]_i(s) = r_i(s) - v_i c_i(s)$.*

This proposition is proved in Appendix A.3.

2.4. A Two-Step Semantics for Stochastic Games

PAs arise naturally from games when one considers fixing only the Player $\Diamond$ strategy, and then checking against all Player $\Box$ strategies if it is winning. Later in the paper, for instance in Theorem 15, we will show how to automatically lift results from the PA (and MDP) world to the game domain (from the literature or proved here). For this, we will need to map strategies of the induced PA to Player $\Box$ strategies of the original game. To facilitate the lifting, we adopt a two-step semantics defined as follows. In the first step we apply a Player $\Diamond$ strategy to a game, leading to an induced PA. Then, in the second step, we apply a Player $\Box$ strategy to the induced PA, resulting in a probability measure that is the same as that obtained by applying both strategies simultaneously (Proposition 3).

2.4.1. First step: inducing the PA

We consider a DU Player $\Diamond$ strategy π (note that this is without loss of generality by Proposition 1). The induced PA $\mathcal{G}^\pi$ essentially corresponds to the game where π has been applied. The memory of Player $\Diamond$ is encoded in the states of the induced PA as depicted in Figure 5. To allow alternation between stochastic and Player $\Box$ states in the induced PA, we transform each Player $\Diamond$ state s' into several Player $\Box$ states, each of the form (s', s''), corresponding to the choice of $s'' \in S_\bigcirc$ as a successor of s'. Any incoming transition $s \to s'$ of the game is thus replaced by several transitions in the PA, each of the form $s \to (s', s'')$ with probability given as a product of $\Delta(s, s')$ and the probability of s'' given by π_c in s'. Formally, the induced PA is defined as follows.

Definition 6. *Let $\mathcal{G} = \langle S, (S_\Diamond, S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta \rangle$ be a game and let π be a Player $\Diamond$ DU strategy. The* induced PA $\mathcal{G}^\pi$ *is $\langle S', (S'_\Box, S'_\bigcirc), \varsigma', \mathcal{A}, \chi', \Delta' \rangle$, where $S'_\Box \subseteq (S_\Box \cup S_\Diamond \times S_\bigcirc) \times \mathfrak{M}$ and $S'_\bigcirc \subseteq S_\bigcirc \times \mathfrak{M}$ are defined inductively as the reachable states from the initial distribution ς', defined by $\varsigma'(s, \pi_\mathsf{d}(s)) = \varsigma(s)$; and through the transition relation Δ' defined as follows. Given states $\mathfrak{s} \in S'$ of the form $(s, \mathfrak{m})$ and $\mathfrak{t} \in S'$ of the form $(t, \mathfrak{m}')$ or $((t, t'), \mathfrak{m}')$, $\Delta'(\mathfrak{s}, \mathfrak{t})$ is not null*

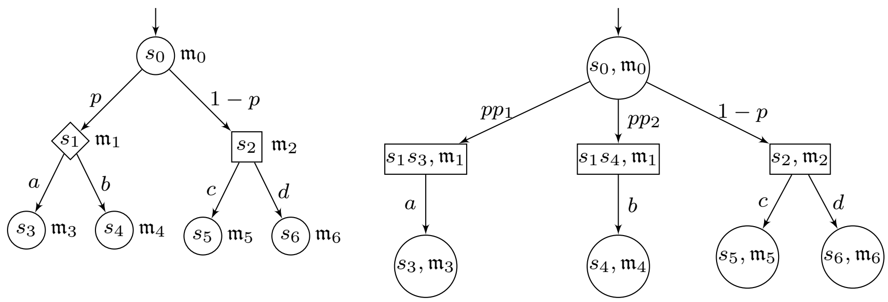


Figure 5: Stochastic game $\mathcal{G}$ (left) and induced PA $\mathcal{G}^\pi$ (right). Memory elements are represented on the right of each state of the game, and encoded in the states of the PA. At s_1 with memory $\mathfrak{m}_1$, the strategy π plays a and b with probability p_1 and p_2, respectively.

only if $\mathfrak{m}' = \pi_{\mathsf{u}}(\mathfrak{m}, t)$ *and is defined by*

$$\Delta'(\mathfrak{s}, \mathfrak{t}) \stackrel{\text{def}}{=} \Delta(s,t) \cdot \begin{cases} \pi_{\mathsf{c}}(t, \mathfrak{m}')(t') & \textit{if } t \stackrel{\text{def}}{=} (t,t') \in S_\Diamond \times S_\bigcirc; \\ 1 & \textit{otherwise} \end{cases}$$

Every state of the form $((s,s'), \mathfrak{m})$ *has only one successor (thus taken with probability* 1*), which is* $(s', \pi_{\mathsf{u}}(\mathfrak{m}, s'))$*. The labelling function is defined by* $\chi'(\mathfrak{s}) = \chi(s)$ *for* $s \in S_\bigcirc$*.*

Example 4. *Figure 2 shows the PA induced from the game of Figure 1 by the memoryless strategy that randomises in* s_0 *between* s_1 *and* s_3 *with the same probability* $\frac{1}{2}$*, as in Example 2. The single memory element* $\mathfrak{m}$ *is omitted for the sake of readability and states* $((s_0, s_1), \mathfrak{m})$*,* $((s_0, s_3), \mathfrak{m})$ *are called* s_0' *and* s_0'' *respectively.*

Remark 2. *The induced PA corresponding to a finite DU strategy has a finite state space, which was not the case with the definition of [5]. We rely on this fact to prove numerous results in the paper.*

2.4.2. Second step: inducing the DTMC

Given a game $\mathcal{G}$ and a DU strategy π, every strategy σ of the induced PA $\mathcal{G}^\pi$ induces a DTMC $(\mathcal{G}^\pi)^\sigma$. One can define a mapping, still denoted by $\mathsf{path}_\mathcal{G}$, from paths of this DTMC to the game. Formally, every state of the DTMC is of the form $((s, \mathfrak{m}), \mathfrak{n})$ or $(((s,s'), \mathfrak{m}), \mathfrak{n})$, which is mapped by $\mathsf{path}_\mathcal{G}$ to s.

An associated probability measure can thus be defined by

$$\mathbb{P}_\mathcal{G}^{(\pi,\sigma)}(\lambda) \stackrel{\text{def}}{=} \sum \{\mathbb{P}_{(\mathcal{G}^\pi)^\sigma}(\lambda') \mid \mathsf{path}_\mathcal{G}(\lambda') = \lambda\},$$

called *the two-step semantics.*

The two-step semantics is justified by Proposition 3, showing equivalence with the original semantics of Definition 4.

Proposition 3 (Equivalence of semantics)**.** *The two-step semantics is equivalent to the semantics of the game of Definition 4 in the following sense. Let $\mathcal{G}$ be a game and π a DU strategy, then for every strategy σ in $\mathcal{G}$ (resp. σ' in $\mathcal{G}^\sigma$) one can build a strategy σ' in $\mathcal{G}^\sigma$ (resp. σ in $\mathcal{G}$) such that $\mathbb{P}_{\mathcal{G}}^{\pi,\sigma} = \mathbb{P}_{\mathcal{G}}^{(\pi,\sigma')}$. Moreover, if π has finite memory, then σ has finite memory if and only if σ' has finite memory.*

To build a Player $\Box$ strategy σ in a game from a strategy σ' in the induced PA, it suffices to simulate the deterministic memory of π (available from the state of the induced PA) in the memory of σ. If the strategies π and σ' are finite then so is the memory of σ. The other direction is straightforward; if σ is a strategy in a game, then one can use it in an induced PA without even taking care of the memory of π.

3. Conjunctions of Pmp Objectives

In this section we consider conjunctions of Pmp objectives, which maintain several mean payoffs almost surely above the corresponding thresholds. We first show in Corollary 2 that we can decide which player wins in co-NP time. Next, to synthesise strategies, we introduce a reduction to expected energy (EE) objectives in Lemma 5. We then construct succinct ε-optimal finite SU strategies in Theorem 7.

3.1. Decision Procedures

In this section we present our decidability result of the achievability problem for Pmp CQs, based on a general class of objectives defined via *shift-invariant submixing* functions. A function $\varrho : \Omega_{\mathcal{G}} \to \mathbb{R}$ is *shift-invariant* if $\forall \kappa \in \Omega_{\mathcal{G}}^{\mathsf{fin}}, \lambda \in \Omega_{\mathcal{G}} \,.\, \varrho(\kappa\lambda) = \varrho(\lambda)$. A function $\varrho : \Omega_{\mathcal{G}} \to \mathbb{R}$ is *submixing* if, for all $\kappa, \kappa', \lambda \in \Omega_{\mathcal{G}}$ such that λ is an interleaving of κ and κ', it holds that $\varrho(\lambda) \leq \max\{\varrho(\kappa), \varrho(\kappa')\}$. Given a measurable function ϱ, we write $\mathsf{P}(\varrho)$ for the objective $\mathbb{P}(\varrho \geq 0) = 1$.

We obtain a co-NP algorithm by studying the strategies Player $\Box$ needs to win for Pmp objectives against Player $\Diamond$, and using that the games are qualitatively determined for Pmp objectives. We have from [33] that MD strategies suffice for Player $\Diamond$ to win for single-dimensional shift-invariant submixing functions.

Theorem 1 (Theorem V.2 of [33])**.** *Let $\mathcal{G}$ be a game, let $\varrho : \Omega_{\mathcal{G}} \to \mathbb{R}$ be measurable, shift-invariant and submixing. Then Player $\Diamond$ has an MD strategy $\tilde{\pi}$ such that $\inf_\sigma \mathbb{E}_{\mathcal{G}}^{\tilde{\pi},\sigma}[\varrho] = \sup_\pi \inf_\sigma \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\varrho]$.*

Further, a game $\mathcal{G}$ with specification φ is *qualitatively determined* if either Player $\Diamond$ has a winning strategy, or Player $\Box$ has a spoiling strategy. It is called Player $\Box$-*positional* if the following implication holds: if Player $\Box$ has a spoiling strategy then it has an MD spoiling strategy.

Theorem 2 (Theorem 7 of [32])**.** *Stochastic games with shift-invariant winning condition are qualitatively determined.*[2]

Given a measurable subset A of $\Omega_{\mathcal{G}}$, we denote by 1_A its indicator function, that is, $1_A(\lambda) \stackrel{\text{def}}{=} 1$ if $\lambda \in A$ and 0 otherwise.

Lemma 3. *Let $\varrho_1, \dots, \varrho_n$ be shift-invariant submixing functions, and let $A \stackrel{\text{def}}{=} \{\lambda \mid \exists i \,.\, -\rho_i(\lambda) < v_i\}$. The function 1_A is shift-invariant and submixing.*

Proof. Since ϱ_i is shift-invariant for all i, also 1_A is shift-invariant. We now show that 1_A is submixing. Let $\lambda, \kappa, \kappa' \in \Omega_{\mathcal{G}}$ such that λ is an interleaving of κ and κ'. If $1_A(\kappa) = 1$ or $1_A(\kappa') = 1$ then $1_A(\lambda) \leq \max\{1_A(\kappa), 1_A(\kappa')\}$. Otherwise, $1_A(\kappa) = 1_A(\kappa') = 0$, that is, $-\varrho_i(\kappa) \geq v_i$ and $-\varrho_i(\kappa') \geq v_i$ for all i. Since ϱ_i is submixing, $\varrho_i(\lambda) \leq \max\{\varrho_i(\kappa), \varrho_i(\kappa')\}$, for all i. Then, for all i, $v_i \leq \min\{-\varrho_i(\kappa), -\varrho_i(\kappa')\} \leq -\varrho_i(\lambda)$. Thus, $1_A(\lambda) = 0 \leq \max\{1_A(\kappa), 1_A(\kappa')\}$ as expected. ☐

Theorem 3. *A game $\mathcal{G}$ with specification $\mathsf{P}(-\vec{\varrho})$, where $\varrho_1, \dots, \varrho_n$ are shift-invariant submixing functions, is $\mathsf{Player}\,\Box$-positional.*

Proof. Assume that $\mathsf{Player}\,\Box$ has a spoiling strategy σ. It means that for every $\mathsf{Player}\,\Diamond$ strategy π, it holds that $\mathbb{E}^{\pi,\sigma}_{\mathcal{G}}[1_A] > 0$ with A as in Lemma 3. Since, by Lemma 3, 1_A is submixing and shift-invariant, by Theorem 1 (via switching players), there exists an MD $\mathsf{Player}\,\Box$ strategy $\tilde{\sigma}$ in $\mathcal{G}$ such that $\forall \pi \,.\, \mathbb{E}^{\pi,\tilde{\sigma}}_{\mathcal{G}}[1_A] > 0$, concluding the proof. ☐

Theorem 4. *Let $\mathcal{G}$ be a game with a specification φ qualitatively determined and $\mathsf{Player}\,\Box$-positional. If for PAs $\mathcal{M}$ with specification φ the problem $\exists \sigma, \mathcal{M}^\sigma \models \varphi$ is in the time-complexity class A, then the problem $\exists \pi \,.\, \forall \sigma \,.\, \mathcal{G}^{\pi,\sigma} \models \varphi$ is in co-NP if $\mathsf{A} \subseteq \mathsf{co}$-$\mathsf{NP}$, and in A if $\mathsf{A} \supseteq \mathsf{co}$-$\mathsf{NP}$.*

Proof. By qualitative determinacy, the decision problem of interest is equivalent $\forall \sigma \,.\, \exists \pi \,.\, \mathcal{G}^{\pi,\sigma} \models \varphi$. The answer is negative exactly if $\exists \sigma \,.\, \forall \pi \,.\, \mathcal{G}^{\pi,\sigma} \models \neg\varphi$, which is equivalent to deciding whether some MD strategy σ satisfies $\forall \pi \,.\, \mathcal{G}^{\pi,\sigma} \models \neg\varphi$. Such an MD spoiling strategy σ can be guessed in polynomial time. To decide $\forall \pi \,.\, \mathcal{G}^{\pi,\sigma} \models \neg\varphi$, it suffices to decide its negation $\exists \pi \,.\, \mathcal{G}^{\pi,\sigma} \models \varphi$, and this problem is in the class A. The overall complexity is hence the maximum complexity of co-NP and A. ☐

Using Theorems 2 and 3, we obtain the following corollary.

Corollary 1. *Let $\mathcal{G}$ be game, let $\varrho_1, \dots, \varrho_n$ be shift-invariant submixing functions, and suppose the problem whether there exists a strategy σ for a PA $\mathcal{M}$ such that $\mathcal{M}^\sigma$ satisfies $\mathsf{P}(-\vec{\varrho})$ is in the time-complexity class A. The problem $\exists \pi \,.\, \forall \sigma \,.\, \mathcal{G}^{\pi,\sigma} \models \mathsf{P}(-\vec{\varrho})$ is in co-NP if $\mathsf{A} \subseteq \mathsf{co}$-$\mathsf{NP}$, and in A if $\mathsf{A} \supseteq \mathsf{co}$-$\mathsf{NP}$.*

[2] This result was originally stated for the weaker assumption of tail conditions, see the discussion in III.B. of [33].

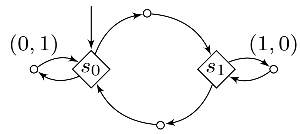


Figure 6: Player $\Diamond$ needs infinite memory to win optimally for $\mathsf{Pmp}(\vec{r})(\frac{1}{4}, \frac{1}{4})$, but finite-memory DU strategies are sufficient for ε-optimality for MDPs ([6]) and for stochastic games as we show in Theorem 13 (example taken from [6]).

Applying this to mean-payoff, we get the following corollary.

Corollary 2. *The Pmp CQ achievability problem is in co-NP.*

Proof. We use the previous results and the fact that the problem $\exists\sigma\,.\,\mathsf{Pmp}(\vec{r})$ is decidable in polynomial time for PAs, by virtue of B.3 of [6]. $\square$

Finally, we consider the complexity of Pareto set computation for Pmp CQs. We approximate the Pareto set of an n-dimensional conjunction, $\mathsf{Pmp}(\vec{r})$, via gridding, by some grid-size ε, the set of targets in the hyperrectangle $\{\vec{v} \in \mathbb{R}^n \,|\, \forall i\,.\, -\rho^* \leq v_i \leq \rho^*\}$, where $\rho^* \stackrel{\text{def}}{=} \max_{i,s\in S}|r_i(s)|$. At every such point $\vec{v}$ in the grid, we call the co-NP decision procedure of Corollary 2, and hence obtain an ε-approximation of the Pareto set by taking the downward closure of the set of achievable points. There are ρ^*/ε sections per dimension, and $2^{|S|}$ strategies to be checked with the polynomial-time oracle of B.3. in [6], and so we obtain the following theorem.

Theorem 5. *An ε-approximation of the Pareto set* $\mathrm{Pareto}(\mathsf{Pmp}(\vec{r}))$*, for an n-dimensional conjunction of Pmp objectives, can be computed using $\mathcal{O}((\rho^*/\varepsilon)^n)$ calls to the co-NP oracle of Corollary 2.*

3.2. Finite Memory Strategies

In general, infinite memory might be required to achieve a multi-objective query: in the game in Figure 6, Player $\Diamond$ has to play the transitions between s_0 and s_1 in order to achieve $\mathsf{Pmp}(\vec{r})(\frac{1}{4}, \frac{1}{4})$, but can only do so optimally if in the limit these transitions are never played; this fact holds already for MDPs, see [6]. Nevertheless, we are able to show that finite-memory DU strategies are sufficient for ε-optimality for stochastic games, see Theorem 13. For MDPs, this was proved in [6]. We work with SU strategies, which can be exponentially more succinct than DU strategies, and were shown to be equally powerful if the memory is not restricted in Proposition 1.

3.2.1. ε-optimality with finite DU Player $\Diamond$ strategies

The following theorem states that Player $\Diamond$ can achieve any target ε-optimally with a finite DU strategy if it is achievable by an arbitrary Player $\Diamond$ strategy.

Theorem 6. *Given a game and a multi-dimensional reward structure $\vec{r}$, then it holds that* $\mathrm{Pareto}(\mathsf{Pmp}(\vec{r})) = \mathrm{Pareto}_{\mathrm{FDU}}(\mathsf{Pmp}(\vec{r}))$.

This theorem is proved in Appendix B.1.

3.2.2. Succinctness of SU strategies

We justify our use of SU strategies by showing that they can be exponentially more succinct than DU strategies.

Proposition 4. *Finite SU strategies can be exponentially more succinct than finite DU strategies for expected and almost sure mean-payoff.*

The proof method is based on similar results in [16, 55]. The proof is included in Appendix B.2.

3.3. Inter-Reduction between Pmp *and* EE

We demonstrate in this section how strategy synthesis for almost sure mean-payoff objectives reduces to synthesis for expected energy objectives, under ε-optimality. Our use of energy objectives is inspired by [16], where non-stochastic games are considered. We first show that for finite DU strategies of $\mathsf{Player}\ \Diamond$ it is sufficient to consider finite $\mathsf{Player}\ \Box$ strategies.

3.3.1. Finite $\mathsf{Player}\ \Box$ *strategies sufficient for* EE

We explain the intuition behind the reduction. When $\mathsf{Player}\ \Diamond$ fixes a finite DU strategy π, aiming to satisfy a conjunction of EE objectives, then the aim of $\mathsf{Player}\ \Box$ is to spoil at least one objective in the finite induced PA $\mathcal{G}^\pi$. In the following we use boldface notation for vectors over the state space, reserving the arrow notation for vectors over the reward dimensions. Let $\mathcal{M} = \langle S, (S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta \rangle$ be a PA, and let r be a single-dimensional reward structure. For a given $\mathsf{Player}\ \Box$ strategy σ, write Δ^σ for the transition function of the induced DTMC $\mathcal{M}^\sigma$. The sequence of *expected non-truncated energies* is inductively defined by $e^0_{s,\mathfrak{m}} \stackrel{\text{def}}{=} 0$, and, for all $k > 0$,

$$e^k_{s,\mathfrak{m}} \stackrel{\text{def}}{=} r(s) + \sum_{(t,\mathfrak{m}') \in \Delta^\sigma(s,\mathfrak{m})} \Delta^\sigma((s,\mathfrak{m}),(t,\mathfrak{m}')) \cdot e^{k-1}_{t,\mathfrak{m}'},$$

for all states $(s,\mathfrak{m})$ of $\mathcal{M}^\sigma$. $\mathsf{Player}\ \Box$ spoils if for every shortfall v_0 there exists a state $(s,\mathfrak{m})$ such that $e^k_{s,\mathfrak{m}} \le v_0$. To witness whether $\mathsf{Player}\ \Box$ can spoil, without needing to induce the DTMC $\mathcal{M}^\sigma$, we also define the sequence $(\mathbf{u}^k)_{k\ge 0}$ of *single-dimensional truncated energy*, parametrised by states of the PA $\mathcal{M}$. That is, for all states s of $\mathcal{M}$ put $u^0_s \stackrel{\text{def}}{=} 0$, and, for every $k > 0$, we define

$$u^{k+1}_s \stackrel{\text{def}}{=} \begin{cases} \min(0, r(s) + \min_{t\in\Delta(s)} u^k_t) & \text{if } s \in S_\Box \\ \min(0, r(s) + \sum_{t\in\Delta(s)} \Delta(s,t) u^k_t) & \text{if } s \in S_\bigcirc. \end{cases} \tag{2}$$

One can see by induction that $(\mathbf{u}^k)_{\ge 0}$ is a non-increasing sequence. We denote by $\mathbf{u}^*$ its limit in $(\mathbb{R}_{\le 0} \cup \{-\infty\})^{|S|}$. Let S_{fin} (resp. S_∞) be the set of states s of $\mathcal{M}$ such that u^*_s is finite (resp. infinitely negative). We now show that $S_\infty \ne \emptyset$ witnesses that $\mathsf{Player}\ \Box$ can spoil the EE objective with a finite strategy.

Proposition 5. *Let $\mathcal{M}$ be a finite PA with a one-dimensional reward structure r. If $S_\infty \neq \emptyset$, then $\mathsf{Player}\ \Box$ has a finite strategy to spoil $\mathsf{EE}(r - \varepsilon)$, for every $\varepsilon > 0$.*

The proof proceeds by showing that, for k large enough and for states in $S_\infty \neq \emptyset$, there is no cut-off used to define $(\mathbf{u}^k)$, and hence $(\mathbf{u}^k)$ satisfies the same linear equations as the expected non-truncated energy $(\mathbf{e}^k)$, see Appendix B.3.

We also require the following lemma proved in Appendix B.4.

Lemma 4. *If $\mathsf{Player}\ \Box$ can spoil $\mathsf{EE}(r)$, in a finite PA with a one-dimensional reward structure r, then $S_\infty \neq \emptyset$.*

Finally, with the help of Lemma 4 and Proposition 5, we can show that it is sufficient to consider finite memory $\mathsf{Player}\ \Box$ strategies for EE objectives.

Proposition 6. *Let π be a finite DU $\mathsf{Player}\ \Diamond$ strategy. If π wins for $\mathsf{EE}(\vec{r} - \vec{\varepsilon})$ for some $\varepsilon > 0$ against all finite $\mathsf{Player}\ \Box$ strategies, then it wins for $\mathsf{EE}(\vec{r})$ for all $\mathsf{Player}\ \Box$ strategies.*

Proof. We show the contrapositive. Assume the strategy π loses for $\mathsf{EE}(\vec{r})$ against an arbitrary strategy of $\mathsf{Player}\ \Box$. Then there is a coordinate r of the rewards $\vec{r}$ such that $\mathsf{Player}\ \Box$ wins $\mathsf{EE}(r)$ in the induced PA $\mathcal{G}^\pi$. By Lemma 4 this implies that $S_\infty \neq \emptyset$, which by Proposition 5 yields that $\mathsf{Player}\ \Box$ spoils $\mathsf{EE}(r - \varepsilon)$, and hence $\mathsf{EE}(\vec{r} - \vec{\varepsilon})$ for every ε, with a finite memory strategy. □

3.3.2. Transforming between EE and Pmp

We are now ready to show that EE and Pmp objectives are equivalent up to ε-achievability, and the proof is included in Appendix B.5.

Lemma 5. *Given a finite strategy π for $\mathsf{Player}\ \Diamond$, the following hold:*

(i) if π achieves $\mathsf{EE}(\vec{r})$, then π achieves $\mathsf{Pmp}(\vec{r})$; and

(ii) if π is DU and achieves $\mathsf{Pmp}(\vec{r})$, then π achieves $\mathsf{EE}(\vec{r} + \vec{\varepsilon})$ for all $\varepsilon > 0$.

The above reduction to energy objectives enables the formulation of our main method, see Theorem 7 below, for computing strategies achieving $\mathsf{EE}(\vec{r} + \vec{\varepsilon})$, and hence, by virtue of Lemma 5(i), deriving ε-optimal strategies for $\mathsf{Pmp}(\vec{r})$. Lemma 5(ii) guarantees completeness of our method, in the sense that, for any target $\vec{v}$ such that $\mathsf{Pmp}(\vec{r})(\vec{v})$ is achievable, we compute an ε-optimal strategy. If $\mathsf{Pmp}(\vec{r})(\vec{v})$ is not achievable, it is detected by the decision procedure of Corollary 2.

3.4. Strategy Synthesis

This section describes the strategy synthesis method and proceeds as follows. We first show in Section 3.4.1 that strategies that can be geometrically represented via what we call an *ε-consistent memory mapping* are ε-optimal for EE objectives (and hence for Pmp objectives as shown in the previous section). We describe the synthesis algorithm in Section 3.4.3, which is based on the construction of a memory mapping obtained by iterating a Bellman operator studied in Section 3.4.2.

3.4.1. Geometry of SU strategies

Given a strategy π and an n-dimensional reward structure $\vec{r}$, a *memory mapping* is a partial function $f_\pi : \mathfrak{M} \times S \to \mathbb{R}^n$; we typically abbreviate $f_\pi(\mathfrak{m}, s) = \vec{\mathfrak{m}}_s$. A memory mapping is *$\varepsilon$-consistent for $\vec{v}_0$* if $\sum_s \varsigma(s) \sum_{\mathfrak{m} \in \mathfrak{M}} \pi_{\mathsf{d}}(s)(\mathfrak{m})\vec{\mathfrak{m}}_s \geq \vec{v}_0$, and, for all $s \in S$, $s' \in \Delta(s)$, $\mathfrak{m} \in \mathfrak{M}$,

$$\begin{aligned} \textstyle\sum_{t\in\Delta(s)} \pi_{\mathsf{c}}(s,\mathfrak{m})(t) \cdot \vec{\mathfrak{a}}(t,\ \mathfrak{m}) &\geq \vec{\mathfrak{m}}_s - \vec{r}(s) - \varepsilon \quad \text{if } s \in S_\Diamond, \\ \vec{\mathfrak{a}}(s',\mathfrak{m}) &\geq \vec{\mathfrak{m}}_s - \vec{r}(s) - \varepsilon \quad \text{if } s \in S_\Box, \qquad \textit{and} \\ \textstyle\sum_{t\in\mathsf{supp}(\mu)} \mu(t) \cdot \vec{\mathfrak{a}}(t,\ \mathfrak{m}) &\geq \vec{\mathfrak{m}}_s - \vec{r}(s) - \varepsilon \quad \text{if } s = (a,\mu) \in S_\bigcirc, \end{aligned}$$

where $\vec{\mathfrak{a}}(t,\mathfrak{m}) \stackrel{\text{def}}{=} \sum_{\mathfrak{m}'\in\mathfrak{M}} \pi_{\mathsf{u}}(\mathfrak{m},t)(\mathfrak{m}') \cdot \vec{\mathfrak{m}}'_t$.

Lemma 6. *Let π be a strategy. If there is a memory mapping that is ε-consistent for $\vec{v}_0$, then π achieves $\mathsf{EE}(\vec{r} + \vec{\varepsilon})$.*

This lemma is proved in Appendix B.6.

3.4.2. Shortfall computation by iteration of a Bellman operator

Before we introduce the Bellman operator, we outline the construction of the space that it acts on. Note that, in a game with a specification consisting of n objectives, we need to keep a set of n-dimensional real-valued vectors for each of the $|S|$ states and moves, where each such n-dimensional vector $\vec{v}$ intuitively corresponds to an achievable target for multi-dimensional truncated energy. Thus, we require that each element of our space is an $|S|$-dimensional vector of subsets of $\mathbb{R}^n$.

Formally, the construction is as follows. Given $M \geq 0$ and a set $A \subseteq \mathbb{R}^n$, define the *M-downward closure* of A by $\mathsf{dwc}(A) \cap \mathrm{Box}_M$, where $\mathrm{Box}_M \stackrel{\text{def}}{=} [-M, 0]^n$. The set of convex closed M-downward-closed subsets of $\mathbb{R}^n$ is denoted by $\mathcal{P}_{c,M}$ and endowed with the partial order $\sqsubseteq$ defined by $A \sqsubseteq B$ if $\mathsf{dwc}(B) \subseteq \mathsf{dwc}(A)$. For a set $X \subseteq (\mathbb{R}^n)^{|S|}$ and state s, we denote by X_s the sth component of X. We define the space $\mathcal{C}_M \stackrel{\text{def}}{=} \mathcal{P}_{c,M}^{|S|}$ and endow it with the product partial order $\sqsubseteq$ defined by $Y \sqsubseteq X$ if, for every $s \in S, Y_s \sqsubseteq X_s$. The set $\bot_M \stackrel{\text{def}}{=} \mathrm{Box}_M^{|S|}$ is a *bottom element* for this partial order (that is, for all $X \in \mathcal{C}_M$, $\bot_M \sqsubseteq X$). More precisely, we have an algebraic characterisation of $\mathcal{C}_M$ as a complete partial order (CPO) shown in Appendix B.7.

Proposition 7. $(\mathcal{C}_M, \sqsubseteq)$ *is a complete partial order.*

We now define operations on the CPO $\mathcal{C}_M$. Given $A, B \in \mathcal{P}_{c,M}$, let $A + B \stackrel{\text{def}}{=} \{\vec{x} + \vec{y} \,|\, \vec{x} \in A, \vec{y} \in B\}$ (the *Minkowski sum*). Given $A \in \mathcal{P}_{c,M}$, let $\alpha \times A \stackrel{\text{def}}{=} \{\alpha \cdot \vec{x} \,|\, \vec{x} \in A\}$ for $\alpha \in \mathbb{R}$, and let $A + \vec{x} \stackrel{\text{def}}{=} \{\vec{x}' + \vec{x} \,|\, \vec{x}' \in A\}$ for $\vec{x} \in \mathbb{R}^n$. Given $Y \in \mathcal{C}_M$, which is a vector of sets, define $[Y + \vec{y}]_s \stackrel{\text{def}}{=} Y_s + \vec{y}$.

The Bellman operator F_M. In games, in order to construct $\mathsf{Player}\ \Diamond$ strategies for EE objectives, we consider the truncated energy for multi-dimensional rewards, which we capture via a Bellman operator $F_{M,\mathcal{G}}$ over the CPO $\mathcal{C}_M$, parameterised by $M \geq 0$. Our operator $F_{M,\mathcal{G}}$ is closely related to the operator

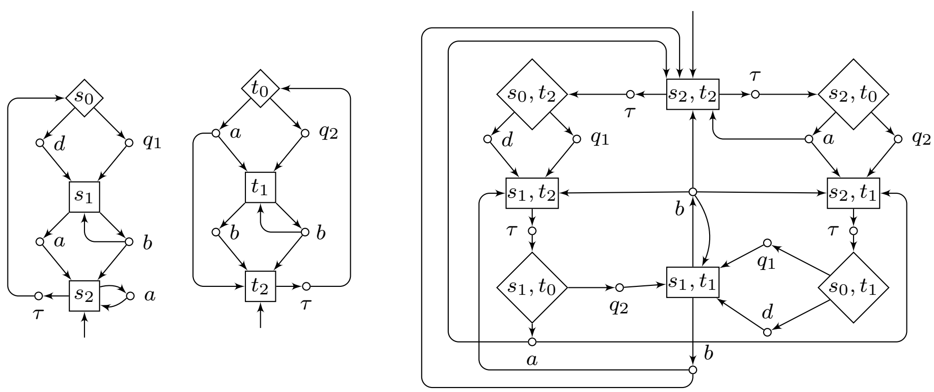


Figure 7: Example games $\mathcal{G}^1$ (left) and $\mathcal{G}^2$ (centre), with their composition $\mathcal{G}$ (right). All distributions are uniform.

for expected total rewards in [19], but here we cut off values outside of Box_M, similarly to the controllable predecessor operator of [16] for computing energy in non-stochastic games. Bounding with M allows us to use a geometric argument to upper-bound the number of iterations of our operator (Proposition 10 below), replacing the finite lattice arguments of [16]. We define the operator $F_{M,\mathcal{G}} : \mathcal{C}_M \to \mathcal{C}_M$ by

$$[F_{M,\mathcal{G}}(X)]_s \stackrel{\text{def}}{=} \mathrm{Box}_M \cap \mathsf{dwc}\left(\vec{r}(s) + \begin{cases} \mathsf{conv}(\bigcup_{t \in \Delta(s)} X_t) & \text{if } s \in S_\Diamond \\ \bigcap_{t \in \Delta(s)} X_t & \text{if } s \in S_\Box \\ \sum_{t \in \Delta(s)} \Delta(s,t) \times X_t & \text{if } s \in S_\bigcirc \end{cases}\right),$$

for all $s \in S$. If the game $\mathcal{G}$ is clear from context, we write just F_M. The operator F_M computes the expected truncated energy Player $\Diamond$ can achieve in the respective state types. In $s \in S_\Diamond$, Player $\Diamond$ can achieve the values in successors (union), and can randomise between them (convex hull). In $s \in S_\Box$, Player $\Diamond$ can achieve only values that are in all successors (intersection), since Player $\Box$ can pick arbitrarily. Lastly, in $s \in S_\bigcirc$, Player $\Diamond$ can achieve values with the prescribed distribution.

Fixpoint of F_M. A fixpoint of F_M is an element $Y \in \mathcal{C}_M$ such that $F_M(Y) = Y$. We show that iterating F_M on $\bot_M$ converges to the least fixpoint of F_M.

Proposition 8. *F_M is order-preserving, and the increasing sequence $F^k_M(\bot_M)$ converges to $\mathsf{fix}(F_M)$ defined by $[\mathsf{fix}(F_M)]_s \stackrel{\text{def}}{=} [\bigcap_{k \geq 0} F^k_M(\bot_M)]_s$. Further, $\mathsf{fix}(F_M)$ is the unique least fixpoint of F_M.*

This proposition is a consequence of Scott continuity of F_M and the Kleene fixpoint theorem. For the proof see Appendix B.7.

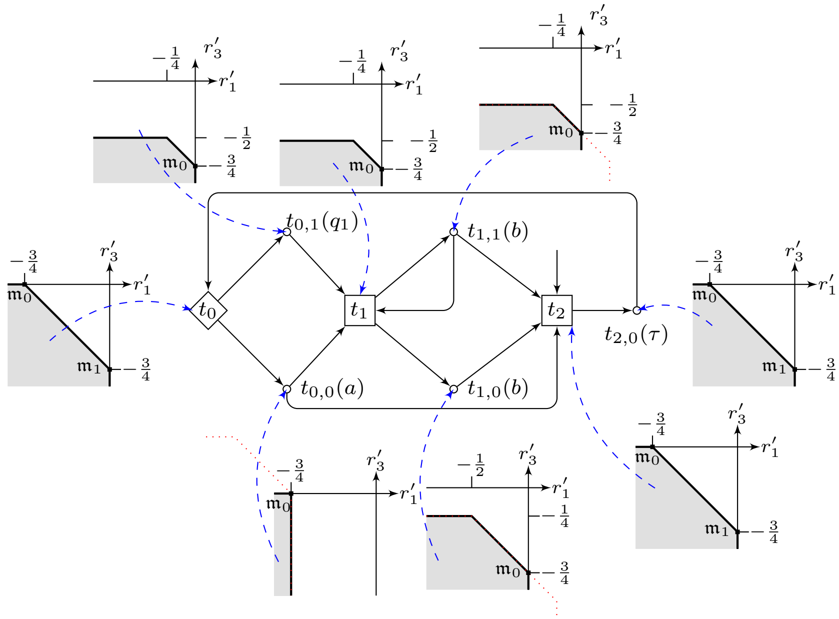


Figure 8: Fixpoint for F_M in $\mathcal{G}^2$ of Figure 7. For easier reference, moves are given state names. Each state s has an associated set $\mathsf{fix}(F_M)(s)$ pointed to by the blue (dashed) arrows, we do not show the box Box_M. A partial memory mapping is annotated at the corner points.

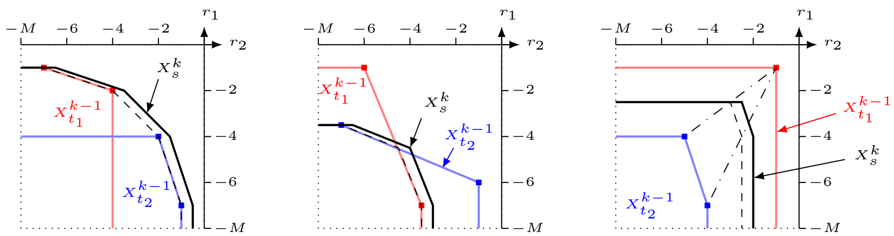
Example 5. *In Figure 8 we show the fixpoint $\mathsf{fix}(F_M)$ for game $\mathcal{G}^2$ of Figure 7 for the reward structures r'_1 and r'_3 defined by $r'_1(a) = -\frac{3}{4}$, $r'_1(b) = \frac{1}{4}$, $r'_3(a) = \frac{3}{4}$, $r'_3(a) = -\frac{1}{4}$, and zero otherwise.*

Non-emptiness of the fixpoint. Non-emptiness of the fixpoint of F_M for some $M > 0$ is a sufficient condition for computing an ε-optimal strategy. To show the completeness of our method, stated in Theorem 7 below, we ensure in the following proposition that, when an ε-optimal strategy exists, then the fixpoint of F_M for some $M > 0$ is non-empty.

Proposition 9. *For every $\varepsilon > 0$, if $\mathsf{EE}(\vec{r} - \vec{\varepsilon})$ is achievable by a finite DU strategy, then $[\mathsf{fix}(F_M)]_s \neq \emptyset$ for every $s \in \mathsf{supp}(\varsigma)$ for some $M \geq 0$.*

This proposition is proved in Appendix B.8.

An ε-approximation of the fixpoint. We approximate $\mathsf{fix}(F_M)$ in a finite number of steps, and thus compute the set of shortfall vectors required for Player $\Diamond$ to win for $\mathsf{EE}(\vec{r} + \vec{\varepsilon})$ given $\varepsilon > 0$. By Proposition 8, the fixpoint $\mathsf{fix}(F_M)$ is the limit of $F_M^k(\bot_M)$ as $k \to \infty$. We let $X^k \stackrel{\text{def}}{=} F_M^k(\bot_M)$. Hence, by applying F_M k times to $\bot_M$ we compute the sets X_s^k of shortfall vectors at state s, so that, for any $\vec{v}_0 \in X_s^k$, Player $\Diamond$ can keep the expected energy



(a) Player $\Diamond$ state s. (b) Player $\Box$ state s. (c) Move $s = (a, \mu)$; μ uniform.

Figure 9: Illustration of the fixpoint computation for a state s with successors t_1, t_2, and rewards $r_1(s) = 0.5$ and $r_2(s) = 0$.

above $\vec{v}_0$ during k steps of the game. We illustrate this fixpoint computation in Figure 9: at iteration k, the set X_s^k of possible shortfalls until k steps is computed from the corresponding sets X_t^{k-1} for successors t of s at iteration $k-1$. The values are restricted to be within Box_M, so that obtaining an empty set at a state s in the value iteration is an indicator of divergence at s. Moreover, given some $\varepsilon > 0$, if, after a finite number of iterations k, successive sets X^{k+1} and X^k satisfy $X^{k+1} + \varepsilon \sqsubseteq X^k$ and $X_s^k \neq \emptyset$ for every $s \in \mathsf{supp}(\varsigma)$, then we can construct a finite-memory strategy achieving $\mathrm{EE}(\vec{r} + \vec{\varepsilon})$. The strategies use memory corresponding to the extreme points $\mathsf{C}(X_s^k)$.

In Proposition 10 below we state a bound on the number of steps k necessary to obtain $X^{k+1} + \varepsilon \sqsubseteq X^k$.

Proposition 10. *Given M, $\varepsilon > 0$, and a sequence $(X^k)_{k \geq 0}$ over $\mathcal{C}_M$ such that $X^k \sqsubseteq X^{k+1}$ for every $k \geq 0$, there exists $k \leq k^{**} \stackrel{\text{def}}{=} \left[n((\lceil \frac{M}{\varepsilon} \rceil + 1)^2 + 2)\right]^{|S|}$, such that $X^{k+1} + \varepsilon \sqsubseteq X^k$.*

This proposition is proved in Appendix B.9 using Theorem 4.5.2 of [51] on graphs.

3.4.3. The synthesis algorithm

The synthesis algorithm for Pmp CQs (Algorithm 1) computes an SU strategy by constructing it together with an ε-consistent memory mapping.

Construction of the memory mapping. We define a Player $\Diamond$ strategy π achieving $\mathsf{EE}(\vec{r} + \vec{\varepsilon})$ in a game $\mathcal{G} = \langle S, (S_\Diamond, S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta \rangle$, for a given $\varepsilon \geq 0$. Let $X \in \mathcal{C}_M$, and denote by $T_X \subseteq S$ the set of states and moves s for which $[F_M(X)]_s \neq \emptyset$. For any point $\vec{p} \in X_s$, there is some $\vec{q} \geq \vec{p}$ that can be obtained by a convex combination of extreme points $\mathsf{C}(X_s)$, and so the strategy we construct uses $\mathsf{C}(X_s)$ as memory, randomising to attain the convex combination $\vec{q}$. We define $\pi = \langle \mathfrak{M}, \pi_\mathsf{c}, \pi_\mathsf{u}, \pi_\mathsf{d} \rangle$ as follows.

- $\mathfrak{M} \stackrel{\text{def}}{=} \bigcup_{s \in T_X} \{(s, \vec{p}) \mid \vec{p} \in \mathsf{C}(X_s)\}$;
- π_d is defined by $\pi_\mathsf{d}(s) = (s, \vec{q}_0^{\,s})$ for any $s \in T_X$ and arbitrary $\vec{q}_0^{\,s} \in \mathsf{C}(X_s)$;

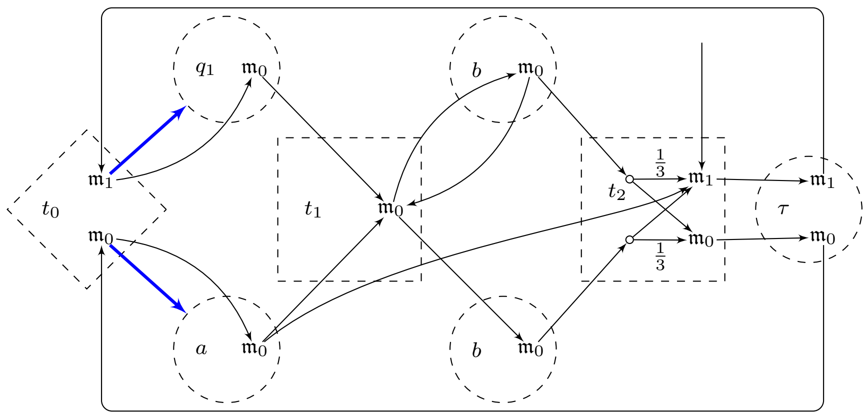


Figure 10: Strategy constructed from the fixpoint in Figure 8, using the memory mapping shown therein.

- The update π_{u} and next move function π_{c} are defined as follows: at state s with memory $(s, \vec{p})$, for all $t \in \Delta(s)$, pick n vectors $\vec{q}_i^{\,t} \in \mathsf{C}(X_t^k)$ for $1 \leq i \leq n$ and distributions $\beta^t \in \mathsf{D}([1,n])$, such that

  for $s \in S_\Diamond$: $\quad \exists \alpha \in \mathsf{D}(\Delta(s) \cap T_X) \,.\, \sum_t \alpha(t) \cdot \sum_i \beta^t(i) \cdot \vec{q}_i^{\,t} \geq \vec{p} - \vec{r}(s) - \varepsilon,$

  for $s \in S_\Box$: $\quad \forall t \in \Delta(s) \,.\, \sum_i \beta^t(i) \cdot \vec{q}_i^{\,t} \geq \vec{p} - \vec{r}(s) - \varepsilon,$

  for $s \in S_\bigcirc$, with $s = (a, \mu)$: $\quad \sum_{t \in \mathsf{supp}(\mu)} \mu(t) \cdot \sum_i \beta^t(i) \cdot \vec{q}_i^{\,t} \geq \vec{p} - \vec{r}(s) - \varepsilon,$

  and let, for all $t \in \Delta(s) \cap T_X$,

$$\pi_{\mathsf{u}}((s,\vec{p}),t)(t,\vec{q}_i^{\,t}) \stackrel{\text{def}}{=} \beta^t(i) \quad \text{for all } i$$

$$\pi_{\mathsf{c}}(s,(s,\vec{p}))(t) \stackrel{\text{def}}{=} \alpha(t) \quad \text{if } s \in S_\Diamond.$$

Example 6. *In Figure 10 we give the strategy constructed from the fixpoint* $\mathit{fix}(F_M)$ *shown in Figure 8.*

We now show that the strategy π defined above is well defined if $F_M(X) + \varepsilon \sqsubseteq X$ for $X \in \mathcal{C}_M$, and achieves $\mathsf{EE}(\vec{r} + \vec{\varepsilon})$ if $[F_M(X)]_s \neq \emptyset$ for every $s \in \mathsf{supp}(\varsigma)$.

Lemma 7. *Let* $\varepsilon \geq 0$*, let* $X \in \mathcal{C}_M$*, and let* $f_\pi(s,\vec{p}) \stackrel{\text{def}}{=} \vec{p}$ *be the memory mapping for the Player* $\Diamond$ *strategy* π *defined above. If* $F_M(X) + \varepsilon \sqsubseteq X$ *and* $[F_M(X)]_s \neq \emptyset$ *for every* $s \in \mathit{supp}(\varsigma)$*, then* f_π *is* ε*-consistent for* M*.*

Proof of this lemma can be found in Appendix B.10.

The Algorithm. We can now summarise our synthesis algorithm. Given a game $\mathcal{G}$, a reward structure $\vec{r}$ with target $\vec{v}$, and $\varepsilon > 0$, Algorithm 1 computes a strategy winning for $\mathsf{Pmp}(\vec{r})(\vec{v} - \varepsilon)$. The algorithm terminates if the specification is achievable, as a large enough value for M in Box_M exists according to Proposition 9, and, if the specification is not achievable, this is captured by our decision procedure of Corollary 2. Note, however, that before starting the algorithm we do not have an a-priori bound on M.

Algorithm 1 PMP Strategy Synthesis

```
1: function SynthPMP(𝒢, r⃗, v⃗, ε)
2:     if Corollary 2 for Pmp(r⃗ − v⃗) yields no then return null;
3:     else
4:         Set the reward structure to r⃗ − v⃗ + ε/2; M ← 2; X ← ⊥_M;
5:         while true do
6:             while F_M(X) + ε/2 ⋢ X do
7:                 X ← F_M(X);
8:             if [F_M(X)]_s ≠ ∅ for every s ∈ supp(ς) then
9:                 Construct π for ε/2 using Lemma 7; return π;
10:            else
11:                M ← M^2; X ← ⊥_M;
```

Theorem 7. *Algorithm 1 terminates, returning a finite* ε*-optimal strategy for* $\mathsf{Pmp}(\vec{r})(\vec{v})$ *if it is achievable, and returning* `null` *otherwise.*

Proof. The case when $\mathsf{Pmp}(\vec{r}-\vec{v})$ is not achievable is covered by Corollary 2. Suppose $\mathsf{Pmp}(\vec{r}-\vec{v})$ is achievable then, by Theorem 6, $\mathsf{Pmp}(\vec{r}-\vec{v}-\frac{\varepsilon}{8})$ is achievable by a finite DU strategy. By Lemma 5 (ii), the objective $\mathsf{EE}(\vec{r}-\vec{v}+\frac{\varepsilon}{4})$ is achievable by a finite DU strategy. Applying Proposition 9 with $\vec{r}' \stackrel{\text{def}}{=} \vec{r}-\vec{v}+\frac{\varepsilon}{4}+\varepsilon'$ and $\varepsilon' = \frac{\varepsilon}{4}$, we have that there exists an M such that, for every $s \in \mathsf{supp}(\varsigma)$, $[\mathsf{fix}(F_M)]_s$ is nonempty for the reward structure $\vec{r}-\vec{v}+\frac{\varepsilon}{2}$. The condition in Line 8 is then satisfied. Further, due to the bound M on the size of the box Box_M in the value iteration, the inner loop terminates after a finite number of steps, as shown in Proposition 10. Then, by Lemma 7 and Lemma 6, the strategy constructed in Line 9 (with degradation factor $\frac{\varepsilon}{2}$ for the reward $\vec{r}-\vec{v}+\frac{\vec{\varepsilon}}{2}$) satisfies $\mathsf{EE}(\vec{r}-\vec{v}+\vec{\varepsilon})$, and hence, using Lemma 5(i), we have $\mathsf{Pmp}(\vec{r})(\vec{v}-\vec{\varepsilon})$. $\square$

4. Boolean Combinations for Expectation Objectives

In this section we consider Boolean combinations of expectation objectives. First, in Section 4.1 we show how to transform Boolean combinations of a general class of expectation objectives to conjunctions of the same type of objective. Then, in Section 4.2, we show how to synthesise strategies for Emp objectives using Pmp objectives for games with the *controllable multichain* property. These two main results of this section then allow us to synthesise (ε-optimally) strategies for arbitrary Boolean combinations of Emp objectives.

4.1. From Conjunctions to Arbitrary Boolean Combinations

In this section we consider generic expectation objectives of the form $\mathbb{E}[\varrho] \geq u$ and their Boolean combinations. We only require that the function ϱ is *integrable*, that is, for every pair of strategies π and σ, $\mathbb{E}^{\pi,\sigma}_{\mathcal{G}}[|\varrho|]$ is well-defined and finite. A function ϱ is called *globally bounded* by B if, for every π and σ,

$\mathbb{E}^{\pi,\sigma}_{\mathcal{G}}[|\varrho|] \leq B$. Given n integrable functions $\varrho_i : \Omega_{\mathcal{G}} \to \mathbb{R}$ for $1 \leq i \leq n$ and a target vector $\vec{u} \in \mathbb{R}^n$, we denote by $\mathsf{E}(\vec{\varrho})(\vec{u})$ the conjunction of objectives $\bigwedge_{i=1}^{n} \mathbb{E}[\varrho_i] \geq u_i$.

We are mainly interested in the following objectives, expressible in terms of integrable and globally bounded functions.

- The **expected total rewards in stopping games**. They were studied in [19, 20], where a preliminary and specialised version of the results of this section was presented.
- The **expected mean-payoff objectives**. A global bound for this objective is $B = \max_S r(s)$.
- The **expected ratio rewards**. They are particularly well suited to our compositional framework, as they are defined on traces and admit synthesis methods for Boolean combinations. A global bound for $\mathsf{ratio}(r/c)$ is $B = \max_S r(s)/c_{\min}$. This result is proved in Appendix C.1.

We establish that Boolean combinations of expectation objectives reduce to conjunctions of linear combinations of expectation objectives. Any Boolean combination of objectives can be converted to conjunctive normal form (CNF), that is, of the form $\bigwedge_{i=1}^{n} \bigvee_{j=1}^{m_i} \mathbb{E}^{\pi,\sigma}_{\mathcal{G}}[\varrho_{i,j}] \geq u_{i,j}$. The total number of objectives is denoted by $\mathcal{N} \stackrel{\text{def}}{=} \sum_{i=1}^{n} m_i$. We denote by $\vec{u}_i$ the vector whose jth component is $u_{i,j}$ for $1 \leq j \leq m_i$ and by $\vec{u} = (\vec{u}_1, \ldots, \vec{u}_n) \in \mathbb{R}^{\mathcal{N}}$ the concatenation of all the $\vec{u}_i$ for $1 \leq i \leq n$. We use the same notational convention for other vectors (e.g. the vector of weights $\vec{x}$ below) and the reward structure $\vec{\rho}$. Given two vectors $\vec{u}, \vec{x} \in \mathbb{R}^{\mathcal{N}}$, we denote by $\vec{x} \cdot_n \vec{u} \stackrel{\text{def}}{=} (\vec{x}_1 \cdot \vec{u}_1, \cdots, \vec{x}_n \cdot \vec{u}_n)$.

Theorem 8. *Let $\mathcal{G}$ be a game, let $\vec{\varrho}_i : \Omega_{\mathcal{G}} \to \mathbb{R}^{m_i}$ be integrable functions, and $\vec{u}_i \in \mathbb{R}^{m_i}$, for $1 \leq i \leq n$ and let π be a Player $\Diamond$ strategy. The following propositions are equivalent:*

- *There exist non-zero weight vectors $\vec{x}_i \in \mathbb{R}^{m_i}_{\geq 0}$ for $1 \leq i \leq n$ such that π is winning for $\mathsf{E}(\vec{x} \cdot_n \vec{\varrho})(\vec{x} \cdot_n \vec{u})$;*
- *π is winning for $\psi = \bigwedge_{i=1}^{n} \bigvee_{j=1}^{m_i} \mathbb{E}[\varrho_{i,j}] \geq u_{i,j}$.*

Here winning means either winning against all strategies or winning against all finite memory strategies.

The theorem is a straightforward consequence of the following lemma that shows how disjunctions of expectation objectives reduce to single-dimensional expectation objectives.

Lemma 8. *Given a game $\mathcal{G}$, an integrable function $\vec{\varrho} : \Omega_{\mathcal{G}} \to \mathbb{R}^m$, a target $\vec{u} \in \mathbb{R}^m$, and a Player $\Diamond$ strategy π, there is a non-zero vector $\vec{x} \in \mathbb{R}^m_{\geq 0}$ such that $\varphi = \mathbb{E}^{\pi,\sigma}_{\mathcal{G}}[\vec{x} \cdot \vec{\varrho}] \geq \vec{x} \cdot \vec{u}$ holds for all (finite) σ if and only if $\psi = \bigvee_{j=1}^{m} \mathbb{E}^{\pi,\sigma}_{\mathcal{G}}[\varrho_j] \geq u_j$ holds for all (finite) σ.*

Proof. The proof method is based on a similar result in [19]. Fix a strategy π.

"If" direction. Assume π achieves ψ. Let $U \stackrel{\text{def}}{=} \mathsf{upc}(\{\vec{y} \in \mathbb{R}^m \mid \exists\sigma\,.\,\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{\varrho}] = \vec{y}\})$. Note that this set is convex. Indeed, for every two vectors $\vec{y}_1, \vec{y}_2 \in U$ and weight p one can construct a strategy for $p\vec{y}_1 + (1-p)\vec{y}_2$ by choosing, with the initial memory distribution, with probability p to play a strategy for $\vec{y}_1$ and with probability $1-p$ to play a strategy for $\vec{y}_2$. Moreover, the strategy is finite if constructed from finite strategies. Since π achieves ψ, there is a j satisfying $y_j \geq u_j$ for every $\vec{y} \in U$. We have that $\vec{u} \notin \text{int}(U)$ where $\text{int}(U)$ is the interior of U. Suppose otherwise, then there is $\varepsilon > 0$ s.t. $\vec{u} - \vec{\varepsilon} \in U$, contradicting that for all $\vec{y} \in U$ there is a j satisfying $y_j \geq u_j$ (take $\vec{y} = \vec{u} - \vec{\varepsilon}$ to derive the contradiction $u_j - \vec{\varepsilon} \geq u_j$). By the separating hyperplane theorem (Theorem 11.3 of [49]), there is a non-zero vector $\vec{x} \in \mathbb{R}^m$, such that for all $\vec{w} \in U$, $\vec{w} \cdot \vec{x} \geq \vec{u} \cdot \vec{x}$. We now show $\vec{x} \geq 0$. Assume for the sake of contradiction that $x_j < 0$ for some j. Take any $\vec{w} \in U$, let $d = \vec{w} \cdot \vec{x} - \vec{u} \cdot \vec{x} \geq 0$, and let $\vec{w}'$ be the vector obtained from $\vec{w}$ by replacing the jth coordinate with $w_j + \frac{d+1}{-x_j}$. Since $\frac{d+1}{-x_j}$ is positive and U is upwards closed in $\mathbb{R}^m$, we have $\vec{w}' \in U$. So

$$\vec{w}' \cdot \vec{x} = \sum_{h=1}^{m} w'_h \cdot x_h = -(d+1) + \sum_{h=1}^{m} w_h \cdot x_h = -(d+1) + \vec{w} \cdot \vec{x} = \vec{u} \cdot \vec{x} - 1,$$

implying $\vec{u} \cdot \vec{x} > \vec{w}' \cdot \vec{x}$, which contradicts $\vec{w}' \in U$.

Now fix a strategy σ. Since $\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{\varrho}] \in U$, it follows that $\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{x} \cdot \varrho] = \vec{x} \cdot \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{\varrho}] \geq \vec{x} \cdot \vec{u}$.

"Only If" direction. Assume there is a non-zero vector $\vec{x} \in \mathbb{R}^m_{\geq 0}$ such that π achieves φ. Assume for the sake of contradiction that π does not achieve ψ. Fix σ such that $\neg(\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\varrho_j] \geq u_j)$ for all j, which exists by assumption. Since $\vec{x}$ is such that π achieves φ, we have $\vec{x} \cdot \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{\varrho}] = \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{x} \cdot \vec{\varrho}] \geq \vec{x} \cdot \vec{u}$. Because $\vec{x}$ is non-zero and has no negative components, there must be a j such that $\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\varrho_j] \geq u_j$, a contradiction. □

The above theorem enables us to transfer results from conjunctions of linear combinations to Boolean combinations of objectives. In particular, we state below two transfer theorems, one for Pareto sets and the other for strategy synthesis.

For the remainder of this section we continue with the same notation as above. We now show how to compute, for every $\varepsilon > 0$, an ε-tight under-approximation of the Pareto set of ψ when one knows how to compute ε-tight under-approximations of the Pareto set of $\mathsf{E}(\vec{x} \cdot_n \vec{\varrho})$ and when the functions ϱ_{ij} are globally bounded by a constant B. We denote by $P_\varepsilon(\vec{x})$ an ε-tight under-approximation of $\text{Pareto}(\mathsf{E}(\vec{x} \cdot_n \vec{\varrho}))$. We define Grid, the set of vectors $\vec{x} \in [0, 1+\varepsilon/(4B)]^{\mathcal{N}}$, such that each $\vec{x}_i$ is non-zero, has norm satisfying $\|\vec{x}_i\|_\infty \in [1-\varepsilon/(4B), 1+\varepsilon/(4B)]$ and whose components are multiples of $\varepsilon/(4B)$.

The first transfer theorem, proved in Appendix C.2, is for Pareto sets.

Theorem 9. *The following set is an ε-tight under-approximation of the Pareto set of $\bigwedge_{i=1}^{n}\bigvee_{j=1}^{m_i}\mathbb{E}[\varrho_{i,j}]\geq u_{i,j}$:*

$$\bigcup_{\vec{x}\in\mathsf{Grid}}\{\vec{u}\in\mathbb{R}^{\mathcal{N}}\mid\vec{x}\cdot_n\vec{u}\in P_\varepsilon(\vec{x})\}.$$

The second transfer theorem deals with ε-optimal synthesis. Proof can be found in Appendix C.3.

Theorem 10. *If we know how to compute an ε-optimal strategy for $\mathsf{E}(\vec{x}\cdot_n\vec{\varrho})(\vec{v})$ for every $\vec{x}$, then we can compute an ε-optimal strategy for $\bigwedge_{i=1}^{n}\bigvee_{j=1}^{m}\mathbb{E}[\varrho_{i,j}]\geq u_{i,j}$.*

Another consequence of Theorem 8 is that synthesis for Pmp CQs enables us to synthesise strategies that are winning for Boolean combinations of expected ratio objectives against every finite strategy.

Theorem 11. *Let $\mathcal{G}$ be a game. For $1\leq i\leq n$, let $\vec{r}_i:S\to\mathbb{R}^{m_i}$ be m_i-dimensional reward structures, c_i be one-dimensional weakly positive reward structures, $\vec{u}_i\in\mathbb{R}^{m_i}$ and $\vec{x}_i\in\mathbb{R}^{m_i}_{\geq 0}$ non-null weight vectors. Let $\psi\stackrel{\text{def}}{=}\bigwedge_{i=1}^{n}\bigvee_{j=1}^{m_i}\mathbb{E}(\mathit{ratio}(r_{i,j}/c_i))\geq u_{i,j}$ and $\varphi_{\vec{x}}\stackrel{\text{def}}{=}\bigwedge_{i=1}^{n}\mathbb{P}(\mathit{mp}(\vec{x}_i\cdot\vec{r}_i-(\vec{x}_i\cdot\vec{u}_i)c_i)\geq 0)=1$. Every finite strategy winning for $\varphi_{\vec{x}}$ is winning for ψ against finite strategies. For every $\varepsilon>0$, there exists $\varepsilon'>0$ such that every ε'-optimal strategy for $\varphi_{\vec{x}}$ is ε-optimal for ψ against finite strategies.*

For the proof see Appendix C.4.

Example 7. *Consider the game $\mathcal{G}^1$ depicted in Figure 7 with the MQ $\varphi^1[(\frac{1}{4},\frac{9}{8})]=\mathit{Eratio}(r_1/c)(1/4)\vee\mathit{Eratio}(r_2/c)(9/8)$. Consider the weight vector $(1,\frac{2}{3})$ and define the single-objective reward structure r' by*

$$\begin{aligned}
r'(a)&=(1,\tfrac{2}{3})\cdot(r_1(a),r_2(a))-((1,\tfrac{2}{3})\cdot(\tfrac{1}{4},\tfrac{9}{8}))c(a)= 0\\
r'(b)&=(1,\tfrac{2}{3})\cdot(r_1(b),r_2(b))-((1,\tfrac{2}{3})\cdot(\tfrac{1}{4},\tfrac{9}{8}))c(b)=-\tfrac{1}{3}\\
r'(d)&=(1,\tfrac{2}{3})\cdot(r_1(d),r_2(d))-((1,\tfrac{2}{3})\cdot(\tfrac{1}{4},\tfrac{9}{8}))c(d)= \tfrac{2}{3},
\end{aligned}$$

and zero everywhere else. Then, by Theorem 11, every winning strategy for $\mathsf{Pmp}(r')$ is winning for $\varphi^1[(\frac{1}{4},\frac{9}{8})]$ against finite memory strategies. The optimal strategy for Player $\Diamond$ here clearly is to always take d. To spoil, the best Player $\Box$ can do is play b, but, due to the distribution, the expected number of times b is taken is at most $\sum_{k\geq 0}2^{-k}=2$ before a is taken again, balancing exactly the mean-payoff to zero. Hence, Player $\Diamond$ wins for $\mathsf{Pmp}(r')$, and also for $\varphi^1[(\frac{1}{4},\frac{9}{8})]$.

4.2. Emp Objectives in Controllable Multichain Games

We now consider synthesis of Boolean combinations of Emp objectives. Our methods are based on the observation that Pmp and Emp are equivalent in MECs of PAs. We define the class of *controllable multichain (CM)* games, in which Player $\Diamond$ can approximate any distribution between the possible MECs (cf. Lemma 12); therefore, we can construct strategies that induce PAs with a single MEC. Strategies synthesised for Pmp straightforwardly carry over to Emp (Remark 1). The main result of this section is a completeness result, showing that, if $\mathsf{Emp}(\vec{r})$ is ε-achievable by a finite DU strategy, then we can synthesise a ε-optimal strategy for $\mathsf{Pmp}(\vec{r})$.

First we note that, in the special case where an induced PA contains only a single MEC, achievability for Emp and Pmp coincide. The lemma is proved in Appendix C.5.

Lemma 9. *If a PA contains only one MEC, then it achieves $\mathsf{Emp}(\vec{r})$ against finite strategies if and only if it achieves $\mathsf{Pmp}(\vec{r})$ against finite strategies.*

We define, for each MEC $\mathcal{E}$ of an induced PA, the worst possible mean-payoff $\vec{\mathsf{z}}^{\mathcal{E}}$ as follows. Given an n-dimensional reward structure $\vec{r}$, and a MEC $\mathcal{E} = (V, U)$ of a PA $\mathcal{M}$, we define the vector $\vec{\mathsf{z}}^{\mathcal{E}} = (\mathsf{z}_1^{\mathcal{E}}, \dots, \mathsf{z}_n^{\mathcal{E}})$ by

$$\mathsf{z}_i^{\mathcal{E}} \stackrel{\text{def}}{=} \min_{t \in S_{\mathcal{E}}} \inf_{\sigma} \mathbb{E}_{\mathcal{E},t}^{\sigma}[\mathsf{mp}(r_i)] = \min_{t \in S_{\mathcal{E}}} \inf_{\sigma} \mathbb{E}_{\mathcal{E},t}^{\sigma}\left[\underline{\lim}_{N\to\infty} \frac{\mathsf{rew}^{N-1}(r_i)}{N}\right] \qquad (3)$$

Note that $\mathsf{Pmp}(\vec{r})$ is satisfied if and only if $\vec{\mathsf{z}}^{\mathcal{E}} \geq \vec{0}$ for every $\mathcal{E}$, because Player $\Box$ can reach any MEC with positive probability. A weaker condition is satisfied when $\mathsf{Emp}(\vec{r})$ is satisfied. In that case, there is a distribution γ over MECs, such that $\sum_{\mathcal{E}} \gamma(\mathcal{E})\vec{\mathsf{z}}^{\mathcal{E}} \geq \vec{0}$ (Lemma 10).

The idea underlying the definition of controllable multichain games (introduced below) is to make all the MECs of an induced PA almost-surely reachable from each other, so that then the distribution γ can be realised by Player $\Diamond$ by the frequencies of visits of each $\mathcal{E}$ in a new strategy, as formalised in Lemma 12. The strategy constructed ε-optimally achieves $\mathsf{Emp}(\vec{r})$, and induces a PA with a single MEC, and hence also satisfies $\mathsf{Pmp}(\vec{r})$ ε-optimally.

Lemma 10. *Let $\mathcal{M}$ be a finite PA for which $\mathsf{Emp}(\vec{r})$ is satisfied and let $\mathfrak{E}$ be the set of MECs in $\mathcal{M}$. Then there exists $\gamma \in \mathsf{D}(\mathfrak{E})$ such that $\sum_{\mathcal{E}\in\mathfrak{E}} \gamma(\mathcal{E})\vec{\mathsf{z}}^{\mathcal{E}} \geq \vec{0}$.*

This lemma is proved in Appendix C.6.

4.2.1. Controllable multichain games

A game $\mathcal{G}$ is *irreducible* if, for all finite DU Player $\Diamond$ strategies π, the induced PA $\mathcal{G}^{\pi}$ with states $S_{\mathcal{G}^{\pi}}$ and transitions $\Delta_{\mathcal{G}^{\pi}}$ forms a single MEC $(S_{\mathcal{G}^{\pi}}, \Delta_{\mathcal{G}^{\pi}})$. We define a *subgame* $\mathcal{H}$ of a game $\mathcal{G} = \langle S, (S_{\Diamond}, S_{\Box}, S_{\bigcirc}), \varsigma, \mathcal{A}, \chi, \Delta\rangle$ as a game $\langle S', (S'_{\Diamond}, S'_{\Box}, S'_{\bigcirc}), s'_{\mathsf{init}}, \mathcal{A}, \chi', \Delta'\rangle$, such that $S' \subseteq S$; $S'_{\Diamond} \subseteq S_{\Diamond}$; $S'_{\Box} \subseteq S_{\Box}$; $S'_{\bigcirc} \subseteq S_{\bigcirc}$; $s'_{\mathsf{init}} \in S'_{\Diamond} \cup S'_{\Box}$ is the unique initial state (the initial distribution is Dirac); $\chi' \subseteq \chi$; $\Delta' \subseteq \Delta$; and where $s \in S'$ if and only if s is reachable from s'_{init} via Δ'. A subgame $\mathcal{H}$ is *Player $\Box$-closed* if, for all $s \in S'_{\Box}$, all transitions $s \xrightarrow{a} \mu$ in $\mathcal{G}$ are also in $\mathcal{H}$, i.e. $s \xrightarrow{a}' \mu$, and so Player $\Box$ cannot escape from $\mathcal{H}$. An irreducible Player $\Box$-closed subgame of $\mathcal{G}$ is called an *irreducible component* (IC) of $\mathcal{G}$. A game $\mathcal{G}$ is a *controllable multichain* (CM) game if each IC $\mathcal{H}$ of $\mathcal{G}$ is reachable almost surely from any state $s \in S$ of $\mathcal{G}$, see Figure 11.

Theorem 12. *The problem of whether a game is CM is in $\mathsf{co\text{-}NP}$.*

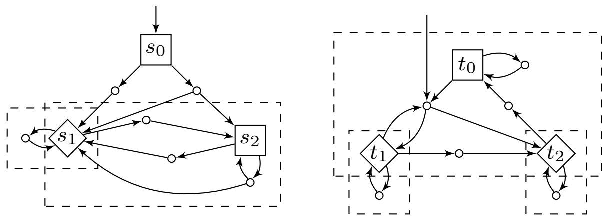


Figure 11: CM game (left) and non-CM game (right). ICs are annotated using dashed rectangles; note that ICs can overlap, as opposed to MECs in PAs. On the right, the IC containing only t_1 cannot be reached by Player $\Diamond$ from t_2; Player $\Diamond$ can achieve an expectation by randomising between t_1 and t_2, but, for almost-sure satisfaction, cooperation with Player $\Box$ is required to not loop in t_0.

Proof. A game is not a CM game if it has an IC $\mathcal{H}$ and a state $s \in S_{\mathcal{G}}$, such that $\mathcal{H}$ is not reachable almost surely from s. One can guess in polynomial time such a subgame $\mathcal{H}$ and a state s, and check in polynomial time whether $\mathcal{H}$ is an IC, and whether $\mathcal{H}$ is not reachable almost surely from s (Lemma 2). Hence, the problem lies in co-NP. $\Box$

The main property we use below is that, for any CM game and any finite DU strategy, MECs in the induced PA are almost surely reachable from everywhere in the PA. This property is in fact equivalent to the definition of CM games as stated in the following lemma. Given a MEC $\mathcal{E} = (S_{\mathcal{E}}, \Delta_{\mathcal{E}})$ of an induced PA $\mathcal{G}^{\pi}$, define the set $S_{\mathcal{G},\mathcal{E}}$ of $\mathcal{G}$-states s occurring in $\mathcal{E}$ (we recall that states of $\mathcal{E}$ are of the form $(s, \mathfrak{m})$ or $((s, s'), \mathfrak{m})$). We have the following lemma, which is proved in Appendix C.7.

Lemma 11. *A game $\mathcal{G}$ is a CM game if and only if, for every finite DU strategy π, for every MEC $\mathcal{E}$ of $\mathcal{G}^{\pi}$, $S_{\mathcal{G},\mathcal{E}}$ is almost surely reachable from every state of s.*

While a Player $\Diamond$ strategy π achieving an Emp CQ may randomise between several MECs, a strategy $\underline{\pi}$ for Pmp CQ must be winning in every reached MEC. Given a strategy π achieving $\mathsf{Emp}(\vec{r})$ in a CM game $\mathcal{G}$, we can construct a strategy $\underline{\pi}$ that ε-achieves $\mathsf{Pmp}(\vec{r})$, by inducing a single MEC in $\mathcal{G}^{\underline{\pi}}$, and simulating the distribution over the MECs in $\mathcal{G}^{\pi}$.

4.2.2. Strategy construction

We construct $\underline{\pi}$ by looping between MECs, where each MEC $\mathcal{E}_l$ is of a PA $\mathcal{G}^{\pi^l}$ and has an associated finite *step count* N_l.

Since $\mathcal{G}$ is CM, from each $s \in S_{\mathcal{G}}$, each MEC $\mathcal{E}$ can be reached almost surely by an MD strategy $\pi^{\mathcal{E}} : S \to S_{\bigcirc}$ (see Lemma 11 and Lemma 2). We first explain the intuition of our construction of $\underline{\pi}$. We start $\underline{\pi}$ by playing $\pi^{\mathcal{E}_1}$, the MD strategy to reach $\mathcal{E}_1$. As soon as $\mathcal{E}_1$ is reached, $\underline{\pi}$ switches to π^1, which is played for N_1 steps, that is, $\underline{\pi}$ stays inside $\mathcal{E}_1$ for N_1 steps. Then, from whatever

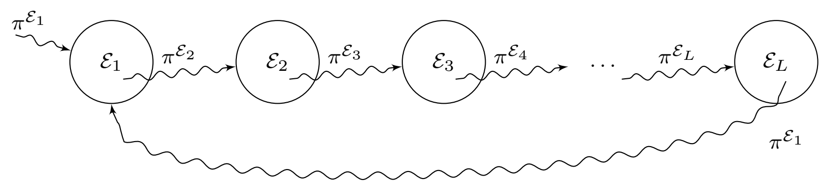


Figure 12: Illustrating the strategy $\underline{\pi}$ to simulate the distribution γ between MECs $\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_L$.

state s in $\mathcal{E}_1$ the game is in, $\underline{\pi}$ plays $\pi^{\mathcal{E}_2}$, and then in a similar fashion switches to π^2 for N_2 steps within $\mathcal{E}_2$. This continues until $\mathcal{E}_L$ is reached, at which point $\underline{\pi}$ goes back to $\mathcal{E}_1$ again. The strategy $\underline{\pi}$ keeps track in memory of whether it is going to a MEC $\mathcal{E}$, denoted $\triangleright\mathcal{E}$, or whether it is at a MEC $\mathcal{E}$ and has played j steps, denoted $j@\mathcal{E}$. We emphasise that the strategies are finite DU. See Figure 12 for an illustration of $\underline{\pi}$.

Definition 7. *Let $\pi^l = \langle \mathfrak{M}^l, \pi^l_{\mathsf{c}}, \pi^l_{\mathsf{u}}, \pi^l_{\mathsf{d}} \rangle$ be finite DU Player $\Diamond$ strategies, for $1 \leq l \leq L$, with respective MECs $\mathcal{E}_l$ and step counts N_l. The* step strategy *$\underline{\pi}$ is defined as $\langle \underline{\mathfrak{M}}, \underline{\pi}_{\mathsf{c}}, \underline{\pi}_{\mathsf{u}}, \underline{\pi}_{\mathsf{d}} \rangle$, where*

$$\underline{\mathfrak{M}} \stackrel{\text{def}}{=} (\mathfrak{M} \times \{ j@\mathcal{E}_l \mid l \leq L, j \leq N_l \}) \cup \textstyle\bigcup_{l=1}^{L} \{ \triangleright\mathcal{E}_l \},$$

and where, for all $s, t, u \in S_{\mathcal{G}}$, $l \leq L$, $j \leq N_l$, and $\mathfrak{m} \in \mathfrak{M}$,

$$\underline{\pi}_{\mathsf{d}}(s) \stackrel{\text{def}}{=} \begin{cases} \triangleright\mathcal{E}_1 & \text{if } s \notin S_{\mathcal{G},\mathcal{E}_1} \\ (\pi^1_{\mathsf{d}}(s), 0@\mathcal{E}_1) & \text{if } s \in S_{\mathcal{G},\mathcal{E}_1} \end{cases}$$

$$\underline{\pi}_{\mathsf{u}}(\triangleright\mathcal{E}_l, t) \stackrel{\text{def}}{=} \begin{cases} \triangleright\mathcal{E}_l & \text{if } t \notin S_{\mathcal{G},\mathcal{E}_l} \\ (\pi^l_{\mathsf{d}}(t), 0@\mathcal{E}_l) & \text{if } t \in S_{\mathcal{G},\mathcal{E}_l} \end{cases}$$

$$\underline{\pi}_{\mathsf{u}}((\mathfrak{m}, j@\mathcal{E}_l), s) \stackrel{\text{def}}{=} \begin{cases} \triangleright\mathcal{E}_{l'} & \text{if } j = N_l \text{ and } l' = 1 + (l \mod L) \\ (\pi^l_{\mathsf{u}}(\mathfrak{m}, s), j+1@\mathcal{E}_l) & \text{if } j < N_l \end{cases}$$

$$\underline{\pi}_{\mathsf{c}}(s, \triangleright\mathcal{E}_l)(u) \stackrel{\text{def}}{=} \pi^{\mathcal{E}_l}(s)(u)$$

$$\underline{\pi}_{\mathsf{c}}(s, (\mathfrak{m}, j@\mathcal{E}_l))(t) \stackrel{\text{def}}{=} \pi^l(s, \mathfrak{m})(t).$$

The following lemma justifies that, for appropriate choices of the step counts N_l, the strategy $\underline{\pi}$ approximates a distribution between MECs of $\mathcal{G}^{\pi}$, while only inducing a single MEC in $\mathcal{G}^{\underline{\pi}}$.

Lemma 12. *Let $\mathcal{G}$ be a CM game, let π^l be finite DU strategies with associated MECs $\mathcal{E}_l$ of $\mathcal{G}^{\pi^l}$, for $1 \leq l \leq L$, and let $\mathfrak{E}$ be the set of MECs $\{\mathcal{E}_l \mid 1 \leq l \leq L\}$. Then, for all $\gamma \in \mathsf{D}(\mathfrak{E})$ and $\varepsilon > 0$, there exists a finite DU strategy $\underline{\pi}$ such that $\mathcal{G}^{\underline{\pi}}$ contains only one MEC, and for all finite Player $\Box$ strategies σ*

$$\mathbb{E}^{\underline{\pi},\sigma}_{\mathcal{G}}[\mathsf{mp}(\vec{r})] \geq \sum_{\mathcal{E} \in \mathfrak{E}} \gamma(\mathcal{E}) \vec{z}^{\mathcal{E}} - \varepsilon.$$

For the proof of the above lemma see Appendix C.8.

We now show in Theorem 13, the main result of this section, that in CM games, for any $\varepsilon > 0$, we can find a strategy $\underline{\pi}$ that achieves $\mathsf{Pmp}(\vec{r}+\varepsilon)$ whenever $\mathsf{Emp}(\vec{r})$ is achievable. The ε degradation is unavoidable for finite strategies, due to the need for infinite memory in general, see Figure 6. Here, the strategy $\underline{\pi}$ has to minimise the transient contribution, which only vanishes if the step counts N_l go to infinity.

Theorem 13. *In CM games, it holds that*

$$\text{Pareto}_{\text{FDU,FSU}}(\mathit{Emp}(\vec{r})) = \text{Pareto}_{\text{FDU}}(\mathit{Emp}(\vec{r})) = \text{Pareto}(\mathit{Pmp}(\vec{r}))$$

Proof. By Theorem 6 and Remark 1 it holds that

$$\text{Pareto}(\mathsf{Pmp}(\vec{r})) = \text{Pareto}_{\text{FDU}}(\mathsf{Pmp}(\vec{r})) \subseteq \text{Pareto}_{\text{FDU}}(\mathsf{Emp}(\vec{r})) \subseteq \text{Pareto}_{\text{FDU,FSU}}(\mathsf{Emp}(\vec{r})).$$

It then remains to show that $\text{Pareto}_{\text{FDU,FSU}}(\mathsf{Emp}(\vec{r})) \subseteq \text{Pareto}(\mathsf{Pmp}(\vec{r}))$. For this purpose, it suffices to show that if a finite DU strategy π achieves $\mathsf{Emp}(\vec{r})$ against finite Player $\Box$ strategies then, for all $\varepsilon > 0$, there is a finite DU strategy $\underline{\pi}$ achieving $\mathsf{Pmp}(\vec{r}+\vec{\varepsilon})$. We find a winning strategy $\underline{\pi}$ such that the induced PA $\mathcal{G}^{\underline{\pi}}$ contains only a single MEC (which is reached w.p. 1, potentially via some transient states). Then we apply Lemma 9 to conclude that $\underline{\pi}$ also wins for Pmp. Let $\varepsilon > 0$ and let π be a finite DU strategy such that $\mathcal{G}^{\pi} \models \mathsf{Emp}(\vec{r})$. The induced PA $\mathcal{G}^{\pi}$ contains a set $\mathfrak{E}$ of L MECs. If $L = 1$, we let $\underline{\pi} = \pi$. If, on the other hand, $L > 1$, we construct a strategy $\underline{\pi}$ such that $\mathcal{G}^{\underline{\pi}} \models \mathsf{Emp}(\vec{r} + \vec{\varepsilon})$ as follows. First, from Lemma 10, we obtain a distribution γ such that $\sum_{\mathcal{E}\in\mathfrak{E}} \gamma(\mathcal{E})\vec{\mathsf{z}}^{\mathcal{E}} \geq \vec{0}$. We then apply Lemma 12 with $\pi^l = \pi$ for each MEC $\mathcal{E}_l \in \mathcal{G}^{\pi}$, to find a strategy $\underline{\pi}$, so that $\mathcal{G}^{\underline{\pi}}$ contains only one MEC, and for all finite Player $\Box$ strategies σ, it holds that

$$\mathbb{E}_{\mathcal{G}}^{\underline{\pi},\sigma}[\mathsf{mp}(\vec{r})] \geq \sum_{l=1}^{L} \gamma(\mathcal{E}_l)\vec{\mathsf{z}}^{\mathcal{E}_l} - \varepsilon \geq -\varepsilon.$$

We conclude that $\underline{\pi}$ achieves $\mathsf{Pmp}(\vec{r} + \vec{\varepsilon})$ using Lemma 9. □

4.2.3. Emp MQs in CM Games

We can now summarise the results of this section in Theorem 14, which allows us to synthesise ε-optimal strategies for Emp objectives in CM games.

Theorem 14. *In CM games with Emp MQs ψ, one can solve the two following problems using the algorithms for Pmp CQs.*

1. *Compute an ε-tight under-approximation of the Pareto set* $\text{Pareto}_{\text{FDU,FSU}}(\psi)$.
2. *Synthesise a strategy winning against every finite strategy for every target vector $\vec{u}$ such that $\vec{u} + \vec{\varepsilon} \in \text{Pareto}_{\text{FDU,FSU}}(\psi)$.*

Proof. 1. According to Theorem 9, it suffices to determine, for every $\vec{x} \in \mathsf{Grid}$, an ε-tight under-approximation of $\text{Pareto}_{\text{FDU,FSU}}(\mathsf{E}(\vec{x} \cdot_n \mathsf{mp}(\vec{r})))$. By

Proposition 13 and Theorem 13 we have $\text{Pareto}_{\text{FDU,FSU}}(\mathsf{E}(\vec{x} \cdot_n \mathsf{mp}(\vec{r}))) = \text{Pareto}_{\text{FDU,FSU}}(\mathsf{Emp}(\vec{x} \cdot_n \vec{r})) = \text{Pareto}(\mathsf{Pmp}(\vec{x} \cdot_n \vec{r}))$. By Theorem 5, ε-tight under-approximation can be computed for these sets.

2. According to Theorem 10, it suffices to solve the synthesis problem for $\mathsf{E}(\vec{x} \cdot_n \mathsf{mp}(\vec{r}))$. Take a vector in $\text{Pareto}_{\text{FDU,FSU}}(\mathsf{E}(\vec{x} \cdot_n \mathsf{mp}(\vec{r}))) = \text{Pareto}(\mathsf{Pmp}(\vec{x} \cdot_n \vec{r}))$, then with Algorithm 1 we synthesise a finite strategy winning for $\mathsf{Pmp}(\vec{x} \cdot_n \vec{r})(-\varepsilon)$, and hence for $\mathsf{Emp}(\vec{x} \cdot_n \vec{r})(-\varepsilon)$. This strategy is also winning against any finite Player $\Box$ strategy for $\mathsf{E}(\vec{x} \cdot_n \mathsf{mp}(\vec{r}))$ thanks to Proposition 13.

$\Box$

5. Compositional Strategy Synthesis

In this section we develop our framework for compositional strategy synthesis. We consider synthesis rules of the form

$$\frac{(\mathcal{G}^i)^{\pi^i} \models \bigwedge_{j=1}^{m} \varphi_j^i \qquad i \in I}{(\|_{i \in I} \mathcal{G}^i)^{\|_{i \in I} \pi^i} \models \varphi,}$$

which hold for all Player $\Diamond$ strategies π^i. Thus, strategies π^i synthesised for the components $\mathcal{G}^i$ yield a strategy $\|_{i \in I} \pi^i$ for the composed game $\|_{i \in I} \mathcal{G}^i$. We develop the game and strategy composition operators ($\|$), and show how to instantiate sound synthesis rules. Note that, in this section, we allow deadlocks in the composed games.

5.1. Game Composition

We provide a synchronising composition of games so that controllability is preserved for Player $\Diamond$, that is, actions controlled by Player $\Diamond$ in the components are controlled by Player $\Diamond$ in the composition. Our composition is inspired by interface automata [24], which have a natural interpretation as (concurrent) games. Each component game is endowed with an alphabet of actions $\mathcal{A}$, where synchronisation on *shared actions* in $\mathcal{A}^1 \cap \mathcal{A}^2$ is viewed as a (blocking) communication over ports, as in interface automata, though for simplicity we do not distinguish inputs and outputs. Synchronisation is multi-way and we do not impose input-enabledness of IO automata [21]. Strategies can choose between moves, and so, within a component, nondeterminism in Player $\Diamond$ states is completely controlled by Player $\Diamond$. In our game composition, synchronisation is over actions only, and hence the choice between several moves with the same action is hidden to other components.

5.1.1. Normal form of a game

The first step of our composition ensures games are in normal form. We can transform every game into its corresponding normal form, which does not affect achievability of specifications defined on traces.

Definition 8. *A game is in* normal form *if every τ-transition $s \xrightarrow{\tau} \mu$ is from a* Player $\Box$ *state s to a* Player $\Diamond$ *state s' with a Dirac distribution $\mu = s'$; and every* Player $\Diamond$ *state s can only be reached by an incoming move (τ, s).*

In particular, every distribution μ of a non-τ-transition, as well as the initial distribution, assigns probability zero to all Player $\Diamond$ states. Given a game $\mathcal{G}$ without τ-transitions, one can construct its normal form by splitting every state $s \in S_\Diamond$ into a Player $\Box$ state $\overline{s}$ and a Player $\Diamond$ state $\underline{s}$, such that (a) the incoming (resp. outgoing) moves of $\overline{s}$ (resp. $\underline{s}$) are precisely the incoming (resp. outgoing) moves of s, with every Player $\Diamond$ state $t \in S_\Diamond$ replaced by $\bar{t}$; and (b) the only outgoing (resp. incoming) move of $\overline{s}$ (resp. $\underline{s}$) is $(\tau, \underline{s})$. Intuitively, at $\overline{s}$ the game is idle until Player $\Box$ allows Player $\Diamond$ to choose a move in $\underline{s}$. Hence, any strategy for a game carries over naturally to its normal form, and for specifications defined on traces we can operate w.l.o.g. with normal-form games. Also, τ can be considered as a scheduling choice. In the transformation to normal form, at most one such scheduling choice is introduced for each Player $\Box$ state, but in the composition several scheduling choices may be present at a Player $\Box$ state, so that Player $\Box$ resolves nondeterminism arising from concurrency.

5.1.2. Composition

Given games $\mathcal{G}^i$, $i \in I$, in normal form with respective player states $S^i_\Diamond \cup S^i_\Box$, the set of player states $S_\Diamond \cup S_\Box$ of the composition is a subset of the Cartesian product $\prod_{i\in I} S^i_\Diamond \cup S^i_\Box$. Due to the normal form, each state $\vec{s} \in S_\Diamond \cup S_\Box$ contains either no Player $\Diamond$ component, in which case $\vec{s} \in S_\Box$, or contains exactly one Player $\Diamond$ component, in which case $\vec{s} \in S_\Diamond$. We denote by s^i the ith component of $\vec{s} \in \prod_{i\in I} S^i$. We denote by $\vec{\mu}$ the *product distribution* of $\mu^i \in \mathsf{D}(S^i)$ for $i \in I$, defined on $\prod_{i\in I} S^i$ by $\vec{\mu}(\vec{s}) \stackrel{\text{def}}{=} \prod_{i\in I} \mu^i(s^i)$. We say that a transition $\vec{s} \xrightarrow{a} \vec{\mu}$ *involves* the ith component if $s^i \xrightarrow{a}{}^i \mu^i$, otherwise the state remains the same $\mu^i(s^i) = 1$. Note that, due to the normal form, $\vec{\varsigma}$ for the composed game is such that $\mathsf{supp}(\vec{\varsigma}) \subseteq S_\Box$. We define the set of actions *enabled* in a state s by $\mathsf{En}(s) \stackrel{\text{def}}{=} \{a \in \mathcal{A} \mid \exists \mu \,.\, s \xrightarrow{a} \mu\}$.

Definition 9. *Given normal-form games $\mathcal{G}^i = \langle S^i, (S^i_\Diamond, S^i_\Box, S^i_\bigcirc), \varsigma^i, \mathcal{A}^i, \chi^i, \Delta^i\rangle$, $i \in I$, their* composition *is the game $\|_{i\in I}\, \mathcal{G}^i \stackrel{\text{def}}{=} \langle S, (S_\Diamond, S_\Box, S_\bigcirc), \prod_{i\in I} \varsigma^i, \bigcup_{i\in I} \mathcal{A}^i, \chi, \Delta\rangle$, where the sets of* Player $\Diamond$ *and* Player $\Box$ *states*

$$S_\Diamond \subseteq \{\vec{s} \in \prod_{i\in I}(S^i_\Diamond \cup S^i_\Box) \mid \exists! \iota \,.\, s^\iota \in S^\iota_\Diamond\} \qquad \textit{and} \qquad S_\Box \subseteq \prod_{i\in I} S^i_\Box,$$

are defined inductively to contain the reachable states, where $S_\bigcirc$, χ, and Δ are defined via

- *$\vec{s} \xrightarrow{a} \vec{\mu}$ for $a \neq \tau$ if at least one component is involved and the involved components are exactly those with a in their action alphabet, and if $\vec{s}$ is a* Player $\Diamond$ *state then its only* Player $\Diamond$ *component $\mathcal{G}^\iota$ is involved; and*
- *$\vec{s} \xrightarrow{\tau} \vec{t}$ if exactly one component $\mathcal{G}^i$ is involved, $\vec{s} \in S_\Box$, and $\mathsf{En}(\vec{t}) \neq \emptyset$.*

We take the view that the identity of the players must be preserved through composition to facilitate synthesis, and thus Player $\Diamond$ actions of the individual components are controlled by a single Player $\Diamond$ in the composition. Player $\Box$ in the composition acts as a scheduler, controlling which component advances and, in Player $\Box$ states, selecting among available actions, whether synchronised or not. Synchronisation in Player $\Diamond$ states means that Player $\Diamond$ in one component may indirectly control some Player $\Box$ actions in another component. In particular, we can impose assume-guarantee contracts at the component level, so that Player $\Diamond$ of different components can cooperate to achieve a common goal: in one component Player $\Diamond$ satisfies the goal B under an assumption A on its environment behaviour (i.e. $A \to B$), while Player $\Diamond$ in the other component ensures that the assumption is satisfied, against all Player $\Box$ strategies.

Under specifications defined on traces, our game composition is both associative and commutative, facilitating a modular model development. We define the relation $\simeq$ between games so that $\mathcal{G}^1 \simeq \mathcal{G}^2$ means that, for all specifications φ defined on traces, $\mathcal{G}^1 \models \varphi$ if and only if $\mathcal{G}^2 \models \varphi$.

Proposition 11. *Given normal-form games $\mathcal{G}^1$, $\mathcal{G}^2$ and $\mathcal{G}^3$, we have $\mathcal{G}^1 \parallel \mathcal{G}^2 \simeq \mathcal{G}^2 \parallel \mathcal{G}^1$ (*commutativity*), and $(\mathcal{G}^1 \parallel \mathcal{G}^2) \parallel \mathcal{G}^3 \simeq \mathcal{G}^1 \parallel (\mathcal{G}^2 \parallel \mathcal{G}^3) \models \varphi$ (*associativity*).*

Our composition is closely related to PA composition [52], with the added condition that in Player $\Diamond$ states the Player $\Diamond$ component must be involved. As PAs are games without Player $\Diamond$ states, the game composition restricted to PAs is the same as classical PA composition. The condition $\mathsf{En}(\vec{t}) \neq \emptyset$ for τ-transitions ensures that a Player $\Diamond$ state is never entered if it were to result in deadlock introduced by the normal form transformation. Deadlocks that were present before the transformation are still present in the normal form. In the composition of normal form games, τ-transitions are only enabled in Player $\Box$ states, and Player $\Diamond$ states are only reached by such transitions; hence, composing normal form games yields a game in normal form.

Example 8. *The game in Figure 7 (right) is the composition of the two games on the left, which are already in normal form. Actions a and b are synchronised. Player $\Box$ controls b in both s_1 and t_1, and so in the composition Player $\Box$ controls b at (s_1, t_1). Player $\Box$ controls a in s_1 and s_2, but Player $\Diamond$ controls a in t_0, and so it is controlled by Player $\Diamond$ in (s_1, t_0) and (s_2, t_0) in the composition.*

5.2. Strategy Composition

For compositional synthesis, we assume the following compatibility condition on component games, which is analogous to that for single-threaded interface automata [24]: we require that moves controlled by Player $\Diamond$ in one game are enabled and fully controlled by Player $\Diamond$ in the composition.

Definition 10. *Games $(\mathcal{G}^i)_{i \in I}$ are* compatible *if, for every Player $\Diamond$ state $\vec{s} \in S_\Diamond$ in the composition with $s^\iota \in S^\iota_\Diamond$, if $s^\iota \xrightarrow{a}{}^\iota \mu^\iota$ then there is exactly one distribution $\vec{\nu}$, denoted by $\langle \mu^\iota \rangle_{\vec{s},a}$, such that $\vec{s} \xrightarrow{a} \vec{\nu}$ and $\nu^\iota = \mu^\iota$. (That is, for $i \neq \iota$ such that $a \in \mathcal{A}^i$, there exists exactly one a-transition enabled in s^i.)*

5.2.1. Composing SU strategies

The memory update function of the composed SU strategy ensures that the memory in the composition is the same as if the SU strategies were applied to the games individually. We assume w.l.o.g. that from the current memory element $\mathfrak{m}$ one can recover the current state denoted $s(\mathfrak{m})$. We let $\Gamma(\vec{\mathfrak{m}}, \vec{s})$ be the set of indices of components that update their memory according to a new (stochastic) state formally defined by

$$\Gamma(\vec{\mathfrak{m}}, \vec{s}) = \begin{cases} \{i \in I \mid s^i \neq s(\mathfrak{m}^i)\} & \text{if } \vec{s} \in S_\Diamond \cup S_\Box \\ \{i \in I \mid a \in \mathcal{A}_i\} & \text{if } \vec{s} = (a, \vec{\mu}) \in S_\bigcirc \text{ s.t. } a \neq \tau \\ \{\iota\} & \text{if } \vec{s} = (\tau, \vec{t}) \in S_\bigcirc \text{ s.t. } s^\iota \in S^\iota_\Diamond \end{cases}$$

Definition 11. *The* composition *of* Player $\Diamond$ *strategies* $\pi^i = \langle \mathfrak{M}^i, \pi^{u,i}, \pi^{c,i}, \pi^{d,i} \rangle$, $i \in I$*, for compatible games is* $\|_{i\in I}\, \pi^i \stackrel{\text{def}}{=} \langle \prod_{i\in I} \mathfrak{M}^i, \pi_c, \pi_u, \pi_d \rangle$*, where*

$$\pi_c(\vec{s}, \vec{\mathfrak{m}})(a, \langle \mu^\iota \rangle_{\vec{s},a}) \stackrel{\text{def}}{=} \pi^{c,\iota}(s^\iota, \mathfrak{m}^\iota)(a, \mu^\iota) \qquad \textit{whenever } s^\iota \in S^\iota_\Diamond$$

$$\pi_u(\vec{\mathfrak{m}}, \vec{s})(\vec{\mathfrak{n}}) \stackrel{\text{def}}{=} \prod_{i\in\Gamma(\vec{\mathfrak{m}},\vec{s})} \pi^{u,i}(\mathfrak{m}^i, s^i)(\mathfrak{n}^i) \qquad \textit{whenever } \mathfrak{m}^i = \mathfrak{n}^i \textit{ for } i \neq \Gamma(\vec{\mathfrak{m}}, \vec{s})$$

$$\pi_d(\vec{s}) \stackrel{\text{def}}{=} \prod_{i\in I} \pi^{d,i}(s^i).$$

Remark 3. *In the above definition, product was defined on SU strategies, as this provides a more compact encoding than with DU strategies. Recall that SU and DU strategies are equally powerful (Proposition 1). For the proofs one can consider w.l.o.g. only products of DU strategies since, when given SU strategies, the product of their determinisation equals the determinisation of their product.*

Strategy composition is commutative and associative.

Proposition 12. *Given compatible normal-form games* $\mathcal{G}^1$*,* $\mathcal{G}^2$ *and* $\mathcal{G}^3$*, and strategies* π^1*,* π^2 *and* π^3*, we have* $(\mathcal{G}^1 \parallel \mathcal{G}^2)^{\pi^1\|\pi^2} \simeq (\mathcal{G}^2 \parallel \mathcal{G}^1)^{\pi^2\|\pi^1}$ *(*commutativity*), and* $((\mathcal{G}^1 \parallel \mathcal{G}^2) \parallel \mathcal{G}^3)^{(\pi^1\|\pi^2)\|\pi^3} \simeq (\mathcal{G}^1 \parallel (\mathcal{G}^2 \parallel \mathcal{G}^3))^{\pi^1\|(\pi^2\|\pi^3)}$ *(*associativity*).*

Note that strategy composition can be implemented efficiently by storing the individual strategies and selecting the next move and memory update of the strategies corresponding to the components $\Gamma(\mathsf{act}(\vec{\mathfrak{m}}), \vec{s})$ involved in the respective transitions.

5.3. Properties of the Composition

We now show that synthesising strategies for compatible individual components is sufficient to obtain a composed strategy for the composed game.

5.3.1. Functional simulations

We introduce functional simulations, which are a special case of classical PA simulations [52], and show that they preserve specifications over traces. Intuitively, a PA $\mathcal{M}'$ functionally simulates a PA $\mathcal{M}$ if all behaviours of $\mathcal{M}$ are present in $\mathcal{M}'$, and if strategies translate from $\mathcal{M}$ to $\mathcal{M}'$. Given a distribution μ, and a partial function $\mathcal{F} : S \to S'$ defined on the support of μ, we write $\overline{\mathcal{F}}(\mu)$ for the distribution defined by $\overline{\mathcal{F}}(\mu)(s') \stackrel{\text{def}}{=} \sum_{\mathcal{F}(s)=s'} \mu(s)$.

Definition 12. *A* functional simulation *from a PA $\mathcal{M}$ to a PA $\mathcal{M}'$ is a partial function $\mathcal{F} : S \to S'$ such that*

(F1) $\overline{\mathcal{F}}(\varsigma) = \varsigma'$*; and*

(F2) if $s \xrightarrow{a} \mu$ *in* $\mathcal{M}$ *then* $\mathcal{F}(s) \xrightarrow{a}{}' \overline{\mathcal{F}}(\mu)$ *in* $\mathcal{M}'$*.*

Lemma 13. *Given a functional simulation from a PA $\mathcal{M}$ to a PA $\mathcal{M}'$ and a specification φ defined on traces, for every (finite) strategy σ there is a (finite) strategy σ' such that $(\mathcal{M}')^{\sigma'} \models \varphi \Leftrightarrow \mathcal{M}^{\sigma} \models \varphi$.*

We have included the proof of this lemma in Appendix D.1.

5.3.2. From PA composition to game composition

When synthesising a strategy π^i for each component $\mathcal{G}^i$, we can induce the PAs $(\mathcal{G}^i)^{\pi^i}$, and compose them to obtain the composed PA $\|_{i\in I} (\mathcal{G}^i)^{\pi^i}$. However, in our synthesis rule we are interested in the PA $(\|_{i\in I} \mathcal{G}^i)^{\|_{i\in I}\pi^i}$, which is constructed by first composing the individual components, and then applying the composed Player $\Diamond$ strategy. The following lemma exhibits a functional simulation between such PAs, which together with Lemma 13 allows us to develop our synthesis rules for specifications defined on traces.

Lemma 14. *Given compatible normal form games $(\mathcal{G}^i)_{i\in I}$, and Player $\Diamond$ strategies $(\pi^i)_{i\in I}$, there is a functional simulation from $(\|_{i\in I} \mathcal{G}^i)^{\|_{i\in I}\pi^i}$ to $\|_{i\in I} (\mathcal{G}^i)^{\pi^i}$.*

Proof can be found in Appendix D.2. In general, there is no simulation in the other direction, as in the PA composition states that were originally Player $\Diamond$ states can no longer be distinguished.

5.4. Composition Rules

Our main result for compositional synthesis is that any verification rule for PAs gives rise to a synthesis rule for games with the same side conditions, shown in Theorem 15 below. The idea is to induce PAs from the games using the synthesised strategies, apply PA rules, and, using Lemma 14, lift the result back into the game domain.

Theorem 15. *Given a rule $\mathfrak{P}$ for PAs $\mathcal{M}^i$ and specifications φ^i_j and φ defined on traces, then the rule $\mathfrak{G}$ holds for all Player $\Diamond$ strategies π^i of compatible games $\mathcal{G}^i$ with the same action alphabets as the corresponding PAs, where*

$$\mathfrak{P} \equiv \frac{\mathcal{M}^i \models \varphi^i_j \quad j \in J \quad i \in I}{\|_{i\in I} \mathcal{M}^i \models \varphi,} \quad \textit{and} \quad \mathfrak{G} \equiv \frac{(\mathcal{G}^i)^{\pi^i} \models \bigwedge_{j\in J} \varphi^i_j \quad i \in I}{(\|_{i\in I} \mathcal{G}^i)^{\|_{i\in I}\pi^i} \models \varphi.}$$

Proof. For all $i \in I$, let $\mathcal{G}^i$ be games, and let π^i be respective Player $\Diamond$ strategies such that $(\mathcal{G}^i)^{\pi^i} \models \bigwedge_{j\in J} \varphi^i_j$. By applying the PA rule $\mathfrak{P}$ with the PAs $\mathcal{M}^i \stackrel{\text{def}}{=} (\mathcal{G}^i)^{\pi^i}$, where $\mathcal{M}^i \models \bigwedge_{j\in J} \varphi^i_j$ for all $i \in I$ from how the strategies π^i were picked, we have that $\|_{i\in I} \mathcal{M}^i \models \varphi$. From Lemma 14, there is a functional simulation from $(\|_{i\in I} \mathcal{G}^i)^{\|_{i\in I}\pi^i}$ to $\|_{i\in I} (\mathcal{G}^i)^{\pi^i}$. Since $\|_{i\in I} (\mathcal{G}^i)^{\pi^i} \models \varphi$, applying Lemma 13 yields $(\|_{i\in I} \mathcal{G}^i)^{\|_{i\in I}\pi^i} \models \varphi$. □

Monolithic synthesis is performed for components $\mathcal{G}^i$, $i \in I$, by obtaining for each i a $\mathsf{Player}\ \Diamond$ strategy π^i for $\mathcal{G}^i \models \bigwedge_{j \in J} \varphi^i_j$. We apply $\mathfrak{P}$ with $\mathcal{M}_i \stackrel{\text{def}}{=} (\mathcal{G}^i)^{\pi^i}$ (which never has to be explicitly constructed) to deduce that $\|_{i \in I}\ \pi^i$ is a winning strategy for $\mathsf{Player}\ \Diamond$ in $\|_{i \in I}\ \mathcal{G}^i$. The rules can be applied recursively, making use of associativity of the game and strategy composition.

Note that, for each choice, the composed strategy takes into account the history of only one component, which is less general than using the history of the composed game. Hence, it may be possible that a specification is achievable in the composed game, while it is not achievable compositionally. Our rules are therefore sound but not complete, even if the PA rules $\mathfrak{P}$ are complete.

5.4.1. Verification rules for PAs

We develop PA assume-guarantee rules for specifications defined on traces. Our rules are based on those in [36], but we emphasise that Theorem 15 is applicable to any PA rule. Given a composed PA $\mathcal{M} =\|_{i \in I} \mathcal{M}^i$, a strategy σ is *fair* if each component makes progress infinitely often with probability 1 [2]. We write $\mathcal{M} \models^{\mathsf{u}} \varphi$ if, for all fair strategies $\sigma \in \Sigma$, $\mathcal{M}^\sigma \models \varphi$. Note that a specification defined on traces remains defined on traces under fairness. In games, fairness is imposed only on $\mathsf{Player}\ \Box$, and for a single component fairness is equivalent to requiring deadlock-freedom. Our game composition does not guarantee freedom from deadlocks, that is, states without outgoing moves. However fair, $\mathsf{Player}\ \Box$ strategies avoid reaching deadlocks and hence yield induced DTMC without deadlocks. If deadlocks are unavoidable then the set of fair $\mathsf{Player}\ \Box$ strategies is empty; in that case the synthesis problem is trivial: every $\mathsf{Player}\ \Diamond$ strategy satisfies any specification under fairness.

Theorem 16. *Given compatible PAs $\mathcal{M}^1$ and $\mathcal{M}^2$, specifications φ^G, φ^{G_1}, φ^{G_2}, φ^{A_1} and φ^{A_2} defined on traces of $\mathcal{A}_G, \mathcal{A}_{G_1} \subseteq \mathcal{A}^1$, $\mathcal{A}_{G_2} \subseteq \mathcal{A}^2$, $\mathcal{A}_{A_1}, \mathcal{A}_{A_2} \subseteq \mathcal{A}^1 \cap \mathcal{A}^2$, then the following rules are sound:*

$$(\textsc{Conj})\frac{\begin{array}{ll}\mathcal{M}^1 & \models^u \varphi^{G_1} \\ \mathcal{M}^2 & \models^u \varphi^{G_2}\end{array}}{\mathcal{M}^1 \parallel \mathcal{M}^2 \models^u \varphi^{G_1} \wedge \varphi^{G_2}} \quad \textit{and} \quad (\textsc{Asym})\frac{\begin{array}{ll}\mathcal{M}^1 & \models^u (\varphi^{A_1} \rightarrow \varphi^G) \wedge \varphi^{A_2} \\ \mathcal{M}^2 & \models^u \varphi^{A_2} \rightarrow \varphi^{A_1}\end{array}}{\mathcal{M}^1 \parallel \mathcal{M}^2 \models^u \varphi^G}.$$

Proof. Let $\mathcal{M} = \mathcal{M}^1 \parallel \mathcal{M}^2$. We first recall concepts of projections from [36]. Given a state $s = (s^1, s^2)$ of $\mathcal{M}$, the projection of s onto $\mathcal{M}^i$ is $s{\upharpoonright}_{\mathcal{M}^i} \stackrel{\text{def}}{=} s^i$, and for a distribution μ over states of $\mathcal{M}$ we define its projection by $\mu{\upharpoonright}_{\mathcal{M}^i}(s^i) \stackrel{\text{def}}{=} \sum_{s{\upharpoonright}_{\mathcal{M}^i} = s^i} \mu(s)$. Given a path λ of $\mathcal{M}$, the projection of λ onto $\mathcal{M}^i$, denoted by $\lambda{\upharpoonright}_{\mathcal{M}^i}$, is the path obtained from λ by projecting each state and distribution, and removing all moves with actions not in the alphabet of $\mathcal{M}^i$, together with the subsequent states. Given a strategy σ of $\mathcal{M}$, its projection $\sigma{\upharpoonright}_{\mathcal{M}^i}$ onto $\mathcal{M}^i$ is such that, for any finite path λ^i of $\mathcal{M}^i$ and transition $\mathsf{last}(\lambda^i) \xrightarrow{a} \mu^i$,

$$\sigma{\upharpoonright}_{\mathcal{M}^i}(\lambda^i)(a, \mu^i) \stackrel{\text{def}}{=} \sum_{\lambda{\upharpoonright}_{\mathcal{M}^i} = \lambda^i} \ \sum_{\mu{\upharpoonright}_{\mathcal{M}^i} = \mu^i} \mathbb{P}^\sigma_{\mathcal{M}}(\lambda) \cdot \sigma(\lambda)(a, \mu) / \mathbb{P}^{\sigma{\upharpoonright}_{\mathcal{M}^i}}_{\mathcal{M}^i}(\lambda^i)$$

From Lemma 7.2.6 in [52], for any trace w over actions $\mathcal{A} \subseteq \mathcal{A}^i$ we have $\mathbb{P}^{\sigma}_{\mathcal{M}}(w) = \mathbb{P}^{\sigma\upharpoonright_{\mathcal{M}^i}}_{\mathcal{M}^i}(w)$. Therefore, if φ is defined on traces of $\mathcal{A} \subseteq \mathcal{A}^i$, we have that $\mathcal{M}^\sigma \models \varphi \Leftrightarrow \varphi(\mathbb{P}^{\sigma}_{\mathcal{M}}) \Leftrightarrow \varphi(\mathbb{P}^{\sigma^i\upharpoonright_{\mathcal{M}^i}}_{\mathcal{M}^i}) \Leftrightarrow (\mathcal{M}^i)^{\sigma^i\upharpoonright_{\mathcal{M}^i}} \models \varphi$.

Take any fair strategy σ of $\mathcal{M}$. From Lemma 2 in [36], the projections $\sigma\upharpoonright_{\mathcal{M}^2}$ and $\sigma\upharpoonright_{\mathcal{M}^1}$ are fair. For the (Conj) rule, we have that $\mathcal{M}^i \models^{\mathsf{u}} \varphi^{G_i}$ implies $(\mathcal{M}^i)^{\sigma\upharpoonright_{\mathcal{M}^i}} \models \varphi^{G_i}$, since $\sigma\upharpoonright_{\mathcal{M}^i}$ is fair; this in turn implies $\mathcal{M}^\sigma \models \varphi^{G_i}$, since $\mathcal{A}_{G^i} \subseteq \mathcal{A}^i$. Since σ was an arbitrary fair strategy of $\mathcal{M}$, this implies $\mathcal{M} \models^{\mathsf{u}} \varphi^{G_1} \wedge \varphi^{G_2}$. For the (Asym) rule, applying the (Conj) yields $\mathcal{M} \models^{\mathsf{u}} (\varphi^{A_1} \to \varphi^{G}) \wedge \varphi^{A_2} \wedge (\varphi^{A_2} \to \varphi^{A_1})$, which reduces to $\mathcal{M} \models^{\mathsf{u}} \varphi^{G}$. □

5.4.2. Under-approximating Pareto sets

We now describe how to pick the targets of the specifications φ^i in a compositional rule, such as from Theorem 15, so that φ in the conclusion of the rule is achievable. To this end, we compositionally compute an under-approximation of the Pareto set for φ; we illustrate this approach in an example in Section 5.5 below.

Consider N reward structures, $r_1, \ldots, r_N$, and objectives φ^i, $i \in I$, over these reward structures for respective games G^i, as well as an objective φ, over the same reward structures, for the composed game $\mathcal{G} = \|_{i\in I} G^i$. Note that, for each $1 \leq j \leq N$, the reward structure r_j may be present in several objectives φ^i. Let P^i be an under-approximation of the Pareto set for $G^i \models \varphi^i$, for $i \in I$, and so each point $\vec{v}^{(i)} \in P^i$ represents a target vector for the MQ $\varphi^i[\vec{v}^{(i)}]$ achievable in the game G^i.

For a set P^i, define the *lifting* $\uparrow P^i$ to all N reward structures by $\uparrow P^i \stackrel{\text{def}}{=} \{\vec{v} \in \mathbb{R}^N \mid$ the coordinates of $\vec{v}$ appearing in φ^i are in $P^i\}$. The set $P' \stackrel{\text{def}}{=} \cap_{i\in I} \uparrow P^i$ is the set of target vectors for all N reward structures, which are consistent with achievability of all objectives φ^i in the respective games. The projection $\downarrow P'$ of P' onto the space of reward structures appearing in φ then yields an under-approximation of the Pareto set P for φ in the composed game $\mathcal{G}$, that is, $\downarrow P' \subseteq P$. A vector $\vec{v} \in \downarrow P'$ can be achieved by instantiating the objectives φ^i with any targets $\vec{v}^{(i)}$ in P' that match $\vec{v}$.

5.5. The Compositional Strategy Synthesis Method

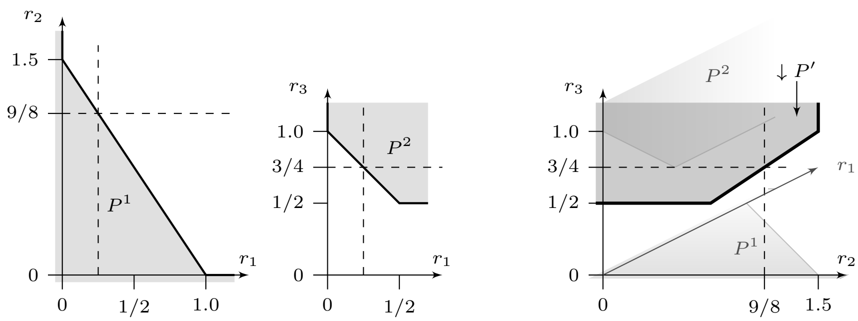
Our method for compositional strategy synthesis, based on monolithic synthesis for individual component games, is summarised as follows:

(S1) **User Input:** A composed game $\mathcal{G} = \|_{i\in I} \mathcal{G}^i$, MQs φ^i, φ, and matching PA rules for use in Theorem 15.

(S2) **First Stage:** Obtain under-approximations of Pareto sets P^i for $\mathcal{G}^i \models \varphi^i$, and compute the compositional under-approximated Pareto set $\downarrow P'$.

(S3) **User Feedback:** Pick targets $\vec{v}$ for the global specification φ from $\downarrow P'$; matching targets $\vec{v}^{(i)}$ for φ^i can be picked automatically from P^i.

(S4) **Second Stage:** Synthesise strategies π^i for $\mathcal{G}^i \models \varphi^i[\vec{v}^{(i)}]$.



$\mathcal{G}^1 \models \mathsf{Eratio}(-r_1/c)(-v_1) \rightarrow \mathsf{Eratio}(r_2/c)(v_2)$ $\quad \mathcal{G}^2 \models \mathsf{Eratio}(-r_1/c)(-v_1) \wedge \mathsf{Eratio}(-r_3/c)(-v_3)$ $\quad \mathcal{G} \models \mathsf{Eratio}(r_2/c)(v_2) \wedge \mathsf{Eratio}(-r_3/c)(-v_3)$

Figure 13: Pareto sets for example games in Figure 7. Specifications are given beneath the respective sets. The rightmost figure shows the compositional Pareto set, $\downarrow P'$, as well as the oblique projections of P^1 and P^2 for reference.

(S5) **Output:** The strategy $\|_{i\in I}\, \pi^i$, winning for $\mathcal{G} \models \varphi[\vec{v}]$ by Theorem 15.

Steps (S1), (S4) and (S5) are sufficient if the targets are known, while (S2) and (S3) are an additional feature enabled by the Pareto set computation.

Example 9. *Consider again the (controllable multichain) games in Figure 7. We want to find a strategy for the composition $\mathcal{G} \stackrel{\text{def}}{=} \mathcal{G}^1 \parallel \mathcal{G}^2$ (right) by compositional synthesis for the components $\mathcal{G}^1$ (left) and $\mathcal{G}^2$ (centre). We give an example of steps (S1) to (S3). We consider reward structures r_1, r_2, r_3 and c, and the MQs $\varphi^1 = \mathsf{Eratio}(-r_1/c)(-v_1) \rightarrow \mathsf{Eratio}(r_2/c)(v_2)$ for $\mathcal{G}^1$, and $\varphi^2 = \mathsf{Eratio}(-r_1/c)(-v_1) \wedge \mathsf{Eratio}(-r_3/c)(-v_3)$ for $\mathcal{G}^3$. The global specification is $\varphi = \mathsf{Eratio}(r_2/c)(v_2) \wedge \mathsf{Eratio}(-r_3/c)(-v_3)$ for $\mathcal{G}$. This constitutes the inputs in step (S1).*

For step (S2), under-approximations of the Pareto sets for $\mathcal{G}^1$ and $\mathcal{G}^2$ are shown in Figure 13 (left) and (centre) respectively, together with the compositionally obtained under-approximated Pareto set $\downarrow P'$ for $\mathcal{G}$ (right). In step (S3), if we want, for example, to find a strategy satisfying φ with $(v_2, v_3) = (\frac{9}{8}, \frac{3}{4})$, we look up a value for v_1 that is consistent with both P^1 and P^2, as indicated by the dashed lines in Figure 13 (left) and (centre), and we find $v_1 = \frac{1}{4}$ to be consistent for both components. In step 4) we synthesise strategies for the MQs $\varphi^1[(\frac{1}{4}, \frac{9}{8})]$ for $\mathcal{G}^1$ and $\varphi^2[(\frac{1}{4}, \frac{3}{4})]$ for $\mathcal{G}^2$. In $\mathcal{G}^1$ the strategy π^1 always plays d (as explained in Example 7), and the strategy π^2 for $\mathcal{G}^2$ is illustrated in Figure 10. Finally, we return the composed strategy $\pi = \pi^1 \parallel \pi^2$ in step (S5).

6. Conclusion

We presented a compositional framework for strategy synthesis in stochastic games, where winning conditions are specified as multi-dimensional long-run objectives. The algorithm proposed in Theorem 7 constructs succinct ε-optimal stochastic memory update strategies, and we show how such winning strategies for component games can be composed to be winning for the composed game. Since building the composed game is not necessary in order to synthesise a strategy to control it, our approach enhances scalability. However, this is at a cost of restricting the class of strategies. The techniques have been implemented and applied to several case studies, as reported separately in [38, 5, 4, 60].

Our compositional framework applies to all specifications defined on traces, which include almost-sure and expected ratio rewards treated here, as well as expected total rewards studied in [19, 20], but not mean payoffs. Nevertheless, the ability to synthesise strategies for mean payoffs at the component level is useful because, as we showed in Proposition 2 and Theorem 11, these enable us to synthesise strategies for conjunctions of almost-sure ratio rewards and Boolean combinations of expected ratio rewards that are well suited to the compositional approach. We anticipate that our framework is sufficiently general to permit further specifications defined on traces, such as Büchi specifications or ratio rewards with arbitrary satisfaction thresholds, but the problem of synthesising winning strategies for such specifications for individual components remains open.

As future work, we intend to investigate satisfaction objectives with arbitrary probability thresholds, and believe that this is possible using ideas from [32]. We would also like to adopt a unifying view between expectation and satisfaction objectives as done for MDPs in [14]. Finally, the compositional framework could be augmented by automatically decomposing games and specifications given a rule schema.

Acknowledgements. The authors thank Vojtěch Forejt, Stefan Kiefer, Benoît Barbot and Dave Parker for helpful discussions. The authors are partially supported by ERC Advanced Grant VERIWARE and EPSRC Mobile Autonomy Programme Grant.

References

[1] C. Baier, C. Dubslaff, S. Klüppelholz, and L. Leuschner. Energy-utility analysis for resilient systems using probabilistic model checking. In *Petri Nets*, pages 20–39. Springer, 2014.

[2] C. Baier, M. Größer, and F. Ciesinski. Quantitative analysis under fairness constraints. In *ATVA*, volume 5799 of *LNCS*, pages 135–150. Springer, 2009.

[3] C. Baier, M. Größer, M. Leucker, B. Bollig, and F. Ciesinski. Controller synthesis for probabilistic systems (extended abstract). In *IFIP TCS*, volume 155, pages 493–506. Springer, 2004.

[4] N. Basset, M. Kwiatkowska, U. Topcu, and C. Wiltsche. Strategy synthesis for stochastic games with multiple long-run objectives. In *TACAS*, volume 9035 of *LNCS*, pages 256–271. Springer, 2015.

[5] N. Basset, M. Kwiatkowska, and C. Wiltsche. Compositional controller synthesis for stochastic games. In *CONCUR*, volume 8704 of *LNCS*, pages 173–187. Springer, 2014.

[6] T. Brázdil, V. Brožek, K. Chatterjee, V. Forejt, and A. Kučera. Two views on multiple mean-payoff objectives in Markov decision processes. *LMCS*, 10(4), 2014.

[7] T. Brázdil, V. Brožek, V. Forejt, and A. Kučera. Stochastic games with branching-time winning objectives. In *LICS*, pages 349–358. ACM/IEEE, 2006.

[8] T. Brázdil, P. Jančar, and A. Kučera. Reachability games on extended vector addition systems with states. In *ICALP*, volume 6199 of *LNCS*, pages 478–489. Springer, 2010.

[9] R. Brenguier and J.F. Raskin. Optimal values of multidimensional mean-payoff games. Research report, Université Libre de Bruxelles (U.L.B.), Belgium, 2014.

[10] V. Bruyère, E. Filiot, M. Randour, and J.F. Raskin. Meet your expectations with guarantees: Beyond worst-case synthesis in quantitative games. *arXiv preprint arXiv:1309.5439*, 2013.

[11] K. Chatterjee, L. Doyen, T.A. Henzinger, and J.F. Raskin. Generalized mean-payoff and energy games. In *FSTTCS*, volume 8 of *LIPIcs*, pages 505–516. Schloss Dagstuhl, 2010.

[12] K. Chatterjee and M. Henzinger. Faster and dynamic algorithms for maximal end-component decomposition and related graph problems in probabilistic verification. In *SODA*, pages 1318–1336. ACM-SIAM, 2011.

[13] K. Chatterjee and T.A. Henzinger. Assume-guarantee synthesis. In *TACAS*, volume 4424 of *LNCS*, pages 261–275. Springer, 2007.

[14] K. Chatterjee, Z. Komárková, and J. Křetínský. Unifying two views on multiple mean-payoff objectives in Markov decision processes. In *LICS*, pages 244–256. ACM/IEEE, 2015.

[15] K. Chatterjee, R. Majumdar, and T.A. Henzinger. Markov decision processes with multiple objectives. In *STACS*, volume 3884 of *LNCS*, pages 325–336. Springer, 2006.

[16] K. Chatterjee, M. Randour, and J.F. Raskin. Strategy synthesis for multi-dimensional quantitative objectives. *Acta Informatica*, 51(3–4):129–163, 2014.

[17] T. Chen, V. Forejt, M. Kwiatkowska, D. Parker, and A. Simaitis. PRISM-games: A model checker for stochastic multi-player games. In *TACAS*, volume 7795 of *LNCS*, pages 185–191. Springer, 2013.

[18] T. Chen, V. Forejt, M. Kwiatkowska, A. Simaitis, A. Trivedi, and M. Ummels. Playing stochastic games precisely. In *CONCUR*, volume 7454 of *LNCS*, pages 348–363, 2012.

[19] T. Chen, V. Forejt, M. Kwiatkowska, A. Simaitis, and C. Wiltsche. On stochastic games with multiple objectives. In *MFCS*, volume 8087 of *LNCS*, pages 266–277. Springer, 2013.

[20] T. Chen, M. Kwiatkowska, A. Simaitis, and C. Wiltsche. Synthesis for multi-objective stochastic games: An application to autonomous urban driving. In *QEST*, volume 8054 of *LNCS*, pages 322–337. Springer, 2013.

[21] L. Cheung, N. Lynch, R. Segala, and F. Vaandrager. Switched PIOA: Parallel composition via distributed scheduling. *TCS*, 365(1–2):83–108, 2006.

[22] B.A. Davey and H.A. Priestley. *Introduction to lattices and order*. Cambridge University Press, 1990.

[23] L. De Alfaro. *Formal verification of probabilistic systems*. PhD thesis, Stanford University, 1997.

[24] L. de Alfaro and T.A. Henzinger. Interface automata. *SIGSOFT Software Engineering Notes*, 26(5):109–120, 2001.

[25] L. de Alfaro, T.A. Henzinger, and R. Jhala. Compositional methods for probabilistic systems. In *CONCUR*, volume 2154 of *LNCS*, pages 351–365. Springer, 2001.

[26] A. Ehrenfeucht and J. Mycielski. Positional strategies for mean payoff games. *International Journal of Game Theory*, 8(2):109–113, 1979.

[27] K. Etessami, M. Kwiatkowska, M.Y. Vardi, and M. Yannakakis. Multi-objective model checking of Markov decision processes. *LMCS*, 4(8):1–21, 2008.

[28] J. Filar and K. Vrieze. *Competitive Markov decision processes*. Springer, 1996.

[29] V. Forejt, M. Kwiatkowska, and D. Parker. Pareto curves for probabilistic model checking. In *ATVA*, volume 7561 of *LNCS*, pages 317–332. Springer, 2012.

[30] M. Gelderie. Strategy composition in compositional games. In *ICALP*, volume 7966 of *LNCS*, pages 263–274. Springer, 2013.

[31] S. Ghosh, R. Ramanujam, and S. Simon. Playing extensive form games in parallel. In *CLIMA*, volume 6245 of *LNCS*, pages 153–170. Springer, 2010.

[32] H. Gimbert and F. Horn. Solving simple stochastic tail games. In *SODA*, pages 847–862. ACM-SIAM, 2010.

[33] H. Gimbert and E. Kelmendi. Two-player perfect-information shift-invariant submixing stochastic games are half-positional. *arXiv preprint arXiv:1401.6575*, 2014.

[34] F. Horn. *Random Games.* PhD thesis, Université Denis Diderot - Paris 7 & Rheinisch-Westfälische Technische Hochschule Aachen, 2008.

[35] G. Katz, D. Peled, and S. Schewe. Synthesis of distributed control through knowledge accumulation. In *CAV*, volume 6806 of *LNCS*, pages 510–525. Springer, 2011.

[36] M. Kwiatkowska, G. Norman, D. Parker, and H. Qu. Compositional probabilistic verification through multi-objective model checking. *Information and Computation*, 232:38–65, 2013.

[37] M. Kwiatkowska and D. Parker. Automated verification and strategy synthesis for probabilistic systems. In *ATVA*, volume 8172 of *LNCS*, pages 5–22. Springer, 2013.

[38] M. Kwiatkowska, D. Parker, and C. Wiltsche. PRISM-games 2.0: A tool for multi-objective strategy synthesis for stochastic games. In *TACAS*, 2016. (submitted).

[39] Marta Kwiatkowska and Maria Svorenova. Quantitative verification and strategy synthesis for stochastic games. In *ECC*, 2016. To appear.

[40] D.A. Levin, Y. Peres, and E.L. Wilmer. *Markov chains and mixing times.* AMS, 2009.

[41] L. MacDermed and C.L. Isbell. Solving stochastic games. In *NIPS*, pages 1186–1194. Curran Associates, Inc., 2009.

[42] P. Madhusudan and P.S. Thiagarajan. A decidable class of asynchronous distributed controllers. In *CONCUR*, volume 7454 of *LNCS*, pages 145–160. Springer, 2002.

[43] S. Mohalik and I. Walukiewicz. Distributed games. In *FSTTCS*, volume 2914 of *LNCS*, pages 338–351. Springer, 2003.

[44] A. Pneuli and R. Rosner. Distributed reactive systems are hard to synthesize. In *FOCS*, pages 746–757. IEEE, 1990.

[45] M.L. Puterman. *Markov decision processes: discrete stochastic dynamic programming.* Wiley-Interscience, 2009.

[46] Michael O. Rabin. Probabilistic automata. *Information and Control*, 6(3):230 – 245, 1963.

[47] M. Randour, J.F. Raskin, and O. Sankur. Variations on the stochastic shortest path problem. In *VMCAI*, volume 8318 of *LNCS*, pages 1–18. Springer, 2014.

[48] M. Randour, J.F. Raskin, and O. Sankur. Percentile queries in multi-dimensional Markov decision processes. In *CAV*, volume 9206 of *LNCS*, pages 123–139. Springer, 2015.

[49] R.T. Rockafellar. *Convex Analysis.* Princeton University Press, 1997.

[50] S.M. Ross. *Stochastic processes*, volume 2. John Wiley & Sons New York, 1996.

[51] O. Sankur. *Robustness in Timed Automata: Analysis, Synthesis, Implementation.* Thèse de doctorat, LSV, ENS Cachan, France, 2013.

[52] R. Segala. *Modelling and Verification of Randomized Distributed Real Time Systems.* PhD thesis, Massachusetts Institute of Technology, 1995.

[53] Lloyd S Shapley. Stochastic games. *PNAS*, 39(10):1095, 1953.

[54] N. Shimkin and A. Shwartz. Guaranteed performance regions in Markovian systems with competing decision makers. *Automatic Control*, 38(1):84–95, 1993.

[55] A. Simaitis. *Automatic Verification of Competitive Stochastic Systems.* PhD thesis, University of Oxford, 2013.

[56] A. Sokolova and E.P. de Vink. Probabilistic automata: system types, parallel composition and comparison. In *VOSS*, volume 2925 of *LNCS*, pages 1–43. Springer, 2004.

[57] Y. Velner. Finite-memory strategy synthesis for robust multidimensional mean-payoff objectives. In *LICS*, pages 79:1–79:10. ACM/IEEE, 2014.

[58] C. von Essen and B. Jobstmann. Synthesizing efficient controllers. In *VMCAI*, volume 7148 of *LNCS*, pages 428–444. Springer, 2012.

[59] D.J. White. Multi-objective infinite-horizon discounted Markov decision processes. *J. Math. Anal. Appl.*, 89(2):639–647, 1982.

[60] C. Wiltsche. *Assume-Guarantee Strategy Synthesis for Stochastic Games.* PhD thesis, University of Oxford, 2016. (forthcoming).

Appendix A. Proofs of results of Section 2

Appendix A.1. Proof of Proposition 1

Proof. The belief $\mathfrak{d}^{\Diamond}_{\lambda}$ after seeing a path λ is defined inductively as follows: $\mathfrak{d}^{\Diamond}_{s} = \pi_{\mathsf{d}}(s)$, $\mathfrak{d}_{\lambda s'} \stackrel{\text{def}}{=} \pi_{\mathsf{u}}(\mathfrak{d}^{\Diamond}_{\lambda}, s')$. We define $\mathfrak{d}^{\Box}_{\lambda}$ for $\mathsf{Player}\ \Box$ similarly. We first remark that, given a game $\mathcal{G}$ and two strategies π, σ, then $\mathbb{P}^{\pi,\sigma}_{\mathcal{G}}$ enjoys the following recursive definition $\mathbb{P}^{\pi,\sigma}_{\mathcal{G}}(s) = \varsigma(s)$,

$$\mathbb{P}^{\pi,\sigma}_{\mathcal{G}}(\lambda s') = \mathbb{P}^{\pi,\sigma}_{\mathcal{G}}(\lambda) \sum_{\mathfrak{m},\mathfrak{n}} \mathfrak{d}^{\Diamond}_{\lambda}(\mathfrak{m}) \mathfrak{d}^{\Box}_{\lambda}(\mathfrak{n}) \sum_{\mathfrak{m}',\mathfrak{n}'} \Delta((s,\mathfrak{m},\mathfrak{n}),(s',\mathfrak{m}',\mathfrak{n}')). \tag{A.1}$$

Indeed, $\mathfrak{d}^{\Diamond}_{\lambda}(\mathfrak{m})\mathfrak{d}^{\Box}_{\lambda}(\mathfrak{n})$ is the probability of having memory element $\mathfrak{m}$ and $\mathfrak{n}$ knowing the path λ and $\sum_{\mathfrak{m}',\mathfrak{n}'} \Delta((s,\mathfrak{m},\mathfrak{n}),(s',\mathfrak{m}',\mathfrak{n}'))$ is the probability to have state s' knowing that λ ends in s with memory element $\mathfrak{m}$ and $\mathfrak{n}$.

For the deterministic update strategy $\bar{\pi}$, (A.1) can be written as:

$$\mathbb{P}^{\bar{\pi},\sigma}_{\mathcal{G}}(\lambda s') = \mathbb{P}^{\pi,\sigma}_{\mathcal{G}}(\lambda) \sum_{\mathfrak{n}} \mathfrak{d}^{\Box}_{\lambda}(\mathfrak{n}) \sum_{\mathfrak{n}'} \Delta((s,\mathfrak{d}^{\Diamond}_{\lambda},\mathfrak{n}),(s',\mathfrak{d}^{\Diamond}_{\lambda s'},\mathfrak{n}')). \tag{A.2}$$

To show that (A.1) and (A.2) yield the same inductive definition, it suffices to note that the base case is satisfied since $\mathbb{P}^{\bar{\pi},\sigma}_{\mathcal{G}}(s) = \varsigma(s) = \mathbb{P}^{\pi,\sigma}_{\mathcal{G}}(s)$, and show that

$$\Delta((s,\mathfrak{d}^{\Diamond}_{\lambda},\mathfrak{n}),(s',\mathfrak{d}^{\Diamond}_{\lambda s'},\mathfrak{n}')) = \sum_{\mathfrak{m},\mathfrak{m}'} \mathfrak{d}^{\Diamond}_{\lambda}(\mathfrak{m})\Delta((s,\mathfrak{m},\mathfrak{n}),(s',\mathfrak{m}',\mathfrak{n}')). \tag{A.3}$$

The right-hand side of (A.3) is equal to

$$\sum_{\mathfrak{m}} \mathfrak{d}^{\Diamond}_{\lambda}(\mathfrak{m}) \sum_{\mathfrak{m}'} \pi_{\mathsf{u}}(\mathfrak{m},s')(\mathfrak{m}') \cdot \sigma_{\mathsf{u}}(\mathfrak{n},s')(\mathfrak{n}') \cdot \begin{cases} \pi_{\mathsf{c}}(s,\mathfrak{m})(s') & \text{if } s \in S_{\Diamond} \\ \sigma_{\mathsf{c}}(s,\mathfrak{n})(s') & \text{if } s \in S_{\Box} \\ \Delta(s,s') & \text{if } s \in S_{\bigcirc} \end{cases} \tag{A.4}$$

which after simplification using $\sum_{\mathfrak{m}'} \pi_{\mathsf{u}}(\mathfrak{m},s')(\mathfrak{m}') = 1$ and $\pi_{\mathsf{c}}(s,\mathfrak{d}^{\Diamond}_{\lambda})(s') = \sum_{\mathfrak{m}\in\mathfrak{M}} \mathfrak{d}^{\Diamond}_{\lambda}(\mathfrak{m})\pi_{\mathsf{c}}(s,\mathfrak{m})(s')$ yields $\sigma_{\mathsf{u}}(\mathfrak{n},s')(\mathfrak{n}') \cdot \begin{cases} \pi_{\mathsf{c}}(s,\mathfrak{d}^{\Diamond}_{\lambda})(s') & \text{if } s \in S_{\Diamond} \\ \sigma_{\mathsf{c}}(s,\mathfrak{n})(s') & \text{if } s \in S_{\Box} \\ \Delta(s,s') & \text{if } s \in S_{\bigcirc} \end{cases}$ which is equal to the left-hand side of (A.3): $\Delta((s,\mathfrak{d}^{\Diamond}_{\lambda},\mathfrak{n}),(s',\mathfrak{d}^{\Diamond}_{\lambda s'},\mathfrak{n}'))$. We have proved that $\mathbb{P}^{\pi,\sigma}_{\mathcal{G}}$ and $\mathbb{P}^{\bar{\pi},\sigma}_{\mathcal{G}}$ satisfy the same inductive definition, thus they are equal. □

Appendix A.2. Some properties of long-run behaviour

We state here several results about the (multi-objective) long-run behaviours of stochastic models as introduced in Section 2.3.

We begin by recalling standard definitions for Markov chains. A *bottom strongly connected component* (BSCC) of a DTMC $\mathcal{D}$ is a nonempty maximal subset of states $\mathcal{B} \subseteq S$ s.t. every state in $\mathcal{B}$ is reachable from any other state in $\mathcal{B}$, and no state outside $\mathcal{B}$ is reachable. A state $s \in S$ of a DTMC $\mathcal{D}$ is called *recurrent* if it is in some BSCC $\mathcal{B}$ of $\mathcal{D}$. A state which is not recurrent is called *transient*. A DTMC is *irreducible* if its state space comprises a single BSCC. Given a BSCC $\mathcal{B} \subseteq S$ of a DTMC $\mathcal{D}$, the *stationary distribution* $\mu_{\mathcal{B}} \in \mathsf{D}(S)$ is such that $\sum_{s\in\mathcal{B}} \mu_{\mathcal{B}}(s) \cdot \Delta(s,t) = \mu_{\mathcal{B}}(t)$ holds for all $t \in \mathcal{B}$; its existence and uniqueness is demonstrated, e.g., by Proposition M.2 in [28].

Theorem 17 (Theorem 4.16 in [40])**.** *Let $\mathcal{D}$ be an irreducible DTMC with a single BSCC $\mathcal{B}$, and let r be a reward structure. The sequence $\frac{1}{N+1}\mathit{rew}^N(r)(\lambda)$ almost surely converges to $\sum_{s\in\mathcal{B}} \mu_{\mathcal{B}}(s) \cdot r(s)$, where $\lambda \in \Omega_{\mathcal{D}}$.*

Remark 4. *From the previous theorem, the mean-payoff in a BSCC $\mathcal{B}$ is the same at every state $s \in \mathcal{B}$, and we define, for a reward structure $\vec{r}$, $\mathsf{mp}(\vec{r})(\mathcal{B}) \stackrel{\text{def}}{=} \sum_{s \in \mathcal{B}} \vec{r}(s)\mu_{\mathcal{B}}(s)$.*

Lemma 15. *Given a finite DTMC $\mathcal{D}$ and a reward structure $\vec{r}$, for $\lambda \in \Omega_{\mathcal{D}}$ the limit $\lim_{N\to\infty} \frac{1}{N+1}\mathsf{rew}^N(\vec{r})(\lambda)$ almost surely exists and takes values $\vec{x}$ in the finite set $\{\mathsf{mp}(\vec{r})(\mathcal{B}) \mid \mathcal{B}$ is a BSCC of $\mathcal{D}\}$ with probability*

$$\sum_{\mathcal{B} \ s.t.\ \mathsf{mp}(\vec{r})(\mathcal{B})=\vec{x}} \mathbb{P}_{\mathcal{D}}(\mathsf{F}\,\mathcal{B}).$$

Proof. Note first that, for every path $\lambda \in \Omega_{\mathcal{D}}$, $\frac{1}{N+1}\mathsf{rew}^N(\vec{r})(\lambda)$ converges if and only if, for every suffix λ' of λ, $\frac{1}{N+1}\mathsf{rew}^N(\vec{r})(\lambda')$ converges to the same limit. For every recurrent state t of $\mathcal{D}$, we denote by W_t the set of paths λ such that t is the first recurrent state along λ. Paths $\lambda \in W_t$ have suffixes λ' distributed according to $\mathbb{P}_{\mathcal{D},t}$. By Theorem 17, $\frac{1}{N+1}\mathsf{rew}^N(\vec{r})(\lambda')$ almost surely converges to $\sum_{t'\in\mathcal{B}} \mu_{\mathcal{B}}(t')r(t')$. Thus, with probability $\mathbb{P}_{\mathcal{D}}(\mathsf{F}\,\mathcal{B}) = \sum_{t\in\mathcal{B}} P_{\mathcal{D}}(W_t)$, the sequence $\frac{1}{N+1}\mathsf{rew}^N(\vec{r})(\lambda)$ converges to $\mathsf{mp}(\vec{r})(\mathcal{B})$. To conclude, it suffices to recall that $\sum_{\mathcal{B}\in\mathfrak{B}(\mathcal{D})} P_{\mathcal{D}}(\mathsf{F}\,\mathcal{B}) = 1$, and thus the result holds almost surely. □

Remark 5. *Consequently, $\mathsf{mp}(\vec{r})(\lambda) \geq 0$ for almost all paths of a DTMC $\mathcal{D}$ if and only if $\mathsf{mp}(\vec{r})(\mathcal{B}) \geq 0$ for every BSCC $\mathcal{B}$ of $\mathcal{D}$ that is reached.*

Lemma 16. *Given a finite DTMC and two reward structures r and c with c weakly positive, then the sequence $\mathsf{rew}^N(r)/(1+\mathsf{rew}^N(c))$ converges almost surely to $\mathsf{mp}(r)/\mathsf{mp}(c)$.*

Proof. Fix a finite DTMC $\mathcal{D}$. By Lemma 15, the limit inferior can be replaced by the true limit in $\mathsf{mp}(c)$ and $\mathsf{mp}(r)$. $\mathsf{ratio}(r/c)(\lambda) = \frac{\mathsf{mp}(r)(\lambda)}{\mathsf{mp}(c)(\lambda)}$. Using the conditions on c imposed by the definition of ratio rewards, we have that, with probability one, $\mathsf{mp}(c) > 0$. Hence,

$$\frac{\mathsf{mp}(r)(\lambda)}{\mathsf{mp}(c)(\lambda)} = \frac{\lim_{N\to\infty} \frac{1}{N+1}\mathsf{rew}^N(r)(\lambda)}{\lim_{N\to\infty} \frac{1}{N+1}\mathsf{rew}^N(c)(\lambda)} = \lim_{N\to\infty} \frac{\frac{1}{N+1}\mathsf{rew}^N(r)(\lambda)}{\frac{1}{N+1}\mathsf{rew}^N(c)(\lambda)}.$$

There is no indeterminacy for this quotient of limits, as the denominator is positive and the numerator is finite. Simplifying the $\frac{1}{N+1}$ term yields the equality $\frac{\mathsf{mp}(r)(\lambda)}{\mathsf{mp}(c)(\lambda)} = \lim_{N\to\infty} \frac{\mathsf{rew}^N(r)(\lambda)}{\mathsf{rew}^N(c)(\lambda)}$. This is almost surely equal to $\mathsf{ratio}(r/c)(\lambda) = \lim_{N\to\infty} \frac{\mathsf{rew}^N(r)(\lambda)}{1+\mathsf{rew}^N(c)(\lambda)}$ since $\mathsf{rew}^N(c)(\lambda) \to +\infty$ almost surely. □

As a consequence of Lemma 15 and Lemma 16 it follows that mean-payoff and ratio rewards are linear in finite DTMCs.

Proposition 13. *Given a finite DTMC, let $\vec{r}$ be an n-dimensional reward structure and c a weakly positive reward structure. For every $\vec{x} \in \mathbb{R}^n_{\geq 0}$, it almost surely holds that $\mathsf{mp}(\vec{x}\cdot\vec{r}) = \vec{x}\cdot\mathsf{mp}(\vec{r})$ and $\mathsf{ratio}(\vec{x}\cdot\vec{r}/c) = \vec{x}\cdot\mathsf{ratio}(\vec{r}/c)$.*

Appendix A.3. Proof of Proposition 2

Proof. First note that $\mathsf{Pratio}(\vec{r}/\vec{c})(\vec{v})$ holds iff $\mathsf{Pratio}((\vec{r} - \vec{v} \bullet \vec{c})/\vec{c})(0)$ holds. So, up to replacing $\vec{r} - \vec{v} \cdot \vec{c}$ by $\vec{r}$, we can assume without loss of generality that $\vec{v} = 0$. We now show equivalence between $\mathsf{Pmp}(\vec{r})(0)$ and $\mathsf{Pratio}(\vec{r}/\vec{c})(0)$. Fix a Player $\Box$ strategy σ and a dimension i. By weak positivity of c_i, for almost every path the sequence $(1 + \mathsf{rew}^N(c_i)(\lambda))/(N+1)$ has a positive limit inferior and, as it takes only positive values, this implies that it has a positive lower bound. It is also upper-bounded as $(1 + \mathsf{rew}^N(c_i)(\lambda))/(N+1) \leq 1 + N \max_{s \in S} c_i(s)/(N+1) \to 1 + \max_{s \in S} c_i(s)$. Now, note that for two real-valued sequences a_N and b_N such that $\underline{\lim}\ a_N \geq 0$, b_N is positive, $\inf b_N > 0$ and $\sup b_N < \infty$ then $\underline{\lim}\ a_N/b_N \geq 0$. We apply this remark with sequences $a_N(\lambda) = \mathsf{rew}^N(r_i)(\lambda)/(N+1)$ and $b_N(\lambda) = (1 + \mathsf{rew}^N(c_i)(\lambda))/(N+1)$, where λ is a path such that $\mathsf{mp}(c_i)(\lambda) > 0$. Then for almost every path the following equivalence holds: $\mathsf{ratio}(r_i/c_i)(\lambda) = \underline{\lim} a_N(\lambda)/b_N(\lambda) \geq 0$ iff $\mathsf{mp}(r_i)(\lambda) \geq 0$. Thus, π is winning for $\mathsf{Pratio}(\vec{r}/\vec{c})(0)$ iff it is winning for $\mathsf{Pmp}(\vec{r} - \vec{v} \bullet \vec{c})(0)$. $\Box$

Appendix B. Proofs of results of Section 3

Appendix B.1. Proof of Theorem 6

We first state a technical lemma used in the proof of Theorem 6.

Lemma 17. *Let $(X_n)_{\geq n}$ be a sequence of real-valued random variables. If $\mathbb{P}(\underline{\lim}_{n\to\infty} X_n \geq v) = 1$, then, for every $\delta > 0$, $\mathbb{P}(X_n < v - \delta) \to 0$ as $n \to \infty$.*

Proof. Assume that $\mathbb{P}(\underline{\lim}_{n\to\infty} X_n \geq v) = 1$, and fix $\delta > 0$. Let $A_n \stackrel{\text{def}}{=} \bigcup_{m \geq n} \{e \mid X_m(e) < v - \delta\}$. As A_n is a non-increasing sequence of events, as $n \to \infty$, it holds that $\mathbb{P}(A_n) \to \mathbb{P}(\bigcap_{n\geq 1} A_n)$, which is zero by hypothesis of the lemma. Hence $\mathbb{P}(X_n < v - \delta)$ also tends to zero as $n \to \infty$, since $\mathbb{P}(X_n < v - \delta) \leq P(A_n)$. $\Box$

We can now proceed to the proof of Theorem 6.

Proof. Let π be a Player $\Diamond$ strategy achieving $\mathsf{Pmp}(\vec{r})$. We show that, for every $\varepsilon > 0$, Player $\Diamond$ has a finite DU strategy to achieve $\mathsf{Pmp}(\vec{r} + \varepsilon)$. Let $\mathcal{M} = \mathcal{G}^\pi$.

Denote by $S_\mathcal{M}$ and $S_\mathcal{G}$ the respective states spaces of $\mathcal{M}$ and $\mathcal{G}$. Without loss of generality, we assume that the memory of π is the set $\Omega^{\mathsf{fin}}_\mathcal{G}$ of paths in $\mathcal{G}$, and so $\mathcal{M}$ is an infinite tree where each state is identified uniquely by a path $\lambda \in \Omega^{\mathsf{fin}}_\mathcal{G}$. Consider the set $S_{\mathcal{M},\mathcal{G}} \stackrel{\text{def}}{=} \{\mathsf{last}(\lambda) \in S_\mathcal{G} \mid \lambda \in S_\mathcal{M}\}$ of states of the game that appear in some state of $\mathcal{M}$. For every $\lambda \in S_\mathcal{M}$, $\mathbb{P}^\sigma_{\mathcal{M},\lambda}(\mathsf{mp}(\vec{r}) \geq 0) = 1$ holds for all Player $\Box$ strategies σ. Consider, for each state $s \in S_{\mathcal{M},\mathcal{G}}$, a path $\lambda_s \in S_\mathcal{M}$ with $\mathsf{last}(\lambda_s) = s$, which uniquely identifies a state in $\mathcal{M}$ (note that, given s, λ_s is not unique, but it suffices to pick an arbitrary one). Then, for every Player $\Box$ strategy σ and every state s, it holds that $\mathbb{P}^\sigma_{\mathcal{M},\lambda_s}(\underline{\lim}_{N\to\infty} \frac{1}{N+1}\mathsf{rew}^N(\vec{r}) \geq 0) = 1$, and hence, by Lemma 17, the quantity $p_{s,h,\sigma} \stackrel{\text{def}}{=} \mathbb{P}^\sigma_{\mathcal{M},\lambda_s}(\frac{1}{h+1}\mathsf{rew}^h(\vec{r}) \leq -\varepsilon/2)$ (defined for a fixed $h > 0$) tends to 0 as $\varepsilon \to 0$. We define $p_{h,\sigma} \stackrel{\text{def}}{=} \max_s p_{s,h,\sigma}$, and let p_h be the maximum $p_{s,h,\sigma}$ over all MD $\mathsf{Player}\ \Box$ strategies σ. As the maxima are taken over finite sets, we have that $p_h \to 0$ as $h \to \infty$.

Now for a fixed positive integer h, construct the finite DU $\mathsf{Player}\ \Diamond$ strategy π_h that plays as follows: starting from $s \in S_{\mathcal{M},\mathcal{G}}$, it initialises its memory to λ_s and plays π for h steps; then, from whatever state $t \in S_{\mathcal{M},\mathcal{G}}$ it arrived at, it resets its memory to λ_t and plays π for a further h steps, and so on. Fix any MD strategy of $\mathsf{Player}\ \Box$, and a BSCC $\mathcal{B}$ of the finite induced DTMC $\mathcal{D} = \mathcal{G}^{\pi_h,\sigma}$. Given a state $s \in S_{\mathcal{M},\mathcal{G}}$, let $\tilde{s} = (s, \lambda_s, \mathfrak{n})$ be the corresponding state of $\mathcal{D}$ (where $\mathfrak{n}$ is the only memory element of σ). Note that by definition of π_h, a state of the form $\tilde{s}$ with $s \in S_{\mathcal{M},\mathcal{G}}$ is seen every h steps. In particular $\mathcal{B}$ must contain at least one state $\tilde{s}_0$ with $s_0 \in S_{\mathcal{M},\mathcal{G}}$. By Remark 4, we then have $\mathbb{P}_{\mathcal{D}}(\mathsf{mp}(\vec{r}) = \mathsf{mp}(\vec{r})(\mathcal{B})) = 1$, so it suffices to find a lower-bound for $\mathsf{mp}(\vec{r})(\mathcal{B})$, which is equivalent to find a lower-bound for $\lim_{N\to\infty} \frac{1}{N+1}\mathbb{E}_{\mathcal{D},\tilde{s}_0}[\mathsf{rew}^N(\vec{r})]$. We have constructed π_h so that every h steps a state in $S_{\mathcal{M},\mathcal{G}}$ is encountered, and hence it holds, for every $k \geq 0$, that $\mathbb{E}_{\mathcal{D},\tilde{s}_0}[\mathsf{rew}^{kh}(\vec{r})] \geq k \cdot \min_{s\in S_{\mathcal{M},\mathcal{G}}} \mathbb{E}_{\mathcal{D},\tilde{s}}[\mathsf{rew}^h(\vec{r})]$. From a state $s \in S_{\mathcal{M},\mathcal{G}}$, with probability less than $p_{h,\sigma}$, the reward accumulated is at least $-h\rho^*$, where $\rho^* = \max_{s\in S_{\mathcal{G}},i} |r_i(s)|$. Further, with probability greater than $1 - p_{h,\sigma}$ the reward accumulated is at least $-h\varepsilon/2$. Therefore, for every state $s \in S_{\mathcal{M},\mathcal{G}}$, $\mathbb{E}_{\mathcal{D},\tilde{s}}[\mathsf{rew}^h(\vec{r})] \geq -p_{h,\sigma}\rho^* - (1 - p_{h,\sigma})h\varepsilon/2 \geq -p_h\rho^* - h\varepsilon/2$. Hence, $\mathbb{E}_{\mathcal{D},\tilde{s}}[\mathsf{rew}^{kh}(\vec{r})] \geq -kh(p_h\rho^* + \varepsilon/2)$. Dividing by $kh+1$ and letting k go towards infinity, we get that $\mathbb{E}_{\mathcal{D},\tilde{s}_0}[\mathsf{mp}(\vec{r})] = \lim_k \frac{1}{kh+1}\mathbb{E}_{\mathcal{D},\tilde{s}_0}[\mathsf{rew}^{kh}(\vec{r})] \geq -p_h\rho^* - \varepsilon/2$. We therefore have, for every BSCC $\mathcal{B}$ of $\mathcal{D}$, that $\mathsf{mp}(\vec{r})(\mathcal{B}) \geq -p_h\rho^* - \varepsilon/2$, and hence, by Remark 5, $\mathsf{Player}\ \Diamond$ achieves $\mathsf{Pmp}(\vec{r} + p_h\rho^* + \varepsilon/2)$ against all MD $\mathsf{Player}\ \Box$ strategies. Then, by Theorem 3, $\mathsf{Player}\ \Diamond$ achieves $\mathsf{Pmp}(\vec{r} + p_h\rho^* + \varepsilon/2)$ against all $\mathsf{Player}\ \Box$ strategies. Since $p_h \to 0$, we can find h large enough so that $p_h\rho^* \leq \varepsilon/2$, and hence have $\mathsf{Pmp}(\vec{r} + \varepsilon)$ against every σ. □

Appendix B.2. Proof of Proposition 6

Proof. The proof method is based on similar results in [16, 55]. Consider the game $\mathcal{G}$ in Figure B.14 with objective $\mathsf{Pmp}(\vec{r})(0)$. From s_0, when $\mathsf{Player}\ \Box$ chooses a sequence w of actions with $|w| \leq n+1$, the total rewards are shifted by the vector $-(\alpha_w, 2^{|w|} - 1 - \alpha_w)$, where $\alpha_w \stackrel{\text{def}}{=} \sum_{j=1}^{|w|} \delta_{w_j = a} 2^{j-1}$ is the number corresponding to the binary word w represented with the least significant bit first, with a coding for 1 and b for 0.

Exponential memory DU strategy. We show that there is a winning DU strategy π for $\mathsf{Player}\ \Diamond$ with exponential memory $\mathfrak{M} \stackrel{\text{def}}{=} \bigcup_{k=1}^{n+1}\{a,b\}^k$, which at state s_{n+1} plays the distribution ν_w defined by $\nu_w(a) \stackrel{\text{def}}{=} \frac{\alpha_w}{2^{n+1}-1}$ and $\nu_w(b) \stackrel{\text{def}}{=} 1 - \nu_w(a)$, where $w \in \mathfrak{M}$ is the current memory, determining α_w. This strategy compensates the shift incurred while going through the $\mathsf{Player}\ \Box$ states, and hence, for every loop, the expected total reward is $(0,0)$. Thus also the expected mean-payoff is $(0,0)$. We now show that the almost sure mean-payoff is $(0,0)$. As the strategy π has finite memory, the induced PA $\mathcal{G}^\pi$ is finite, and it suffices to consider MD strategies for $\mathsf{Player}\ \Box$ in $\mathcal{G}^\pi$, cf. Lemma 1. Let R_i be the random variable equals to the total reward of the ith loop. The random variables $(R_i)_{i\geq 0}$ are independent identically distributed and of expectation zero, and we

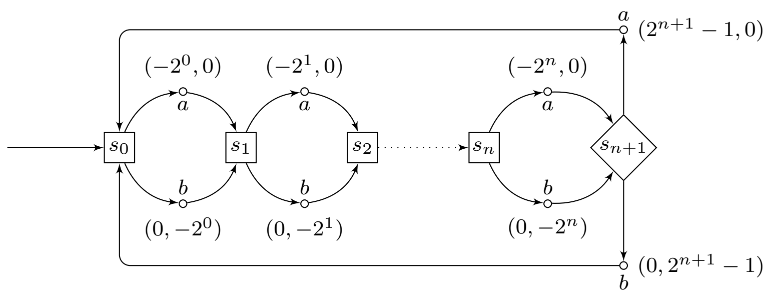


Figure B.14: Finite-memory SU strategies are exponentially more succinct than finite-memory DU strategies for $\mathsf{Pmp}(\vec{r})$.

apply the strong law of large numbers to obtain that $(1/N)\sum_{i=0}^{N} R_i$ converges almost surely towards the common mean 0. Hence, π is winning for almost sure convergence.

Linear memory SU strategy. We now show how the distribution ν_w can be simulated by an SU strategy π that contains only $2(n+1)$ memory elements. Let $\mathfrak{M} \stackrel{\text{def}}{=} \bigcup_{i=0}^{n+1}\{a_i, b_i\}$, and let $\pi_{\mathsf{c}}(s_{n+1}, l_{n+1}) \stackrel{\text{def}}{=} l$ for $l \in \{a, b\}$, that is, l_i is the memory at state s_i corresponding intuitively to the intention of Player $\Diamond$ to play the action l.

We denote by $\mathbb{P}(l_i|w)$ the probability of Player $\Diamond$ being in memory l_i after having read the sequence w of length i, starting from s_0. We now inductively define a memory update function such that, for $i \le n+1$ and $w \in \{a,b\}^i$, $\mathbb{P}(a_i|w) = \frac{\alpha_w}{2^i-1}$ (and $\mathbb{P}(b_i|w) = 1 - \mathbb{P}(a_i|w)$), so that, in particular, when $i = n+1$, Player $\Diamond$ chooses the next move according to the distribution ν_w. In the base case (when $i = 0$ and w is the empty word), $\mathbb{P}(a_0|w) = 1$ necessitates that the initial memory as well as the memory when returning after each loop to s_0 is $\pi_{\mathsf{d}}(s_0) \stackrel{\text{def}}{=} \pi_{\mathsf{u}}(l_{n+1}, l') \stackrel{\text{def}}{=} a_0$.

When going from s_i to s_{i+1} via an action q, the memory $l_i \in \{a_i, b_i\}$ in s_i is updated to l'_{i+1} in s_{i+1}, under the condition

$$\mathbb{P}(l'_{i+1}|wq) = \mathbb{P}(a_i|w)\cdot\pi_{\mathsf{u}}(a_i, q)(l'_{i+1}) + \mathbb{P}(a_i|w)\cdot\pi_{\mathsf{u}}(b_i, q)(l'_{i+1}). \tag{B.1}$$

Taking $l' = a$ and taking q to be a or b in (B.1) gives as necessary conditions

$$\mathbb{P}(a_{i+1}|wa) = \frac{\alpha_w + 2^i}{2^{i+1}-1} = \frac{\alpha_w}{2^i-1}\cdot\pi_{\mathsf{u}}(a_i, a)(a_{i+1}) + \left(1 - \frac{\alpha_w}{2^i-1}\right)\cdot\pi_{\mathsf{u}}(b_i, a)(a_{i+1}) \tag{B.2}$$

$$\mathbb{P}(a_{i+1}|wb) = \frac{\alpha_w}{2^{i+1}-1} = \frac{\alpha_w}{2^i-1}\cdot\pi_{\mathsf{u}}(a_i, a)(a_{i+1}) + \left(1 - \frac{\alpha_w}{2^i-1}\right)\cdot\pi_{\mathsf{u}}(b_i, b)(a_{i+1}); \tag{B.3}$$

Taking $l' = b$ in (B.1) gives symmetric conditions.

We now define the memory update function according to these conditions. Define $\pi_{\mathsf{u}}(a_i, a) \stackrel{\text{def}}{=} a_{i+1}$ and $\pi_{\mathsf{u}}(b_i, b) \stackrel{\text{def}}{=} b_{i+1}$, following the intuition that there is no need to change the intention to play a or b, corresponding to the current memory a_i and b_i, respectively, when the intention is followed. Further, using the conditions in (B.2), we obtain, for $l, \bar{l} \in \{a, b\}$ with $\bar{l} \neq l$ that $\pi_{\mathsf{u}}(l_i, \bar{l})(l_{i+1}) \stackrel{\text{def}}{=} \frac{2^i-1}{2^{i+1}-1}$ and $\pi_{\mathsf{u}}(l_i, \bar{l})(\bar{l}_{i+1}) \stackrel{\text{def}}{=} \frac{2^i}{2^{i+1}-1}$. We have thus defined π so that at s_{n+1} the choices it made were according to ν_w. Then, as shown above, this strategy is winning. Moreover, π contains $2(n+1)$ memory elements, and is therefore exponentially smaller than the DU strategy described above. Note that this strategy could have been described with only two memory elements a and b but the strategy would still need a linear space to encode the memory updates (as distinct game transitions lead to distinct updating rules).

No sub-exponential DU strategy. We show that every finite DU strategy achieving $\mathsf{Pmp}(\vec{r})$ requires at least exponential memory. Consider a finite DU strategy π with less than $2^{n+1} - 1$ memory elements. We show that it loses against some finite strategy σ. For every memory element $\mathfrak{m} \in \mathfrak{M}$, there exist at least two distinct sequences $w^1_{\mathfrak{m}}$ and $w^2_{\mathfrak{m}}$ such that the memory updated from $\mathfrak{m}$ is the same after seeing either $w^1_{\mathfrak{m}}$ or $w^2_{\mathfrak{m}}$, denoted $f(\mathfrak{m})$, and such that $\mathsf{rew}(w^1_{\mathfrak{m}}) \geq \mathsf{rew}(w^2_{\mathfrak{m}}) + 1$ for r_1. Consider the finite memory strategy σ^1 (resp. σ^2) that simulates the deterministic memory of π and plays the actions in $w^1_{\mathfrak{m}}$ (resp. $w^2_{\mathfrak{m}}$) from s_0 and memory $\mathfrak{m}$. The strategy π reacts to $f(\mathfrak{m})$ at state s_{n+1}, so the rewards associated to $w^1_{\mathfrak{m}}$ or $w^2_{\mathfrak{m}}$ are not compensated. Let $\mathcal{D}_i \stackrel{\text{def}}{=} \mathcal{G}^{\pi,\sigma^i}$. We extend the reasoning to k loops as follows. For pairwise associated sequences $w^i = w^i_{\mathfrak{m}_1} l_1 w^i_{\mathfrak{m}_2} l_2 \cdots w^i_{\mathfrak{m}_k} l_k$ with $i \in \{1, 2\}$, it holds that $\mathsf{rew}(r_1)(w^1) \geq \mathsf{rew}(r_1)(w^2) + k$ and $\mathbb{P}_{\mathcal{D}_1}(w^1) = \mathbb{P}_{\mathcal{D}_2}(w^2)$. Hence, the average rewards in the two DTMCs are separated by $1/L$, where L is the length of a loop. Hence, if π wins against σ^1, then $\mathbb{P}_{\mathcal{D}_1}(\mathsf{mp}(r_1) = 0) = \mathbb{P}_{\mathcal{D}_1}(\mathsf{mp}(r_2) = 0) = 1$, and hence, $\mathbb{P}_{\mathcal{D}_2}(\mathsf{mp}(r_1) \leq -1/L) = 1$. The strategy π loses against σ^1 or σ^2, which concludes the proof. □

Appendix B.3. Proof of Proposition 5

The proof uses notations on matrix and vectors that we introduce now. We recall that we use boldface notation for vectors over the state space; in particular, given a scalar a, we write $\mathbf{a}$ for the vector with a in each component. With this notation a one-dimensional reward structure r is represented by the vector $\mathbf{r}$ whose sth component is $r(s)$. We use the notation $[\mathbf{v}]_s$ to refer to the sth component v_s of a vector $\mathbf{v}$. and use the notation $[A]_{s,t}$ to refer to the sth row and tth column of a matrix A. We use the *induced matrix norm* of A defined by $\|A\|_\infty \stackrel{\text{def}}{=} \max_{1 \leq i \leq m} \sum_{j=1}^{n} |A_{ij}|$. This norm is sub-multiplicative, i.e. $\|AB\|_\infty \leq \|A\|_\infty \|B\|_\infty$. Given a vector $\mathbf{v}$ with entries indexed by the state space S, we denote by $\mathbf{v}_E$ the vector with entries indexed by the subset $E \subseteq S$, such that $[\mathbf{v}_E]_s = v_s$ for all $s \in E$. Similarly, given a matrix A with entries indexed by a set S, we denote by $A_{E,E'}$ the $|E| \times |E'|$ submatrix of A with entries indexed by $E, E' \subseteq S$, such that $[A_{E,E'}]_{s,t} = A_{s,t}$ for $(s,t) \in E \times E'$.

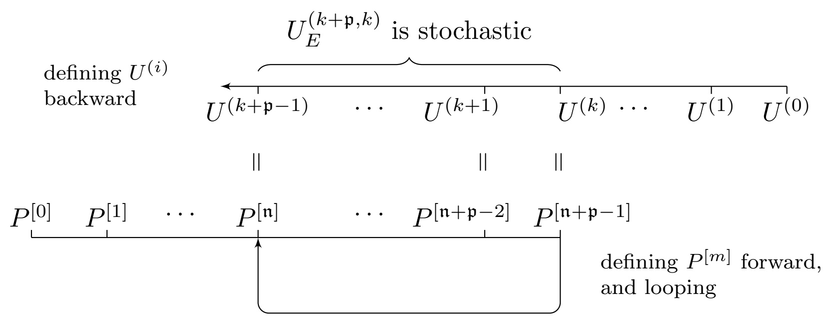


Figure B.15: Matrices $U^{(i)}$ and $P^{[m]}$ to define the ultimately periodic matrix based strategy for $\mathsf{Player}\ \Box$ to spoil $\mathsf{EE}(r-\varepsilon)$ in the proof of Proposition 5.

For a state $s \in S$ and $E \subseteq S$, we write $A_{s,E}$ instead of $A_{\{s\},E}$ and A_E instead of $A_{E,E}$. We denote by I_S the $|S| \times |S|$ identity matrix with entries indexed by S. A square matrix A with nonnegative real entries is *(right) stochastic* if $\sum_t A_{s,t} = 1$ for all rows s of A.

Proof. The proof is as follows. For k large enough and for states in S_∞, there is no cut-off used to define $\mathbf{u}^k$, and hence $\mathbf{u}^k$ satisfies the same linear equations as the expected non-truncated energy $\mathbf{e}^k$, which we proceed to express in terms of matrices. We then construct a finite memory $\mathsf{Player}\ \Box$ strategy from the sequence $\mathbf{u}^k$ and the associated matrices, so that the expected energy with respect to the reward r is bounded. By operating with the reward $r - \varepsilon$, we substract $-\varepsilon$ at each step, and so the expected energy goes to $-\infty$, falsifying $\mathsf{EE}(r-\varepsilon)$.

Let k_0 be the least integer such that, for all $k \geq k_0$, $u_s^k < 0$ for every $s \in S_\infty$. For $k \geq 0$ and $s \in S_\Box$, let $\sigma_k(s)$ be a successor of s, for which the minimum is attained, that is, $u_s^{k+1} = \min\{0, r(s) + u^k_{\sigma_k(s)}\}$. Let $U^{(k)}$ be the $S \times S$ matrix for $\mathcal{M}$ defined by

$$U^{(k)}_{s,t} = \begin{cases} 1 & \text{if } s \in S_\Box \wedge t = \sigma_k(s) \\ \Delta(s,t) & \text{if } s \in S_\bigcirc \\ 0 & \text{otherwise,} \end{cases}$$

for all $s, t \in S$. Let $U^{(j,i)}$ be the matrix product $U^{(j-1)} \cdot U^{(j-2)} \cdot \ldots \cdot U^{(i)}$ for $j > i$, and let $U^{(i,i)} = I$ (the identity). We use the following block decomposition of the matrix $U^{(k)}$

$$U^{(k)} = \begin{pmatrix} U^{(k)}_{S_\infty} & U^{(k)}_{S_\infty, S_{\mathsf{fin}}} \\ 0 & U^{(k)}_{S_{\mathsf{fin}}} \end{pmatrix}. \tag{B.4}$$

The zero block in the lower left corner of $U^{(k)}$ arises because all successors of states in S_{fin} are in S_{fin}. In particular, $U^{(j,i)}_{S_\infty} = U^{(j-1)}_{S_\infty} \cdot U^{(j-2)}_{S_\infty} \cdot \ldots \cdot U^{(i)}_{S_\infty}$.

Remark 6. *For every $k \geq k_0$, it holds that $\mathbf{u}^{k+1}_{S_\infty} = \mathbf{r}_{S_\infty} + [U^{(k)} \cdot \mathbf{u}^k]_{S_\infty}$.*

We now proceed to show Proposition 5 as a consequence of Lemmas 18–23.

Lemma 18. *For every $l \geq 0$, there exists a constant $b_l \geq 0$, such that, for every $k \geq k_0$, it holds that $\|\mathbf{u}_{S_\infty}^{k+l}\|_\infty \leq \|U_{S_\infty}^{(k+l,k)}\|_\infty \cdot \|\mathbf{u}_{S_\infty}^{k}\|_\infty + b_l$.*

Proof. We show the following more general statement by induction:

$$\mathbf{u}_{S_\infty}^{k+l} \geq U_{S_\infty}^{(k+l,k)} \cdot \mathbf{u}_{S_\infty}^{k} + \mathbf{a} - l\boldsymbol{\rho}^*, \tag{B.5}$$

where $\mathbf{a}$ and $\boldsymbol{\rho}^*$ are the constant vector with equal components $a \stackrel{\text{def}}{=} \min_{s\in S_{\mathsf{fin}}} u_s^*$ and $\rho^* \stackrel{\text{def}}{=} \max_{s\in S} |r(s)|$, respectively.

The base case, for $l = 0$, is satisfied. Now assume that the result is true for some index l, and we show that it implies that it is true for $l+1$. Recall that for $k \geq k_0$ and $s \in S_\infty$, there is no cut-off of positive values in u_s^k. We thus obtain

$$\begin{aligned}
\mathbf{u}_{S_\infty}^{k+l+1} &= \mathbf{r}_{S_\infty} + [U^{(k+l)} \cdot \mathbf{u}^{k+l}]_{S_\infty} && \text{(Remark 6)}\\
&= \mathbf{r}_{S_\infty} + U_{S_\infty}^{(k+l)} \cdot \mathbf{u}_{S_\infty}^{k+l} + U_{S_\infty,S_{\mathsf{fin}}}^{(k+l)} \cdot \mathbf{u}_{S_{\mathsf{fin}}}^{k+l} && \text{(by (B.4))}\\
&\geq -\boldsymbol{\rho}^* + U_{S_\infty}^{(k+l)} \cdot \mathbf{u}_{S_\infty}^{k+l} + U_{S_\infty,S_{\mathsf{fin}}}^{(k+l)} \cdot \mathbf{a} && \text{(definition of } \mathbf{a} \text{ and } \boldsymbol{\rho}^*)\\
&\geq -\boldsymbol{\rho}^* + U_{S_\infty}^{(k+l)} \cdot (U_{S_\infty}^{(k+l,k)} \cdot \mathbf{u}_{S_\infty}^{k} + \mathbf{a} - l\boldsymbol{\rho}^*) + U_{S_\infty,S_{\mathsf{fin}}}^{(k+l)} \cdot \mathbf{a} \\
& && \text{(induction hypothesis)}\\
&\geq U_{S_\infty}^{(k+l+1,k)} \cdot \mathbf{u}_{S_\infty}^{k} + (U_{S_\infty}^{(k+l)} + U_{S_\infty,S_{\mathsf{fin}}}^{(k+l)}) \cdot \mathbf{a} - (l+1)\boldsymbol{\rho}^* && \text{(rearranging)}\\
&\geq U_{S_\infty}^{(k+l+1,k)} \cdot \mathbf{u}_{S_\infty}^{k} + \mathbf{a} - (l+1)\boldsymbol{\rho}^*. && (U^{(k+l)} \text{ is stochastic})
\end{aligned}$$

It now suffices to define $b_l \stackrel{\text{def}}{=} a - l\rho^*$, and take the norm in (B.5):

$$\begin{aligned}
\|\mathbf{u}_{S_\infty}^{k+l}\|_\infty &= \max_{s\in S_\infty} (-\mathbf{u}_s^{k+l}) && \text{(norm)}\\
&\leq \max_{s\in S_\infty} (U_{S_\infty}^{(k+l,k)} \cdot (-\mathbf{u}_{S_\infty}^{k}) + b_l) && \text{(by (B.5))}\\
&= \|U_{S_\infty}^{(k+l,k)} \cdot (-\mathbf{u}_{S_\infty}^{k})\|_\infty + b_l \\
&\leq \|U_{S_\infty}^{(k+l,k)}\|_\infty \cdot \|\mathbf{u}_{S_\infty}^{k}\|_\infty + b_l. && \text{(sub-multiplicativity)}
\end{aligned}$$ □

Lemma 19. *Let $b \geq 0$, and let $(x_m)_{m\in\mathbb{N}}$ and $(c_m)_{m\in\mathbb{N}}$ be non-negative real sequences. If $x_m \to \infty$ as $m \to \infty$, and, for every $m \geq 0$, $x_{m+1} \leq c_m x_m + b$ and $c_m \leq 1$, then it holds that $\sup_{m\geq 0} c_m = 1$.*

Proof. Assume toward a contradiction that there exists $\theta < 1$ such that, for every m, $c_m \leq \theta$. As $x_m \to \infty$, there exists m_0 such that, for every $m \geq m_0$, it holds that $x_m > b/(1-\theta)$, and hence that $x_{m+1}/x_m \leq c_m + b/x_m < \theta + b/(b/(1-\theta)) = 1$. This yields that, from the index m_0, the sequence $(x_m)_{m\geq m_0}$ is decreasing, and thus cannot go to $+\infty$, a contradiction. □

Lemma 20. *If $S_\infty \neq \emptyset$, then there exists a set $E \subseteq S_\infty$ and indices $j > i \geq k_0$ such that $U_E^{(i,j)}$ is stochastic.*

Proof. Given a subset of states $A \subseteq S$, and a $S \times S$ stochastic matrix P, we define $\texttt{Reach}(A, P) \stackrel{\text{def}}{=} \{s' \mid \exists s \in A \,.\, P_{s,s'} > 0\}$. Note that P_E is stochastic if and only if $\texttt{Reach}(E, P) \subseteq E$, and further that $\texttt{Reach}(\texttt{Reach}(A, P), P') = \texttt{Reach}(A, P \cdot P')$. Let $l = 2^{|S|}$, $s \in S$ and $k \in \mathbb{N}$. Consider the sets $\texttt{Reach}(\{s\}, U^{(k+l,k+l)})$, $\texttt{Reach}(\{s\}, U^{(k+l,k+l-1)})$, …, $\texttt{Reach}(\{s\}, U^{(k+l,k)})$. By the pigeonhole principle, there are at least two indices i, j with $k \leq i < j \leq k + l$ such that $\texttt{Reach}(\{s\}, U^{(k+l,j)}) = \texttt{Reach}(\{s\}, U^{(k+l,i)})$, and we denote this common set by $E_{s,k}$. We thus have that

$$
\begin{aligned}
\texttt{Reach}(E_{s,k}, U^{(j,i)}) &= \texttt{Reach}(\texttt{Reach}(\{s\}, U^{(k+l,j)}), U^{(j,i)}) \\
&= \texttt{Reach}(\{s\}, U^{(k+l,j)} \cdot U^{(j,i)}) \\
&= \texttt{Reach}(\{s\}, U^{(k+l,i)}) \\
&= E_{s,k}.
\end{aligned}
$$

Hence $U^{(j,i)}_{E_{s,k}}$ is stochastic. It now suffices to prove that $E_{s,k} \subseteq S_\infty$ for some $s \in S_\infty$ and $k \geq k_0$. Assume for the sake of contradiction that $E_{s,k} \cap S_{\mathsf{fin}} \neq \emptyset$ for every $s \in S_\infty$ and $k \geq k_0$. By definition of $E_{s,k}$ there exists i such that $E_{s,k} = \texttt{Reach}(\{s\}, U^{(k+l,i)})$ and hence such that $U^{(k+l,i)}_{s,S_{\mathsf{fin}}} \neq 0$. Now recall that $U^{(i,k)}_{S_{\mathsf{fin}}}$ is stochastic, since every successor of a state in S_{fin} is in S_{fin}. We deduce that $U^{(k+l,k)}_{s,S_{\mathsf{fin}}} = U^{(k+l,i)}_{s,S_{\mathsf{fin}}} \cdot U^{(i,k)}_{S_{\mathsf{fin}}} \neq 0$. The matrix $U^{(k+l,k)}$ is the product of l matrices, each of which has entries either zero or greater than $p_{\min}$, the minimal probability on edges of the PA $\mathcal{M}$. Therefore, coefficients of $U^{(k+l,k)}$ are either zero or greater than $p^l_{\min}$, and so $\|U^{(k+l,k)}_{s,S_{\mathsf{fin}}}\|_\infty \geq p^l_{\min}$. Since $U^{(k+l,k)}$ is stochastic, its row-sum are equal to one, that is, $\sum_{s' \in S_{\mathsf{fin}}} U^{(k+l,k)}_{s,s'} + \sum_{s' \in S_\infty} U^{(k+l,k)}_{s,s'} = 1$, for every $s \in S$ and $k \geq 0$. This implies that $\sum_{s' \in S_\infty} U^{(k+l,k)}_{s,s'} \leq 1 - p^l_{\min}$, for every $s \in S$ and $k \geq 0$. We let $c_m \stackrel{\text{def}}{=} \|U^{(k_0+lm+l,k_0+lm)}_{S_\infty}\|_\infty$, and have by the above discussion that $\sup_m c_m \leq 1 - p^l_{\min} < 1$, to which we now derive a contradiction. Let $x_m \stackrel{\text{def}}{=} \|\mathbf{u}^{k_0+lm}_{S_\infty}\|_\infty$, for which we have, by Lemma 18, that $x_{m+1} \leq c_m \cdot x_m + b_l$. We now use Lemma 19 to obtain $\sup_m c_m = 1$, a contradiction. □

We now define Player □ strategies and the expected energies they induce in terms of matrices. We consider *ultimately periodic* sequences of matrices that after a finite prefix $\mathfrak{n}$ keep repeating the same $\mathfrak{p}$ elements in a loop. Formally, an ultimately periodic sequence $(P^{[m]})_{m \in \mathbb{N}}$ with *prefix* $\mathfrak{n}$ and *period* $\mathfrak{p}$ is such that the mth element is equal to the element of index $m \mod (\mathfrak{n}, \mathfrak{p})$ (that is, $P^{[m]} = P^{[m \mod (\mathfrak{n},\mathfrak{p})]}$), where

$$
m \mod (\mathfrak{n}, \mathfrak{p}) \stackrel{\text{def}}{=} \begin{cases} m & \text{if } m \leq \mathfrak{n} + \mathfrak{p} - 1 \\ \mathfrak{n} + (m - \mathfrak{n} \mod \mathfrak{p}) & \text{otherwise.} \end{cases}
$$

A stochastic matrix P *conforms* to $\mathcal{M}$ if, for every $s \in S_{\bigcirc}$ and all $s' \in \Delta(s)$, it holds that $P_{s,s'} = \Delta(s, s')$. We define a finite strategy by an ultimately periodic sequence of matrices $(P^{[k]})_{k\in\mathbb{N}}$ that conform to $\mathcal{M}$: the memory is a counter $m \leq \mathfrak{n}+\mathfrak{p}$ that is updated at every step from m to $m+1 \mod (\mathfrak{n},\mathfrak{p})$; and in state s and memory $\mathfrak{m}$ the choice function selects s' with probability $P^{[\mathfrak{m}]}_{s,s'}$. To express several steps of the strategy, we introduce the interval matrices $P^{[m,m+l]} = P^{[m]}\cdots P^{[m+l-1]}$ with $P^{[m,m]} = I_S$, and the corresponding cumulative matrices $\hat{P}^{[m,m+l]} = \sum_{q=0}^{l-1} P^{[m,m+q]}$ with $\hat{P}^{[m,m]} = 0$.

For every step $k \geq 0$ and memory $\mathfrak{m}$, we define a vector $\mathbf{e}^k_{(\mathfrak{m})}(r)$, where the entry for s is defined as $e^k_{s,\mathfrak{m}}$ in the PA with reward structure r, that is, the expected energy for r after k steps at state $(s,\mathfrak{m})$ of the induced DTMC.

Lemma 21. *Given a strategy based on an ultimately periodic matrix with prefix* $\mathfrak{n}$ *and period* $\mathfrak{p}$*, it holds that* $\mathbf{e}^l_{(m \mod (\mathfrak{n},\mathfrak{p}))}(r) = \hat{P}^{[m,m+l]} \cdot \mathbf{r}$*, for all* $l \geq 0$ *and* $m \geq 0$*.*

Proof. We show this statement by induction on l. The base case for $l = 0$ is satisfied. Now assume the statement holds for l, and we show for $l+1$. As the strategy with memory $m \mod (\mathfrak{n},\mathfrak{p})$ plays according to the matrix $P^{[m]}$, and increments its memory to $m+1 \mod (\mathfrak{n},\mathfrak{p})$, it holds that

$$\begin{aligned}\mathbf{e}^{l+1}_{(m \mod (\mathfrak{n},\mathfrak{p}))}(r) &= \mathbf{r} + P^{[m]} \cdot \mathbf{e}^l_{(m+1 \mod (\mathfrak{n},\mathfrak{p}))}(r)\\ &= P^{[m]} \cdot \hat{P}^{[m+1,m+l+1]} \cdot \mathbf{r}\\ &= \hat{P}^{[m,m+l+1]} \cdot \mathbf{r}.\end{aligned}$$ □

We now show that the strategy based on ultimately periodic matrices is able to decrease the expected energy in the periodic phase by a nonzero amount every $\mathfrak{p}$ number of steps.

Lemma 22. *Given a strategy based on an ultimately periodic matrix with prefix* $\mathfrak{n}$ *and period* $\mathfrak{p}$*, and a set* E *such that* $A = P_E^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]}$ *is stochastic, then, for all* $j \geq 0$*, it holds that* $[e^{j\mathfrak{p}}_{(\mathfrak{n})}(r-\varepsilon)]_E = \sum_{k=0}^{j-1} A^k \cdot [\hat{P}^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]} \cdot \mathbf{r}]_E - j\mathfrak{p}\varepsilon$*.*

Proof. Note first that $P^{[\mathfrak{n},\mathfrak{n}+j\mathfrak{p}]} = (P^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]})^j$, that $\hat{P}^{[\mathfrak{n},\mathfrak{n}+j\mathfrak{p}]} \cdot \mathbf{1} = j\mathfrak{p}$, and that $\hat{P}^{[\mathfrak{n},\mathfrak{n}+j\mathfrak{p}]} = \sum_{k=0}^{j-1}(P^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]})^k \cdot \hat{P}^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]}$. Since the restriction of $P^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]}$ to the set E is stochastic, it holds, for every vector $\mathbf{x}$, that $[P^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]} \cdot \mathbf{x}]_E = P_E^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]} \cdot \mathbf{x}_E$. We apply Lemma 21 with $l = j\mathfrak{p}$ and $m = \mathfrak{n}$, and thus get, for all $j \geq 0$, that

$$\begin{aligned}[\mathbf{e}^{j\mathfrak{p}}_{(\mathfrak{n})}(r-\varepsilon)]_E &= [\hat{P}^{[\mathfrak{n},\mathfrak{n}+j\mathfrak{p}]} \cdot (\mathbf{r}-\boldsymbol{\varepsilon})]_E\\ &= \left[\sum_{k=0}^{j-1}(P^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]})^k \cdot \hat{P}^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]} \cdot \mathbf{r} - j\mathfrak{p}\boldsymbol{\varepsilon}\right]_E\\ &= \sum_{k=0}^{j-1}(P_E^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]})^k \cdot [\hat{P}^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]} \cdot \mathbf{r}]_E - j\mathfrak{p}\boldsymbol{\varepsilon}.\end{aligned}$$ □

We now describe a situation where the cut-off of positive values in the definition of $\mathbf{u}^k$ does not occur.

Lemma 23. *For $k \geq k_0$, and $E \subseteq S_\infty$ such that $U_E^{(k+\mathfrak{p},k)}$ is stochastic,*

$$\mathbf{u}_E^{k+\mathfrak{p}} = [\hat{U}^{(k+\mathfrak{p},k)} \cdot \mathbf{r}]_E + U_E^{(k+\mathfrak{p},k)} \cdot \mathbf{u}_E^k. \tag{B.6}$$

Proof. We show, by induction on l, the following more general statement: for all $l \geq 0$, $k \geq k_0$, and $E, E' \subseteq S_\infty$ such that $E' = \mathtt{Reach}(E, U^{(k+l,k)})$, it holds that

$$\mathbf{u}_E^{k+l} = [\hat{U}^{(k+l,k)} \cdot \mathbf{r}]_E + U_{E,E'}^{(k+l,k)} \cdot \mathbf{u}_{E'}^k.$$

The base case for $l = 0$ is straightforward. Now suppose that the result holds for l, and we show it for $l+1$. Let $k \geq k_0$ and $E, E' \subseteq S_\infty$ such that $E' = \mathtt{Reach}(E, U^{(k+l+1,k)})$, and let $E'' = \mathtt{Reach}(E, U^{(k+l+1,k+1)})$. Note that $\mathtt{Reach}(E'', U^{(k)}) = E' \subseteq S_\infty$, and hence that $E'' \subseteq S_\infty$, since every predecessor of a state in S_∞ is in S_∞. As $k+1 \geq k_0$ and $E'' \subseteq S_\infty$, it holds that $\mathbf{u}_{E''}^{k+1} = \mathbf{r}_{E''} + U_{E'',E'}^{(k+1)} \mathbf{u}_{E'}^k$, and hence we can conclude the proof by

$$\begin{aligned}
\mathbf{u}_E^{k+l+1} &= [\hat{U}^{(k+l+1,k+1)} \cdot \mathbf{r}]_E + U_{E,E''}^{(k+l+1,k+1)} \cdot \mathbf{u}_{E''}^{k+1} \\
&= [\hat{U}^{(k+l+1,k+1)} \cdot \mathbf{r}]_E + U_{E,E''}^{(k+l+1,k+1)} \cdot \mathbf{r}_{E''} + U_{E,E''}^{(k+l+1,k+1)} \cdot U_{E'',E'}^{(k+1)} \cdot \mathbf{u}_{E'}^k \\
&= [\hat{U}^{(k+l+1,k)} \cdot \mathbf{r}]_E + U_{E,E'}^{(k+l+1,k)} \cdot \mathbf{u}_{E'}^k,
\end{aligned}$$

where the first equality is due to the induction hypothesis. □

We can now complete the proof of Proposition 5. We assume that $S_\infty \neq \emptyset$. By Lemma 20, there exists a set $E \subseteq S_\infty$, and indices $k_0 \leq k < k + \mathfrak{p}$, such that $\mathtt{Reach}(E, U^{(k+\mathfrak{p},k)}) = E$. By Lemma 23, it holds that $\mathbf{u}_E^{k+\mathfrak{p}} = \mathbf{y} + A \cdot \mathbf{u}_E^k$ with $\mathbf{y} = [\hat{U}^{(k+\mathfrak{p},k)} \cdot \mathbf{r}]_E$ and $A = U_E^{(k+\mathfrak{p},k)}$.

We define $\mathsf{Player}\ \Box$ strategy σ based on ultimately periodic matrices $U^{(k+\mathfrak{p})}$, …, $U^{(k+1)}$ (involved in the definition of A). The prefix of this strategy ensures that the set E is reachable from the initial states, and hence that the states of E are in the induced DTMC $\mathcal{M}^\sigma$. We let $P^{[0]}, \ldots P^{[\mathfrak{n}-1]}$ be matrices that conform to $\mathcal{M}$, such that $E \cap \mathtt{Reach}(\mathsf{supp}(\varsigma), P^{[0,\mathfrak{n}-1]}) \neq \emptyset$; for instance, we can take $P^{[i]}$ to be the matrix corresponding to choosing successors in $\mathsf{Player}\ \Box$ states with uniform probability. Then we define the periodic phase with $\mathfrak{p}$ matrices by letting $P^{[\mathfrak{n}+i]} = U^{(k+\mathfrak{p}-i)}$ for $0 \leq i \leq \mathfrak{p} - 1$.

Note that $P^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]} = U^{(k+\mathfrak{p},k)}$ and $\mathbf{y} \stackrel{\text{def}}{=} [\hat{U}^{(k+\mathfrak{p},k)} \cdot \mathbf{r}]_E = [\hat{P}^{[\mathfrak{n},\mathfrak{n}+\mathfrak{p}]} \cdot \mathbf{r}]_E$. Further, for states $s \in E \cap \mathtt{Reach}(\mathsf{supp}(\varsigma), P^{[0,\mathfrak{n}-1]})$, we have that the state $(s, \mathfrak{n})$ is in the induced DTMC $\mathcal{M}^\sigma$. We now show that $e_{(s,\mathfrak{n})}^{j\mathfrak{p}} \to -\infty$ as $j \to \infty$, and hence that the strategy σ spoils $\mathsf{EE}(r - \varepsilon)$. From Lemma 22 we have $[e_{(\mathfrak{n})}^{j\mathfrak{p}}(r-\varepsilon)]_E = \sum_{k=0}^{j-1} A^k \cdot \mathbf{y} - j\mathfrak{p}\varepsilon$. It remains to show that the sequence $\sum_{k=0}^{j-1} A^k \cdot \mathbf{y}$ is upper-bounded, in order to have convergence of $\mathbf{e}_{(s,\mathfrak{n})}^{j\mathfrak{p}}$ toward $-\infty$. We have $\mathbf{y} = \mathbf{u}_E^{k+\mathfrak{p}} - A \cdot \mathbf{u}_E^k \leq (I - A) \cdot \mathbf{u}_E^k$, and thus

$$\left(\sum_{i=0}^{j-1} A^i\right) \cdot \mathbf{y} \leq \left(\sum_{i=0}^{j-1} A^i\right) \cdot (I - A) \cdot \mathbf{u}_E^k = (I - A^j) \cdot \mathbf{u}_E^k \leq -A^j \cdot \mathbf{u}_E^k \leq \|\mathbf{u}_E^k\|_\infty \cdot \mathbf{1},$$

where we use for the last inequality that $\|A^j\|_\infty = 1$, since A^j is stochastic. □

Appendix B.4. Proof of Lemma 4

Proof. Fix a Player $\Box$ strategy σ for $\mathcal{M}$. We first show by induction on k that $u_s^k \leq e_{s,\mathfrak{m}}^k$ for every s and $\mathfrak{m}$. The base case for $k = 0$ is satisfied as $e_{s,\mathfrak{m}}^k = u_s^k = 0$. Now assume that $u_s^k \leq e_{s,\mathfrak{m}}^k$ holds for some k and for every s, $\mathfrak{m}$, and we show it holds for $k+1$. In each Player $\Box$ state s, we have

$$\begin{aligned}
u_s^{k+1} &\leq r(s) + \min_{t \in \Delta(s)} u_t^k && \text{(definition)} \\
&\leq r(s) + \sum_{(t,\mathfrak{m}') \in \Delta^\sigma(s,\mathfrak{m})} \Delta^\sigma((s,\mathfrak{m}),(t,\mathfrak{m}')) u_t^k \\
&\leq r(s) + \sum_{(t,\mathfrak{m}') \in \Delta^\sigma(s,\mathfrak{m})} \Delta^\sigma((s,\mathfrak{m}),(t,\mathfrak{m}')) e_{t,\mathfrak{m}'}^k && \text{(induction hypothesis)} \\
&= e_{s,\mathfrak{m}}^{k+1}. && \text{(definition)}
\end{aligned}$$

Since Player $\Box$ can falsify $\mathsf{EE}(r)$, for every v_0 there is $(s,\mathfrak{m})$ such that $e_{s,\mathfrak{m}}^k \leq v_0$ and hence $u_s^* \leq u_s^k \leq e_{s,\mathfrak{m}}^k \leq v_0$. As $\mathcal{M}$ is finite and v_0 can be taken arbitrary low, it means that there is one state for which $u_s^* = -\infty$, and thus $S_\infty \neq \emptyset$. $\Box$

Appendix B.5. Proof of Lemma 5

Proof. Instead of proving $\forall\sigma\,.\,\mathcal{G}^{\pi,\sigma} \models \psi \Rightarrow \forall\sigma\,.\,\mathcal{G}^{\pi,\sigma} \models \varphi$, we prove the stronger statement $\forall\sigma\,.\,(\mathcal{G}^{\pi,\sigma} \models \psi \Rightarrow \mathcal{G}^{\pi,\sigma} \models \varphi)$. Fix finite strategies π and σ. Let $\mathcal{D} = \mathcal{G}^{\pi,\sigma}$, which is a finite DTMC. By Lemma 15, the limit $\lim_{N\to\infty} \frac{1}{N+1}\mathsf{rew}^N(\vec{r})$ almost surely exists. For every N and path λ, we have $|\frac{1}{N+1}\mathsf{rew}^N(\vec{r})(\lambda)| \leq \max_{s\in S_\mathcal{D}} |\vec{r}(s)|$, where the maximum is taken componentwise, and so we have

$$\mathbb{E}_{\mathcal{D},s}\left[\lim_{N\to\infty} \tfrac{1}{N+1}\mathsf{rew}^N(\vec{r})\right] = \lim_{N\to\infty} \mathbb{E}_{\mathcal{D},s}\left[\tfrac{1}{N+1}\mathsf{rew}^N(\vec{r})\right] \tag{B.7}$$

by the Lebesgue dominated convergence theorem.

Proof of (i). By Theorem 3 it suffices to consider MD Player $\Box$ strategies. Assume that $\mathsf{EE}(\vec{r})$ is satisfied. Fix a finite shortfall $\vec{v}_0$ such that, for all $s \in S_\mathcal{D}$, it holds that

$$\begin{aligned}
&\forall N \geq 0\,.\,\mathbb{E}_{\mathcal{D},s}[\mathsf{rew}^N(\vec{r})] \geq \vec{v}_0 && \text{(by assumption)} \\
&\forall N \geq 0\,.\,\mathbb{E}_{\mathcal{D},s}[\tfrac{1}{N+1}\mathsf{rew}^N(\vec{r})] \geq \tfrac{\vec{v}_0}{N+1} && \text{(dividing by } N+1) \\
&\lim_{N\to\infty} \mathbb{E}_{\mathcal{D},s}\left[\tfrac{1}{N+1}\mathsf{rew}^N(\vec{r})\right] \geq 0 && \text{(taking limits)} \\
&\mathbb{E}_{\mathcal{D},s}\left[\lim_{N\to\infty} \tfrac{1}{N+1}\mathsf{rew}^N(\vec{r})\right] \geq 0. && \text{(by (B.7))}
\end{aligned}$$

From Lemma 15, whenever s is in a BSCC $\mathcal{B}$ of $\mathcal{D}$ (that is, $\mathbb{P}_{\mathcal{D},s}(\mathsf{F}\,\mathcal{B}) = 1$), we have $\mathsf{mp}(\vec{r})(\mathcal{B}) = \mathbb{E}_{\mathcal{D},s}[\lim_{N\to\infty} \frac{1}{N+1}\mathsf{rew}^N(\vec{r})]$. Therefore, for every BSCC $\mathcal{B}$, $\mathsf{mp}(\vec{r})(\mathcal{B}) \geq \vec{0}$. Thus, again by Lemma 15, $\mathsf{Pmp}(\vec{r})$ is satisfied.

Proof of (ii). Assume π is DU, and so, by Proposition 6, it suffices to consider finite Player $\Box$ strategies. Fix $\varepsilon > 0$. Assume that $\mathcal{D} \models \mathsf{Pmp}(\vec{r})$, and so, by

Lemma 15, $\mathsf{rew}(\vec{r})(\mathcal{B}) \geq 0$ for every BSCC $\mathcal{B}$ of $\mathcal{D}$. Thus, for all states $s \in S_{\mathcal{D}}$, we have

$$\lim_{N\to\infty} \mathbb{E}_{\mathcal{D},s}[\tfrac{1}{N+1}\mathsf{rew}^N(\vec{r})] \geq \vec{0} \qquad \text{(by (B.7))}$$

$$\exists N_{\varepsilon,s} \geq 0 \,.\, \forall N \geq N_{\varepsilon,s} \,.\, \mathbb{E}_{\mathcal{D},s}[\tfrac{1}{N+1}\mathsf{rew}^N(\vec{r})] \geq -\vec{\varepsilon} \qquad \text{(definition of limit)}$$

$$\forall N \geq 0 \,.\, \mathbb{E}_{\mathcal{D},s}[\mathsf{rew}^N(\vec{r})] \geq -(N+1)\cdot\vec{\varepsilon} + \vec{v}_0^s$$

$$\text{(fixing } N_{\varepsilon,s} \text{ and letting } v_{0,i}^s \stackrel{\text{def}}{=} \min_{N \leq N_{\varepsilon,s}} \mathbb{E}_{\mathcal{D},s}[\mathsf{rew}^N(r_i)])$$

$$\forall N \geq 0 \,.\, \mathbb{E}_{\mathcal{D},s}[\mathsf{rew}^N(\vec{r}+\varepsilon)] \geq \vec{v}_0^s \geq \vec{v}_0. \qquad \text{(letting } v_{0,i} \stackrel{\text{def}}{=} \min_{s\in S_{\mathcal{D}}} v_{0,i}^s)$$

Since $\vec{v}_0$ is finite, $\mathcal{D}$ satisfies $\mathsf{EE}(\vec{r}+\vec{\varepsilon})$. □

Appendix B.6. Proof of Lemma 6

Proof. Let f_π be an ε-consistent memory mapping for $\vec{v}_0$, and we write $\vec{\mathfrak{m}}_s$ for $f_\pi(\mathfrak{m}, s)$. Let σ be a Player □ strategy, let $\mathcal{D} \stackrel{\text{def}}{=} \mathcal{G}^{\pi,\sigma}$, and let s_o be a state of $\mathcal{D}$, which has the form $s_o = (p_o, \vec{\mathfrak{m}}_{p_o}, \mathfrak{n})$, where $\vec{\mathfrak{m}}_{p_o}$ is the mapped memory of Player ◇. We show that $\mathbb{E}_{\mathcal{D},s_o}[\mathsf{rew}^N(\vec{r})] \geq \vec{\mathfrak{m}}_{p_o} - N\varepsilon$. For this we show that the memory of π is always above $\vec{\mathfrak{m}}_{p_o} - \mathbb{E}_{\mathcal{D},s_o}[\mathsf{rew}^N(\vec{r})] - N\varepsilon$, and, since this memory is always non-positive, we get the desired result.

Let $Y_N : \Omega_{\mathcal{D}} \to \mathbb{R}^n$ be the random variable that assigns $\vec{\mathfrak{m}}_s$ to a path $\lambda = s_0 s_1 \ldots$ for which $s_N = (s, \vec{\mathfrak{m}}_s, \mathfrak{n})$. Since $\mathbb{E}_{\mathcal{D},s_o}[Y_N] \leq \vec{0}$ for all $N \geq 0$, it is sufficient to show, for all s_o, that

$$\mathbb{E}_{\mathcal{D},s_o}[Y_N] \geq \vec{\mathfrak{m}}_{p_o} - \mathbb{E}_{D,s_o}[\mathsf{rew}^N(\vec{r})] - N\cdot\varepsilon \tag{B.8}$$

in order to conclude that $\mathbb{E}_{\mathcal{D},s_o}[\mathsf{rew}^N(\vec{r})] \geq \vec{0}$, and thus that $\mathcal{D}$ satisfies $\mathsf{EE}(\vec{r}+\vec{\varepsilon})$.

We show (B.8) by induction on the length N of paths $\Omega_{\mathcal{D}}$. In the base case, for $N = 0$, we have $\mathbb{E}_{\mathcal{D},s_o}[Y_0] = \vec{\mathfrak{m}}_{p_o}$, corresponding to the memory at the initial state s_o. For the induction step, assume that $\mathbb{E}_{\mathcal{D},s_o}[Y_N] \geq \vec{\mathfrak{m}}_{p_o} - \mathbb{E}_{\mathcal{D},s_o}[\mathsf{rew}^N(\vec{r})] - N\varepsilon$. Let W_N be the set of all finite paths of length N in $\mathcal{D}$, and we use the notation $\lambda' = \lambda(s, \vec{\mathfrak{m}}_s, \mathfrak{n})$ for paths $\lambda' \in W_N$. We have

$$\mathbb{E}_{\mathcal{D},s_o}[Y_{N+1}|\lambda'] = \begin{cases} \sum_t \pi_{\mathsf{c}}(s,\mathfrak{m})(t)\cdot\sum_{\mathfrak{m}'}\pi_{\mathsf{u}}(\mathfrak{m},t)(\mathfrak{m}')\cdot\vec{\mathfrak{m}}'_t & \text{if } s\in S_\Diamond \\ \sum_t \sigma_{\mathsf{c}}(s,\mathfrak{n})(t)\cdot\sum_{\mathfrak{m}'}\pi_{\mathsf{u}}(\mathfrak{m},t)(\mathfrak{m}')\cdot\vec{\mathfrak{m}}'_t & \text{if } s\in S_\Box \\ \sum_t \mu(t)\cdot\sum_{\mathfrak{m}'}\pi_{\mathsf{u}}(\mathfrak{m},t)(\mathfrak{m}')\cdot\vec{\mathfrak{m}}'_t & \text{if } s=(a,\mu)\in S_\bigcirc. \end{cases}$$

Therefore, by the ε-consistency of the memory mapping f_π, we have

$$\mathbb{E}_{\mathcal{D},s_o}[Y_{N+1}|\lambda'] \geq \vec{\mathfrak{m}}_s - \vec{r}(s) - \varepsilon. \tag{B.9}$$

Further, evaluating expectations over paths in W_N yields

$$\mathbb{E}_{\mathcal{D},s_o}[\mathsf{rew}^{N+1}(\vec{r})] - \mathbb{E}_{\mathcal{D},s_o}[\mathsf{rew}^N(\vec{r})] = \textstyle\sum_{\lambda'\in W_N}\vec{r}(s)\cdot\mathbb{P}_{\mathcal{D},s_o}(\lambda') \tag{B.10}$$

$$\mathbb{E}_{\mathcal{D},s_o}[Y_N] = \textstyle\sum_{\lambda'\in W_N}\mathbb{P}_{\mathcal{D},s_o}(\lambda')\cdot\vec{\mathfrak{m}}_s. \tag{B.11}$$

We can now conclude our induction step to establish (B.8) as follows:

$$\begin{aligned}
\mathbb{E}_{\mathcal{D},s_o}[Y_{N+1}] &= \textstyle\sum_{\lambda' \in W_N} \mathbb{E}_{\mathcal{D},s_o}[Y_{N+1}|\lambda'] \cdot \mathbb{P}_{\mathcal{D},s_o}(\lambda') && \text{(law of total probability)}\\
&\geq \textstyle\sum_{\lambda' \in W_N} (\vec{\mathfrak{m}}_s - \vec{r}(s) - \varepsilon) \cdot \mathbb{P}_{\mathcal{D},s_o}(\lambda') && \text{(by equation (B.9))}\\
&= \mathbb{E}_{\mathcal{D},s_o}[Y_N] - (\mathbb{E}_{\mathcal{D},s_o}[\mathsf{rew}^{N+1}(\vec{r})] - \mathbb{E}_{\mathcal{D},s_o}[\mathsf{rew}^{N}(\vec{r})]) - \varepsilon\\
& && \text{(by equations (B.10) and (B.11))}\\
&\geq \vec{\mathfrak{m}}_{p_o} - \mathbb{E}_{\mathcal{D},s_o}[\mathsf{rew}^{N+1}(\vec{r})] - (N+1) \cdot \varepsilon. && \text{(induction hypothesis)}
\end{aligned}$$

□

Appendix B.7. Proof of Proposition 8

We first recall concepts about fixpoints from [22]. Given a partially ordered set $\mathcal{C}$ with a partial order $\preceq$, and a set $Y \subseteq \mathcal{C}$, an element $x \in \mathcal{C}$ is an *upper bound* of Y if $y \preceq x$ for all $y \in Y$, and the *supremum* of Y is its least upper bound, written $\sup Y$. Given a map $\Phi : \mathcal{C} \to \mathcal{C}$, we say that $x \in \mathcal{C}$ is a *fixpoint* of Φ if $\Phi(x) = x$. We write $\mathsf{fix}(\Phi)$ for the least fixpoint of Φ.

A nonempty subset D of an ordered set $\mathcal{C}$ is *directed* if, for every finite subset $F \subseteq D$, an upper bound of F is in D. An ordered set $\mathcal{C}$ is a *complete partially ordered set* (CPO) if $\sup D$ exists for each directed subset D of $\mathcal{C}$, and $\mathcal{C}$ has a *bottom element* $\bot$, which is the least element with respect to the order $\preceq$. A map $\Phi : \mathcal{C} \to \mathcal{C}$ over a CPO $\mathcal{C}$ is *Scott-continuous* if, for every directed set D in $\mathcal{C}$, $\Phi(\sup D) = \sup \Phi(D)$. By Lemma 3.15 in [22], every continuous map is *order-preserving*, meaning that $\Phi(x) \preceq \Phi(y)$ for all $x, y \in \mathcal{C}$ such that $x \preceq y$.

Theorem 18 (Theorem 4.5 (ii) in [22], Kleene fixpoint theorem)**.** *Let $\mathcal{C}$ be a CPO, and let $\Phi : \mathcal{C} \to \mathcal{C}$ be a Scott-continuous map. The least fixpoint $\mathsf{fix}(\Phi)$ exists and is equal to* $\sup_{k \geq 0} \Phi^k(\bot)$.

We now give more details on the set $\mathcal{C}_M$ and show that it is a CPO. For $D \subseteq \mathcal{C}_M$, the supremum $\sup D$ is defined via $[\sup\{X \in D\}]_s \overset{\text{def}}{=} \bigcap_{X \in D} X_s$ for all $s \in S$. The intersection of convex, closed, M-downward-closed sets is itself convex, closed, and M-downward-closed, and so $\sup D \in \mathcal{C}_M$ for any directed set D. Hence, $\mathcal{C}_M$ is a CPO.

Proof. The properties claimed in the proposition are consequences of Scott continuity of F_M and the Kleene fixpoint theorem, (Theorem 18). To show Scott continuity, it is sufficient to show that, for every countable directed set D, we have that $[F_M(\sup D)]_s = \sup([F_M(D)]_s)$ for all $s \in S$. Take any countable directed set $D = \{X^k \in \mathcal{C}_M \mid k \geq 0\} \subseteq \mathcal{C}_M$, and any $s \in S$. We first show intermediate results about this directed set D.

Lemma 24. *For finite $T \subseteq S$, $\mathsf{conv}(\bigcup_{t \in T} \bigcap_{k \geq 0} X^k_t) = \bigcap_{k \geq 0} \mathsf{conv}(\bigcup_{t \in T} X^k_t)$.*

Proof. We first define $Y^k \overset{\text{def}}{=} \mathsf{conv}(\bigcup_{t \in T} X^k_t)$, and let $Y^\infty \overset{\text{def}}{=} \bigcap_{k \geq 0} Y^k$. The sets X^k_t are compact and convex, and so their convex hull Y^k is also compact and convex, by Theorem 17.2 in [49]. Moreover, Y^k is M-downward closed, and so, for every k, $Y^k \in \mathcal{P}_{c,M}$.

We now show the equality of the lemma. For the $\subseteq$ direction, take $\vec{y} \in \mathsf{conv}(\bigcup_{t\in T} \bigcap_{k\geq 0} X_t^k)$. Then $\vec{y} = \sum_{t\in T} \mu(t) \cdot \vec{x}_t$ for some distribution $\mu \in \mathsf{D}(T)$ and some $\vec{x}_t \in \bigcap_{k\geq 0} X_t^k$. Hence, for every k, $\vec{y} \in Y^k$, and so $\vec{y} \in Y^\infty$.

For the $\supseteq$ direction, take $\vec{y}^\infty \in Y^\infty$. We note that, for every $k \geq 0$, $\vec{y}^\infty = \sum_{t\in T} \mu_k(t) \cdot \vec{x}_t^k$ for some distribution $\mu_k \in \mathsf{D}(T)$ and some vector $\vec{x}_t^k \in X_t^k$. The sets X^k are in $\mathcal{P}_{c,M}$, and thus compact, and so one can extract a subsequence of indices i_k such that μ_{i_k} and $\vec{x}_t^{i_k}$ converge toward limits, which we respectively denote μ and $\vec{x}_t$ for every $t \in T$. Moreover, $\lim_{k\to\infty} \vec{x}_t^{i_k} = \vec{x}_t \in Y_t^l$ for every $l \geq 0$ as Y^l is compact. Hence, $\vec{x}_t \in \bigcap_{k\geq 0} X_t^k$ for every t and we conclude $\vec{y}^\infty = \sum_{t\in T} \mu(t) \cdot \vec{x}_t \in \mathsf{conv}(\bigcup_{t\in T} \bigcap_{k\geq 0} X_t^k)$. □

Lemma 25. *For finite $T \subseteq S$, $\bigcap_{t\in T} \bigcap_{k\geq 0} X_t^k = \bigcap_{k\geq 0} \bigcap_{t\in T} X_t^k$.*

Proof. Straightforward reordering of countable intersections. □

Lemma 26. *For finite $T \subseteq S$, $\sum_{t\in T} \mu(t) \times \bigcap_{k\geq 0} X_t^k = \bigcap_{k\geq 0} \sum_{t\in T} \mu(t) \times X_t^k$.*

Proof. The $\subseteq$ direction is straightforward. For the $\supseteq$ direction, take $\vec{x} \in \bigcap_{k\geq 0} \sum_{t\in T} \mu(t) \times X_t^k$, and so, for all $k \geq 0$, there exist vectors $\vec{x}_t^k \in X_t^k$ for $t \in T$, such that $\vec{x} = \sum_{t\in T} \mu(t) \cdot \vec{x}_t^k$. We extract a subsequence of indices i_k such that $\vec{x}_t^{i_k}$ tends to a limit $\vec{x}_t$, which necessarily lies in $\bigcap_{k\geq 0} X_t^k$, by the same arguent as in Lemma 25. Hence $\vec{x} = \sum_{t\in T} \mu(t)\vec{x}_t \in \sum_{t\in T} \mu(t) \times \bigcap_{k\geq 0} X_t^k$. □

We now continue the proof of Proposition 8 by considering three cases. For $s \in S_\Diamond$, we have

$$
\begin{aligned}
[F_M(\sup(D))]_s &\stackrel{\text{def}}{=} \mathrm{Box}_M \cap \mathsf{dwc}(\vec{r}(s) + \mathsf{conv}(\textstyle\bigcup_{t\in\Delta(s)} \bigcap_{k\geq 0} X_t^k)) \\
&= \mathrm{Box}_M \cap \mathsf{dwc}(\vec{r}(s) + \textstyle\bigcap_{k\geq 0} \mathsf{conv}(\bigcup_{t\in\Delta(s)} X_t^k)) && \text{(Lemma 24)} \\
&= \textstyle\bigcap_{k\geq 0}(\mathrm{Box}_M \cap \mathsf{dwc}(\vec{r}(s) + \mathsf{conv}(\bigcup_{t\in\Delta(s)} X_t^k))) \\
&\stackrel{\text{def}}{=} [\sup F_M(D)]_s.
\end{aligned}
$$

For $s \in S_\Box$, we have

$$
\begin{aligned}
[F_M(\sup(D))]_s &= \mathrm{Box}_M \cap \mathsf{dwc}(\vec{r}(s) + \textstyle\bigcap_{t\in\Delta(s)} \bigcap_{k\geq 0} X_t^k) \\
&= \mathrm{Box}_M \cap \mathsf{dwc}(\vec{r}(s) + \textstyle\bigcap_{k\geq 0} \bigcap_{t\in\Delta(s)} X_t^k) && \text{(Lemma 25)} \\
&= \textstyle\bigcap_{k\geq 0}(\mathrm{Box}_M \cap \mathsf{dwc}(\vec{r}(s) + \bigcap_{t\in\Delta(s)} X_t^k)) \\
&\stackrel{\text{def}}{=} [\sup F_M(D)]_s.
\end{aligned}
$$

Finally, for $s \in S_{\bigcirc}$, we have

$$
\begin{aligned}
[F_M(\sup(D))]_s &\stackrel{\text{def}}{=} \mathrm{Box}_M \cap \mathsf{dwc}(\vec{r}(s) + \textstyle\sum_{t\in\Delta(s)} \Delta(s,t) \times \bigcap_{k\geq 0} X_t^k) \\
&= \mathrm{Box}_M \cap \mathsf{dwc}(\vec{r}(s) + \textstyle\bigcap_{k\geq 0} \sum_{t\in\Delta(s)} \Delta(s,t) \times X_t^k) \quad \text{(Lemma 26)} \\
&= \textstyle\bigcap_{k\geq 0}(\mathrm{Box}_M \cap \mathsf{dwc}(\vec{r}(s) + \sum_{t\in\Delta(s)} \Delta(s,t) \times X_t^k)) \\
&\stackrel{\text{def}}{=} [\sup F_M(D)]_s.
\end{aligned}
$$

This concludes the proof of Scott continuity for F_M. Then, by Theorem 18, the least fixpoint exists, and is equal to $\mathsf{fix}(F_M) = \bigcap_{k\geq 0} F_M^k(\bot_M)$. □

Appendix B.8. Proof of Proposition 9

Proof. We first show two intermediate lemmas. In Lemma 27, we show that we can consider the fixpoints $\mathsf{fix}[F_{M,\mathcal{M}}]_s$ for a PA $\mathcal{M}$, and in Lemma 28 we reduce the problem to the study of one-dimensional expected truncated energy, which we used earlier in Proposition 5 and Lemma 4.

Lemma 27. *Given a game $\mathcal{G}$, a DU strategy π and a constant M, if $[\mathsf{fix}(F_{M,\mathcal{G}^\pi})]_s \neq \emptyset$ for all $s \in S_{\mathcal{G}^\pi}$, then $[\mathsf{fix}(F_{M,\mathcal{G}})]_s \neq \emptyset$ for every $s \in \mathsf{supp}(\varsigma)$.*

Proof. We first describe how to compare elements of the CPOs $\mathcal{C}_{M,\mathcal{G}}$ and $\mathcal{C}_{M,\mathcal{G}^\pi}$ associated with $F_{M,\mathcal{G}^\pi}$ and $F_{M,\mathcal{G}}$, respectively. Given $X \in \mathcal{C}_{M,\mathcal{G}}$ and $Y \in \mathcal{C}_{M,\mathcal{G}^\pi}$ we say that $Y \succ X$ if the following conditions are satisfied:

$$
\begin{cases}
Y_{(s,\mathfrak{m})} \subseteq X_s & \text{for } (s,\mathfrak{m}) \in S_{\mathcal{G}^\pi} \text{ with } s \in S_\Box \cup S_{\bigcirc}; \\
\sum_{s'\in\Delta(s)} \pi_{\mathsf{c}}(s,\mathfrak{m})(s') Y_{((s,s'),\mathfrak{m})} \subseteq X_s & \text{for } s \in S_\Diamond \text{ and } \mathfrak{m} \text{ such that } ((s,s'),\mathfrak{m}) \in S_{\mathcal{G}^\pi} \text{ for some } s' \in S_{\bigcirc}
\end{cases}
$$

We now show that $\mathsf{fix}(F_{M,\mathcal{G}^\pi}) \succ \mathsf{fix}(F_{M,\mathcal{G}})$. Recall that $\mathsf{fix}(F_{M,\mathcal{G}^\pi}) = \cap_{k\in\mathbb{N}} Y^k$ and $\mathsf{fix}(F_{M,\mathcal{G}}) = \cap_{k\in\mathbb{N}} X^k$ where $Y^k \stackrel{\text{def}}{=} F^k_{M,\mathcal{G}^\pi}(\bot_M)$ and $X^k \stackrel{\text{def}}{=} F^k_{M,\mathcal{G}^\pi}(\bot_M)$. It hence suffices to show by induction that, for every $k \in \mathbb{N}$, $Y^k \succ X^k$.

For $k = 0$, the property holds as all sets involved are equal to Box_M. We now assume that the property is proved at rank $k-1$ and show that it holds at rank k.

Let $s \in S_\Diamond$ and $\mathfrak{m}$ such that $((s,s'),\mathfrak{m}) \in S_{\mathcal{G}^\pi}$ for some $s' \in S_{\bigcirc}$. It holds that

$$
\begin{aligned}
\sum_{s'\in\Delta(s)} \pi_{\mathsf{c}}(s,\mathfrak{m})(s') Y^k_{((s,s'),\mathfrak{m})} &= \sum_{s'\in\Delta(s)} \pi_{\mathsf{c}}(s,\mathfrak{m})(s') \mathrm{Box}_M \cap \mathsf{dwc}\left(\vec{r}(s) + Y^{k-1}_{(s',\pi_{\mathsf{u}}(\mathfrak{m},s'))}\right) \\
&\subseteq \sum_{s'\in\Delta(s)} \pi_{\mathsf{c}}(s,\mathfrak{m})(s') \mathrm{Box}_M \cap \mathsf{dwc}\left(\vec{r}(s) + X^{k-1}_{s'}\right) \\
&\subseteq \mathsf{conv}\left(\bigcup_{s'\in\Delta(s)} \mathrm{Box}_M \cap \mathsf{dwc}\left(\vec{r}(s) + X^{k-1}_{s'}\right)\right) \\
&= X^k_s.
\end{aligned}
$$

Let $(s, \mathfrak{m}) \in \mathcal{G}^\pi$ with $s \in S_\Box$. It holds that

$$\begin{aligned} Y^k_{(s,\mathfrak{m})} &= \mathrm{Box}_M \cap \mathsf{dwc}\left(\vec{r}(s) + \bigcap_{t\in\Delta(s)} Y^{k-1}_{(t,\pi_\mathsf{u}(\mathfrak{m},t))}\right) \\ &\subseteq \mathrm{Box}_M \cap \mathsf{dwc}\left(\vec{r}(s) + \bigcap_{t\in\Delta(s)} X^{k-1}_t\right) \\ &= X^k_s. \end{aligned}$$

Let $(s, \mathfrak{m}) \in \mathcal{G}^\pi$ with $s \in S_\bigcirc$. It holds that

$$Y^k_{(s,\mathfrak{m})} = \mathrm{Box}_M \cap \mathsf{dwc}\left(\vec{r}(s) + E_1 + E_2\right)$$

where

$$E_1 \stackrel{\text{def}}{=} \sum_{s'\in\Delta(s)\cap S_\Box} \mu(s') Y^{k-1}_{(s',\pi_\mathsf{u}(\mathfrak{m},s'))}$$

and

$$E_2 \stackrel{\text{def}}{=} \sum_{s'\in\Delta(s)\cap S_\Diamond} \mu(s') \sum_{s''\in\Delta(s')} \pi_\mathsf{c}(s',\pi_\mathsf{u}(\mathfrak{m},s'))(s'') Y^{k-1}_{(s's'',\pi_\mathsf{u}(\mathfrak{m},s'))}.$$

Applying the induction hypothesis yields

$$Y^k_{s,\mathfrak{m}} \subseteq \mathrm{Box}_M \cap \mathsf{dwc}\left(\vec{r}(s) + \sum_{s'\in\Delta(s)} \mu(s') X^{k-1}_{s'}\right) = X^k_s.$$

We have shown by induction that, for every $k \in \mathbb{N}$, $Y^k \succ X^k$. Thus $\mathsf{fix}(F_{M,\mathcal{G}^\pi}) \succ \mathsf{fix}(F_{M,\mathcal{G}})$. The conclusion of the lemma follows. □

Lemma 28. *Given a PA $\mathcal{M}$ with rewards $\vec{r}$ and a state s, if $\mathsf{fix}[F_{M,\mathcal{M}}]_s = \emptyset$ for every $M < \infty$, then there exists i such that $u^*_s = -\infty$ for the reward r_i.*

Proof. Fix a PA $\mathcal{M} = \langle S, (S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta\rangle$. We prove the lemma by contraposition: given a state s_0, we assume that $u^*_{s_0} > -\infty$ for rewards r_i for all i, and show that there is an M for which $\mathsf{fix}[F_{M,\mathcal{M}}]_{s_0} \neq \emptyset$. We consider a multi-dimensional version of the truncated energy sequence defined in (2), and get that the fixpoint of the multi-dimensional truncated energy, as $k \to \infty$, is

$$\vec{u}^*_s = \begin{cases} \min(\vec{0}, \vec{r}(s) + \min_{t\in\Delta(s)} \vec{u}^*_t) & \text{if } s \in S_\Box \\ \min(\vec{0}, \vec{r}(s) + \sum_{t\in\Delta(s)} \Delta(s,t)\vec{u}^*_t & \text{if } s \in S_\bigcirc, \end{cases}$$

where the minima are taken componentwise.

Observe that, for a state s, if $\vec{u}^*_s$ has no infinite coordinate, then neither do its successors. As all states of the PA are reachable from the initial state, then for every state s, $\vec{u}^*_s$ has no infinite coordinate. Therefore, there is a global bound M, such that $\vec{u}_s^* \in \mathrm{Box}_M$ for every s. We now show that $Y \in \mathcal{C}_M$, defined by $Y_s \stackrel{\text{def}}{=} \mathrm{Box}_M \cap \mathsf{dwc}(\vec{u}_s^*)$, is a fixpoint of $F_{M,\mathcal{M}}$, and hence that the least-fixpoint of $F_{M,\mathcal{M}}$ is non-empty. Taking the downward-closure gives

$$\mathsf{dwc}(\vec{u}_s^*) = \begin{cases} \mathbb{R}_{\leq 0} \cap (\vec{r}(s) + \bigcap_{t \in \Delta(s)} \mathsf{dwc}(\vec{u}_t^*)) & \text{if } s \in S_\Box \\ \mathbb{R}_{\leq 0} \cap (\vec{r}(s) + \sum_{t \in \Delta(s)} \Delta(s,t) \times \mathsf{dwc}(\vec{u}_t^*)) & \text{if } s \in S_\bigcirc, \end{cases}$$

and hence

$$Y_s = \begin{cases} \mathrm{Box}_M \cap (\vec{r}(s) + \bigcap_{t \in \Delta(s)} \mathsf{dwc}(\vec{u}_t^*)) & \text{if } s \in S_\Box \\ \mathrm{Box}_M \cap (\vec{r}(s) + \sum_{t \in \Delta(s)} \Delta(s,t) \times \mathsf{dwc}(\vec{u}_t^*)) & \text{if } s \in S_\bigcirc. \end{cases}$$

Since $\vec{u}_t^* \in \mathrm{Box}_M$, Y_t is nonempty, and we have

$$\begin{aligned} \vec{r}(s) + \textstyle\bigcap_{t \in \Delta(s)} \mathsf{dwc}(\vec{u}_t^*) &= \mathsf{dwc}(\vec{r}(s) + \textstyle\bigcap_{t \in \Delta(s)} Y_t) && \text{for } s \in S_\Box \\ \vec{r}(s) + \textstyle\sum_{t \in \Delta(s)} \Delta(s,t) \times \mathsf{dwc}(\vec{u}_t^*) &= \mathsf{dwc}(\vec{r}(s) + \textstyle\sum_{t \in \Delta(s)} \Delta(s,t) \times Y_t) && \text{for } s \in S_\bigcirc. \end{aligned}$$

This implies that $Y = F_{M,\mathcal{M}}(Y)$, and hence that $\mathsf{fix}[F_{M,\mathcal{M}}]_{s_0} \sqsubseteq Y_{s_0}$. We thus conclude from $Y_{s_0} \neq \emptyset$ that $\mathsf{fix}[F_{M,\mathcal{M}}]_{s_0} \neq \emptyset$. □

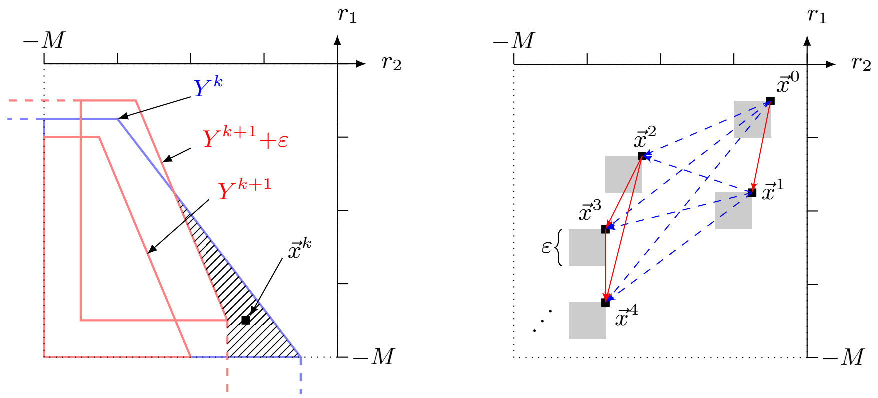
We can now conclude the proof of Proposition 9. Fix a game $\mathcal{G}$ and $\varepsilon > 0$. We show the contrapositive: if, for every M, $[\mathsf{fix}(F_{M,\mathcal{G}})]_s = \emptyset$ for some $s \in \mathsf{supp}(\varsigma)$, then $\mathsf{EE}(\vec{r} - \varepsilon)$ is not achievable by a finite strategy (against finite strategies). Assume that, for every M, $[\mathsf{fix}(F_{M,\mathcal{G}})]_s = \emptyset$, for some $s \in \mathsf{supp}(\varsigma)$, and let π be an arbitrary finite DU strategy. By Lemma 27, $[\mathsf{fix}(F_{M,\mathcal{G}^\pi})]_s = \emptyset$ for some $s \in S_{\mathcal{G}^\pi}$. Thus by Lemma 28 there is a dimension i such that $u_s^* = -\infty$ for some $s \in S_{\mathcal{G}^\pi}$ for the reward r_i, and hence $S_\infty \neq \emptyset$. We conclude, using Proposition 5, that Player □ can spoil $\mathsf{EE}(r - \varepsilon)$ in the PA $\mathcal{G}^\pi$. We have thus shown the contrapositive, that is, there is no winning strategy for Player ♢ to achieve $\mathsf{EE}(\vec{r} - \varepsilon)$, whenever, for every M, $[\mathsf{fix}(F_{M,\mathcal{G}})]_s = \emptyset$ for some $s \in \mathsf{supp}(\varsigma)$. □

Appendix B.9. Proof of Proposition 10

The proof we use a Ramsey like theorem (Theorem 19). We first recall the necessary definitions. A *graph* $G = (V, E)$ consists of a finite set V of *nodes* and a set $E \subseteq V \times V$ of *edges*. A graph is *linearly-ordered complete*, if for some strict linear order $\succ$ on V, $(v, w) \in E$ if and only if $v \succ w$. An *n-colouring* of a graph (V, E) is a function $E \to \{1, \ldots, n\}$, assigning one of n possible colours to each edge. A *monochromatic directed path of length N* is a sequence of nodes $v_1, \ldots, v_N$ such that $(v_i, v_{i+1}) \in E$ for all $1 \leq i < N$, and such that each node v_i is assigned the same colour.

Theorem 19 (Theorem 4.5.2 of [51])**.** *Let $G = (V, E)$ be a linearly-ordered complete graph over m nodes, with an n-colouring of its edges. Then G contains a monochromatic directed path of length $\lfloor \sqrt{m/n - 2} \rfloor - 1$.*

We first consider a single state in Lemma 29, and then use an inductive argument on the number of states to find the bound for all states in Proposition 10.



(a) The hatched region is $Y^k \cap (\mathrm{Box}_M \setminus \mathsf{dwc}(Y^{k+1} + \varepsilon))$, where $\vec{x}^k$ has to lie.

(b) The red (solid) and blue (dashed) arrows represent distance greater than ε in dimensions r_1 and r_2 resp.

Figure B.16: Illustrations for Lemma 29 for two dimensions r_1 and r_2.

Lemma 29. *Let $(Y^k)_{k\in\mathbb{N}}$ be a sequence over $\mathcal{P}_{c,M}$ that is non-decreasing for $\sqsubseteq$. For every $I \subseteq \mathbb{N}$ such that $|I| \geq k^* \overset{\text{def}}{=} n \cdot ((\lceil \frac{M}{\varepsilon} \rceil + 1)^2 + 2)$, there exists $k \in I$ such that $Y^{k+1} + \varepsilon \sqsubseteq Y^k$.*

Proof. Fix a sequence $(Y^k)_{k\in\mathbb{N}}$ non-decreasing for $\sqsubseteq$, and fix $I \subseteq \mathbb{N}$ such that $|I| \geq k^*$. We assume towards a contradiction that for every $k \in I$, $Y^{k+1} + \varepsilon \not\sqsubseteq Y^k$. Consider the linearly-ordered complete graph over nodes I, and with edges (j,k) for $j < k$ and $j,k \in I$. We define below an n-colouring c of this graph where colours represent dimensions of the M-polyhedrals, see Figure B.16 (b). Note first that, if two sets satisfy $B \not\sqsubseteq A$, then there exists $\vec{x} \in A \setminus \mathsf{dwc}(B)$. Hence, the hypothesis $Y^{k+1} + \varepsilon \not\sqsubseteq Y^k$ for every $k \in I$ implies the existence of a sequence $(\vec{x}^k)_{k\in I} \in Y^k \setminus \mathsf{dwc}(Y^{k+1} + \varepsilon)$ of points, illustrated in Figure B.16 (a). We show that, for all $j < k$, there exists a coordinate $c(j,k)$ for which $x^j_{c(j,k)} - x^k_{c(j,k)} > \varepsilon$ and define $c(j,k)$ as the colour of the edge (j,k). Assume otherwise, that is, $\vec{x}^j - \vec{\varepsilon} \leq \vec{x}^k$ for $j < k$. Then $\vec{x}^j - \vec{\varepsilon} \in \mathsf{dwc}(Y^k)$, and, since $Y^k \sqsubseteq Y^{j+1}$, we deduce $\vec{x}^j \in \mathsf{dwc}(Y^{j+1} + \vec{\varepsilon})$, a contradiction to the definition of the sequence $(\vec{x}^k)_{k\leq m}$. By Theorem 19, there exists a monochromatic path $j_1 \to j_2 \to \cdots \to j_l$ of length $l = \lfloor \sqrt{|I|/n - 2} \rfloor - 1 \geq \lceil \frac{M}{\varepsilon} \rceil$, and thus by denoting c the colour of this path it holds that $x^{j_1}_c > x^{j_2}_c + \varepsilon > \ldots > x^{j_l}_c + l\varepsilon \geq -M + \frac{M}{\varepsilon}\varepsilon \geq 0$, a contradiction. ☐

Lemma 30. *Let U be a finite set, let P be a predicate over $U \times \mathbb{N}$, and let K be a positive integer. The implication "P1 ⇒ P2" holds, where*

P1 *"For every $s \in U$ and every $I \subseteq \mathbb{N}$ such that $|I| \geq K$, there exists $i \in I$, such that $P(s,i)$ holds."*

P2 "For every $I \subseteq \mathbb{N}$ such that $|I| \geq K^{|U|}$, there exists $i \in I$ such that, for every $s \in U$, $P(s, i)$ holds."

Proof. We show the result by induction on the cardinality of U. If U is empty the result is true. Now assume that the implication "P1 ⇒ P2" holds for sets U' of cardinality c, and let $U = U' \cup \{t\}$ be of cardinality $c+1$. Let P be a predicate over $U \times \mathbb{N}$ and let K be a positive integer, such that P1 is satisfied for U. Let $I \subseteq \mathbb{N}$ such that $|I| \geq K^{|U|}$. We want to find an index i such that $P(s, i)$ holds for all $s \in U$. We partition I into K parts $I_1, \ldots, I_K$, each containing at least $K^{|U|-1}$ elements. Since P1 is satisfied for U, it is also satisfied for $U \setminus \{t\}$, and so, by the induction hypothesis, for every I_k there is an index $i_k \in I_k$ such that, for every $s \in U \setminus \{t\}$, $P(s, i_k)$ holds. The set $\{i_1, \ldots, i_K\}$ contains K elements and hence we can apply P1 (which holds for U by assumption), and extract one i such that also $P(t, i)$ is true. Hence, i is such that for every $s \in U$, $P(s, i)$ is true, concluding the induction step.

□

We can now conclude the proof of Proposition 10.

Fix M and $\varepsilon > 0$. Let $\mathcal{G}$ be a game with state space S. Let $(X^k)_{k \geq 0}$ be a sequence over $\mathcal{C}_M$ that is non-decreasing for $\sqsubseteq$. We apply Lemma 30 with $U = S$, $K = k^*$, and with the predicate $X_s^{k+1} + \varepsilon \sqsubseteq X_s^k$ for P, noting that P1 is satisfied by Lemma 29, and that P2 is the statement we set out to prove. □

Appendix B.10. Proof of Lemma 7

Proof. Let $X \in \mathcal{C}_M$ such that $F_M(X) + \varepsilon \sqsubseteq X$ and $[F_M(X)]_s \neq \emptyset$ for every $s \in \mathsf{supp}(\varsigma)$. We now show that the strategy constructed in Section 3.4.3 is well-defined. First note that $s \in T_X$ for every $s \in \mathsf{supp}(\varsigma)$, and, if $s \in T_X \cap (S_\Box \cup S_\bigcirc)$, then, for every $t \in \mathsf{succ}(s)$, $[F_M(X)]_t + \varepsilon \sqsubseteq X_t \neq \emptyset$, and hence $t \in T_X$.

For any $s \in T_X$, depending on the type of s (i.e. Player ◇, Player □, or move), we define an auxiliary set Y_s without the cut-off by Box_M. We then show that we can find the required distributions α and β, and the extreme points for every point in Y_s, and prove that for all extreme points $\vec{p}$ of X_s we have $\vec{p} - \varepsilon$ in Y_s for $k \geq 0$, allowing us to show well-definedness of the strategy. Take $s \in T_X$.

- **Case** $s \in S_\Diamond$**.** Let $Y_s \stackrel{\text{def}}{=} \vec{r}(s) + \mathsf{conv}(\bigcup_{t \in \Delta(s) \cap T_X} X_t)$. Take any $\vec{p}' \in Y_s$. There are distributions $\alpha \in \mathsf{D}(\Delta(s) \cap T_X)$, $\beta^t \in \mathsf{D}([1, n])$, and points $\vec{q}_i^{\,t} \in \mathsf{C}(X_t)$ for $t \in \Delta(s) \cap T_X$, such that $\sum_t \alpha(t) \cdot \sum_i \beta^t(i) \cdot \vec{q}_i^{\,t} \geq \vec{p}' - \vec{r}(s)$.

- **Case** $s \in S_\Box$**.** Let $Y_s \stackrel{\text{def}}{=} \mathsf{dwc}(\vec{r}(s) + \bigcap_{t \in \Delta(s)} X_t)$. Take any $\vec{p}' \in Y_s$. For any $t \in \Delta(s)$, there are distributions $\beta^t \in \mathsf{D}([1, n])$ and points $\vec{q}_i^{\,t} \in \mathsf{C}(X_t)$ such that $\sum_i \beta_i^t \cdot \vec{q}_i^{\,t} \geq \vec{p}' - \vec{r}(s)$.

- **Case** $s = (a, \mu) \in S_\bigcirc$**.** Let $Y_s \stackrel{\text{def}}{=} \vec{r}(s) + \sum_{t \in \mathsf{supp}(\mu)} \mu(t) \times X_t$. Take any $\vec{p} \in Y_s$. Due to the Minkowski sum, there are distributions $\beta^t \in \mathsf{D}([1, n])$ and points $\vec{q}_i^{\,t} \in \mathsf{C}(X_t)$ such that $\sum_{t \in \mathsf{supp}(\mu)} \mu(t) \cdot \sum_i \beta_i^t \cdot \vec{q}_i^{\,t} \geq \vec{p}' - \vec{r}(s)$.

Note that, if two sets satisfy $A \sqsubseteq B$, they also satisfy $A-\varepsilon \sqsubseteq B-\varepsilon$. We have $F_M(X)+\varepsilon \sqsubseteq X$, and so $\mathsf{dwc}(Y_s) \cap \mathrm{Box}_M = [F_M(X)]_s \sqsubseteq X_s - \vec{\varepsilon}$, for all $s \in T_X$. Then, for any point $\vec{p} \in \mathsf{C}(X_s)$, it holds that $\vec{p} - \varepsilon \in \mathsf{dwc}(Y_s) \cap \mathrm{Box}_M$. Hence, we can find for $\vec{p}' = \vec{p} - \vec{\varepsilon}$ the corresponding distributions and extreme points to construct the strategy π, together with the memory mapping f_π, which is ε-consistent for $\vec{q}_0^s \geq -M$. □

Appendix C. Proofs of results of Section 4

Appendix C.1. Proof that expected ratio rewards are globally-bounded

Lemma 31. *$\mathsf{ratio}(r/c)$ is integrable and globally bounded by $B \stackrel{\text{def}}{=} \max_S r(s)/c_{\min}$.*

Proof. Fix two strategies π, σ. The function $|\mathsf{ratio}(r/c)|$ is non-negative and measurable, so the quantity $\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}(|\mathsf{ratio}(r/c)|)$ is well-defined in $\mathbb{R}_{\geq 0} \cup \{+\infty\}$. We show that this quantity is finite and bounded by B independently of π, σ. We let $\rho^* = \max_S r(s)$ and use that, for every N, $\frac{\mathsf{rew}^N(r)}{N+1} \leq \rho^*$. Hence,

$$\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}(|\mathsf{ratio}(r/c)|) = \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left(\underline{\lim}_{N\to\infty} \frac{|\mathsf{rew}^N(r)(\lambda)|}{1+\mathsf{rew}^N(c)(\lambda)}\right) \leq \rho^* \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left(\underline{\lim}_{N\to\infty} \frac{N+1}{1+\mathsf{rew}^N(c)(\lambda)}\right).$$

Note that for a sequence $(x_N)_{N\geq 0}$ of positive numbers it holds that

$$\underline{\lim}_{N\to\infty} \frac{1}{x_N} = \frac{1}{\overline{\lim}_{N\to\infty} x_N} \leq \frac{1}{\underline{\lim}_{N\to\infty} x_N}.$$

This implies that almost surely

$$\underline{\lim}_{N\to\infty} \frac{N+1}{1+\mathsf{rew}^N(c)(\lambda)} \leq \frac{1}{\underline{\lim}_{N\to\infty}\left(\frac{1+\mathsf{rew}^N(c)(\lambda)}{N+1}\right)} = \frac{1}{\mathsf{mp}(c)(\lambda)} \leq \frac{1}{c_{\min}}.$$

Hence, $\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}(|\mathsf{ratio}(r/c)|) \leq \max_S r(s)/c_{\min}$ as expected. □

Appendix C.2. Proof of Theorem 9

Proof. Consider $\vec{u} \in \mathrm{Pareto}(\psi)$, then the vector $\vec{u} - \varepsilon/4$ is achievable. Using Theorem 8, there exists a vector $\vec{y} \in \mathbb{R}^{\mathcal{N}}$ with every $\vec{y}_i$ non-negative and non-null such that $\bigwedge_{i=1}^n \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{y}_i \cdot \vec{\varrho}_i] \geq \vec{y}_i \cdot (\vec{u}_i - \varepsilon/4)$ is achievable. Up to dividing each $\vec{y}_i$ by $\|\vec{y}_i\|_\infty$ we assume that $\|\vec{y}_i\|_\infty = 1$. Let $\vec{x}$ be such that $\vec{x} - \varepsilon/(4B) \leq \vec{y} \leq \vec{x}$ and such that each coordinate of $\vec{x}$ is multiple of $\varepsilon/(4B)$. It remains to show that $\bigwedge_{i=1}^n \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{x}_i \cdot \vec{\varrho}_i] \geq \vec{x}_i \cdot \vec{u}_i - \varepsilon$ and that $\vec{x} \in \mathsf{Grid}$. Fix $i \leq n$, we first note that $|(\vec{x}_i - \vec{y}_i) \cdot \vec{\varrho}_i| \leq \|\vec{x}_i - \vec{y}_i\|_\infty B \leq \varepsilon/4$. Hence

$$|\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[(\vec{x}_i - \vec{y}_i) \cdot \vec{\varrho}_i]| \leq \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[|(\vec{x}_i - \vec{y}_i) \cdot \vec{\varrho}_i|] \leq \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\varepsilon/4] = \varepsilon/4.$$

and then

$$\begin{aligned}
\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{x}_i \cdot \vec{\varrho}_i] &\geq \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\vec{y}_i \cdot \vec{\varrho}_i] - \varepsilon/4 \\
&\geq \vec{y}_i \cdot (\vec{u}_i - \varepsilon/4) - \varepsilon/4 \\
&\geq (\vec{x}_i - \varepsilon/(4B)) \cdot (\vec{u}_i - \varepsilon/4) - \varepsilon/4 \\
&\geq \vec{x}_i \cdot \vec{u}_i - (\varepsilon/4)(\|\vec{x}_i\|_\infty + \|\vec{u}_i\|_\infty/B) - \varepsilon/4 \\
&\geq \vec{x}_i \cdot \vec{u}_i - \varepsilon.
\end{aligned}$$

The last inequality is justified by $\|\vec{u}_i\|_\infty \leq B$ and $\|\vec{x}_i\|_\infty \leq \|\vec{y}_i\|_\infty + \varepsilon/(4B) \leq 1 + \varepsilon/(4B) \leq 2$. It also holds that $\|\vec{x}_i\|_\infty \geq \|\vec{y}_i\|_\infty - \varepsilon/(4B) \geq 1 - \varepsilon/(4B)$, and hence $\vec{x} \in \mathsf{Grid}$. □

Appendix C.3. Proof of Theorem 10

Proof. Take $\vec{u}$ an approximable target for $\bigwedge_{i=1}^{n} \bigvee_{j=1}^{m} \mathbb{E}[\varrho_{i,j}] \geq u_{i,j}$. Then we apply Theorem 9 with $\varepsilon/2$. Thus one can find a weight vector $\vec{x} \in \mathsf{Grid}$ such that $\vec{x} \cdot_n (\vec{u} - \varepsilon/2) \in P_{\varepsilon/2}(\vec{x})$. For every i, $\vec{x}_i$ has a positive component which is at least $\varepsilon/(8B)$, and hence $\vec{x} \cdot_n \varepsilon/2 \geq \varepsilon'$, where we define $\varepsilon' \stackrel{\text{def}}{=} \varepsilon^2/(16B)$. By assumption, one can synthesise an ε'-optimal strategy π for $\mathsf{E}(\vec{x} \cdot_n \vec{\varrho})(\vec{x} \cdot_n (\vec{u} - \varepsilon/2))$, meaning that π is winning for $\vec{x} \cdot_n (\vec{u} - \varepsilon/2) - \varepsilon'$. By definition of ε' it holds that $\vec{x} \cdot_n (\vec{u} - \varepsilon/2) - \varepsilon' \geq \vec{x} \cdot_n (\vec{u} - \varepsilon/2) - \vec{x} \cdot_n \varepsilon/2 = \vec{x} \cdot_n (\vec{u} - \varepsilon)$. Thus, π is winning for $\mathsf{E}(\vec{x} \cdot_n \vec{\varrho})(\vec{x} \cdot_n (\vec{u} - \varepsilon))$, and hence for $\bigwedge_{i=1}^{n} \bigvee_{j=1}^{m} \mathbb{E}[\varrho_{i,j}] \geq u_{i,j} - \varepsilon$. □

Appendix C.4. Proof of Theorem 11

Proof. By Proposition 2 and Remark 1, $\varphi_{\vec{x}}$ implies $\bigwedge_{i=1}^{n} \mathbb{E}(\mathsf{ratio}(\vec{x}_i \cdot \vec{r}_i/c_i)) \geq \vec{x}_i \cdot \vec{u}_i$. We need to consider only pairs of finite strategies as the statements are for finite $\mathsf{Player}\ \Diamond$ strategies winning against finite $\mathsf{Player}\ \Box$ strategies (we recall that $\varphi_{\vec{x}}$ is $\mathsf{Player}\ \Box$-positional by Theorem 3). Fix two finite strategies π, σ, then the induced DTMC $\mathcal{G}^{\pi,\sigma}$ is finite. Hence, by Proposition 13, $\mathsf{ratio}(\vec{x}_i \cdot \vec{r}_i/c_i) = \vec{x}_i \cdot \mathsf{ratio}(\vec{r}_i/c_i)$ and then $\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}(\mathsf{ratio}(\vec{x}_i \cdot \vec{r}_i/c_i) = \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}(\vec{x}_i \cdot \mathsf{ratio}(\vec{r}_i/c_i)$. Now we can apply Theorem 8 and deduce that π is winning for ψ against finite strategies whenever it is winning for $\varphi_{\vec{x}}$, where $c_{\min}$ is a bound such that for every i it holds that $\mathsf{mp}(c_i) \geq c_{\min}$ almost surely under any pair of strategies. Let $\varepsilon > 0$ and $\varepsilon' \stackrel{\text{def}}{=} \varepsilon \cdot c_{\min} \cdot \min(\vec{x}_i \cdot \vec{x}_i / \|\vec{x}_i\|_\infty)$. Let π, σ be two finite strategies. Now, note that almost surely $\varepsilon' \leq (\vec{x}_i \cdot \vec{x}_i) \cdot \mathsf{mp}(c_i) \cdot \varepsilon / \|\vec{x}_i\|_\infty$. Hence $\mathsf{mp}(\vec{x}_i \cdot \vec{r}_i - (\vec{x}_i \cdot \vec{u}_i) c_i) \geq -\varepsilon'$ implies $\mathsf{mp}(\vec{x}_i \cdot \vec{r}_i - (\vec{x}_i \cdot (\vec{u}_i - (\varepsilon/\|\vec{x}_i\|_\infty) \vec{x}_i)) c_i) \geq 0$. Thus, if π is ε'-optimal for $\varphi_{\vec{x}}$ then it is winning for ψ with the targets $u_{i,j} - (x_{i,j}/\|\vec{x}_i\|_\infty) \cdot \varepsilon$, and hence with the ε-optimal targets $u_{i,j} - \varepsilon$. □

Appendix C.5. Proof of Lemma 9

Proof. If $\mathsf{Pmp}(\vec{r})$ then $\mathsf{Emp}(\vec{r})$ by Remark 1. We show the other direction by contraposition. If $\mathsf{Pmp}(\vec{r})$ does not hold in a PA $\mathcal{M}$ with a single MEC, then there exists a finite strategy σ such that $\mathbb{P}_{\mathcal{M}}^{\sigma}(\mathsf{mp}(r_i) < 0) > 0$ for some i. By Lemma 15, there exists a BSCC $\mathcal{B}$ in the induced DTMC $\mathcal{M}^\sigma$ such that $\mathsf{mp}(r)(\mathcal{B}) < 0$. By Lemma 2, the set of states of the PA corresponding to the BSCC, formally given by $\mathcal{B}_{\mathcal{M}} \stackrel{\text{def}}{=} \{s \mid \exists \mathfrak{m}\,.\,(s,\mathfrak{m}) \in \mathcal{B}\}$, is reachable with probability one by an MD strategy from all states in $\mathcal{M}$. Hence, the strategy σ' that first reaches $\mathcal{B}_{\mathcal{M}}$ and then plays as σ to form the BSCC $\mathcal{B}$ is finite and induces a DTMC with a single BSCC $\mathcal{B}'$ in which the mean-payoff is $\mathsf{mp}(r)(\mathcal{B}') = \mathsf{mp}(r)(\mathcal{B}) < 0$. By Lemma 15, we have $\mathbb{P}^{\sigma'}_{\mathcal{M}}(\mathsf{mp}(r) = \mathsf{mp}(r)(\mathcal{B})) = \mathbb{P}^{\sigma'}_{\mathcal{M}}(\mathsf{F}\,\mathcal{B}) = 1$. Thus $\mathbb{P}^{\sigma'}_{\mathcal{M}}(\mathsf{mp}(r) < 0) = 1$, and hence $\mathbb{E}^{\sigma}_{\mathcal{M}}[\mathsf{mp}(r_i)] < 0$. We conclude that $\mathsf{Emp}(\vec{r})$ does not hold when $\mathsf{Pmp}(\vec{r})$ does not. □

Appendix C.6. Proof of Lemma 10

Proof. We first show that in the definition of $\vec{\mathsf{z}}_i^{\mathcal{E}} \stackrel{\text{def}}{=} \min_{t \in S_{\mathcal{E}}} \inf_{\sigma} \mathbb{E}^{\sigma}_{\mathcal{E},t}[\mathsf{mp}(r_i)]$, the minimum is reached for every state of the MEC.

Lemma 32. *Given a MEC $\mathcal{E}$, and an index i, the value $\inf_{\sigma} \mathbb{E}^{\sigma}_{\mathcal{E},t}[\mathsf{mp}(r_i)]$ does not depend on t, and is hence equal to $\vec{\mathsf{z}}_i^{\mathcal{E}}$.*

Proof. Consider two states t, t' of a MEC $\mathcal{E}$. Consider a strategy σ in the PA $\mathcal{E}_t$. Consider the strategy σ' in $\mathcal{E}_{t'}$ that first plays memoryless deterministic to reach t with probability 1 (it is possible in a MEC) and then switches to σ as soon as t is reached for the first time. Then $\mathbb{E}^{\sigma'}_{\mathcal{E},t'}[\mathsf{mp}(r_i)] = \mathbb{E}^{\sigma}_{\mathcal{E},t}[\mathsf{mp}(r_i)]$. Hence, for every t, t', $\inf_{\sigma'} \mathbb{E}^{\sigma'}_{\mathcal{E},t'}[\mathsf{mp}(r_i)] \leq \inf_{\sigma} \mathbb{E}^{\sigma}_{\mathcal{E},t}[\mathsf{mp}(r_i)]$. Reversing role of t, t' leads to an equality. □

We can now proceed to the proof of Lemma 10.

Let σ be an arbitrary Player □ strategy. Given a MEC $\mathcal{E} = (S_{\mathcal{E}}, \Delta_{\mathcal{E}})$, we denote by $\mathcal{E}^{(k)}$ the set of paths that stay forever in $\mathcal{E}$ after the first k steps, and define $\mathcal{E}^{(\infty)} = \cup_k \mathcal{E}^{(k)}$. We define the distributions $\gamma^k(\mathcal{E}) \stackrel{\text{def}}{=} \mathbb{P}^{\pi,\sigma}_{\mathcal{G}}(\mathcal{E}^{(k)})$ and $\gamma(\mathcal{E}) \stackrel{\text{def}}{=} \mathbb{P}^{\pi,\sigma}_{\mathcal{G}}(\mathcal{E}^{(\infty)})$. Note that $(\mathcal{E}^{(k)}_{\geq 0}$ is a non-decreasing sequence with respect to $\subseteq$, and hence $\gamma^k(\mathcal{E})$ is a non-decreasing sequence that converges towards $\gamma(\mathcal{E})$. By Theorem 3.2 of [23], with probability 1, the (player and stochastic) states seen infinitely often along a path form an end component, and hence are included in a MEC. Since MECs are disjoint, a further consequence is that $\sum_{\mathcal{E}} \gamma(\mathcal{E}) = 1$.

Now fix $\delta > 0$. Consider, for every state s that is in some MEC $\mathcal{E}$, and every $\delta > 0$, a δ-optimal strategy $\sigma_{s,\delta}$, that is, such that $\mathbb{E}^{\sigma_{s,\delta}}_{\mathcal{M},s}[\mathsf{mp}(\vec{r})] \leq \vec{\mathsf{z}}^{\mathcal{E}} + \delta$ (which exists due to Lemma 32). Consider the strategy $\sigma_{k,\delta}$ that plays as σ for the k first steps, and then switches to the δ-optimal strategy $\sigma_{s,\delta}$ if it is at a state s in some MEC, or plays arbitrarily if not in a MEC. Hence, it holds that

$$\vec{0} \leq \mathbb{E}^{\sigma_{k,\delta}}_{\mathcal{M}}[\mathsf{mp}(\vec{r})] \leq \sum_{\mathcal{E}} \sum_{s \in S_{\mathcal{E}}} \mathbb{P}^{\sigma}_{\mathcal{M}}(\mathsf{F}^{=k}\,\{s\}) \cdot \mathbb{E}^{\sigma_{s,\delta}}_{\mathcal{M},s}[\mathsf{mp}(\vec{r})] + (1 - \sum_{\mathcal{E}} \mathbb{P}^{\sigma}_{\mathcal{M}}(\mathsf{F}^{=k}\,S_{\mathcal{E}}))\rho^*],$$

where the second term is an upper bound on the reward contributed by the paths that are not in a MEC after k steps. We define $p_k(\mathcal{E}) \stackrel{\text{def}}{=} \mathbb{P}^{\sigma}_{\mathcal{M}}(\mathsf{F}^{=k}\,S_{\mathcal{E}}) = \sum_{s \in S_{\mathcal{E}}} \mathbb{P}^{\sigma}_{\mathcal{M}}(\mathsf{F}^{=k}\,\{s\})$, and have that

$$\vec{0} \leq \sum_{\mathcal{E}} p_k(\mathcal{E})(\vec{\mathsf{z}}^{\mathcal{E}} + \delta) + (1 - \sum_{\mathcal{E}} p_k(\mathcal{E}))\rho^*. \tag{C.1}$$

We now show that $p_k(\mathcal{E}) \to \gamma(\mathcal{E})$ for every $\mathcal{E}$. Indeed, it holds that

$$\gamma^k(\mathcal{E}) \leq p_k(\mathcal{E}) \leq 1 - \sum_{\mathcal{E}' \neq \mathcal{E}} p_k(\mathcal{E}') \leq 1 - \sum_{\mathcal{E}' \neq \mathcal{E}} \gamma^k(\mathcal{E}'),$$

and the outermost terms converge to the same limit $\gamma(\mathcal{E}) = 1 - \sum_{\mathcal{E}' \neq \mathcal{E}} \gamma(\mathcal{E}')$, and hence so does the inner term $p_k(\mathcal{E})$. Finally, we let $k \to +\infty$ and $\delta \to 0$ in (C.1) to obtain the desired result $\vec{0} \leq \sum_{\mathcal{E} \in \mathfrak{E}} \gamma(\mathcal{E}) \vec{\mathsf{z}}^{\mathcal{E}}$. □

Appendix C.7. Proof of Lemma 11

Proof. "Only if" direction. Assume $\mathcal{G}$ is CM. Fix a finite DU Player ◇ strategy π, and let $\mathcal{E} = (S_{\mathcal{E}}, \Delta_{\mathcal{E}})$ be a MEC of $\mathcal{G}^\pi$. It suffices to show that there exists an IC $\mathcal{H}$ such that $S_{\mathcal{H}} \subseteq S_{\mathcal{G},\mathcal{E}}$, since by the CM property $S_{\mathcal{H}}$ is reachable almost surely. We first build a Player □-closed subgame $\mathcal{H}'$ as follows. Define $\mathcal{H}' \stackrel{\text{def}}{=} \langle S_{\mathcal{G},\mathcal{E}}, (S_{\mathcal{G},\mathcal{E}} \cap S_\Diamond, S_{\mathcal{G},\mathcal{E}} \cap S_\Box, S_{\mathcal{G},\mathcal{E}} \cap S_\bigcirc), s'_{\mathsf{init}}, \mathcal{A}', \chi', \Delta' \rangle$, where $s'_{\mathsf{init}} \in S_{\mathcal{G},\mathcal{E}}$ is arbitrary, the move (a, μ) is in $S'_\bigcirc$ whenever there is $\mathfrak{m}$ such that $(a, \mu^\pi_{\mathfrak{m}})$ is in $\mathcal{E}$ (which also defines χ'), and Δ' is defined by $\Delta'(s, (a, \mu)) = \Delta_{\mathcal{E}}((s, \mathfrak{m}), a, \mu^\pi_{\mathfrak{m}})$ whenever $s \in S_\Box$ and $\mathfrak{m}$ as before, $\Delta'(s, (a, \mu)) = \Delta_{\mathcal{E}}((s, (a, \mu), \mathfrak{m}), a, \mu^\pi_{\mathfrak{m}})$ whenever $s \in S_\Diamond$ and $\mathfrak{m}$ as before. Hence, $s \xrightarrow{a} \mu$ in $\mathcal{G}$ whenever $s \xrightarrow{a}' \mu$ in $\mathcal{H}'$, and so we have $\Delta' \subseteq \Delta$. Further, since there is a finite path within $\mathcal{E}$ between each $s', t' \in S_{\mathcal{E}}$, there is also a finite path within $\mathcal{H}'$ between each $s, t \in S_{\mathcal{G},\mathcal{E}}$; hence, $\mathcal{H}'$ is a game. Finally, since for a MEC in $\mathcal{G}^\pi$ we require that $\mathsf{supp}(\mu^\pi_{\mathfrak{m}}) \subseteq S_{\mathcal{E}}$ whenever $\Delta_{\mathcal{E}}(s', a, \mu^\pi_{\mathfrak{m}}) > 0$, all successors of Player □ states $S_{\mathcal{G},\mathcal{E}} \cap S_\Box$ must be in $S_{\mathcal{G},\mathcal{E}}$. Thus, $\mathcal{H}'$ is Player □-closed. We now remove all but one choice per Player ◇ state in $\mathcal{H}'$, and obtain a subgame $\mathcal{H}''$ of $\mathcal{G}$, which corresponds to an MDP as Player ◇ has no longer any choice. Since we remove only Player ◇ choices, $\mathcal{H}''$ is still Player □ closed. A corollary of Lemma 2.2 of [12] is that every bottom strongly connected component (g-BSCC) in the graph of an MDP is a MEC. We can thus take a g-BSCC in the graph of $\mathcal{H}''$, which corresponds to a MEC, and thus an irreducible Player □-closed subgame $\mathcal{H}$ of $\mathcal{G}$.

"If" direction. Assume that, for every finite DU Player ◇ strategy π, for every MEC $\mathcal{E}$ of $\mathcal{G}^\pi$, $S_{\mathcal{G},\mathcal{E}}$ is almost surely reachable from every state of $\mathcal{G}$. Take any IC $\mathcal{H}$ in $\mathcal{G}$. Hence, for any π', $\mathcal{H}^{\pi'}$ forms a single MEC $\mathcal{E}$. Take the strategy π that plays arbitrarily outside of $\mathcal{H}$, and plays π' upon reaching $\mathcal{H}$. Then $\mathcal{E}$ is also a MEC in $\mathcal{G}^\pi$. By assumption, $S_{\mathcal{G},\mathcal{E}}$ is almost surely reachable from every state of $\mathcal{G}$. Since $S_{\mathcal{G},\mathcal{E}} = S_{\mathcal{H}}$, $S_{\mathcal{H}}$ is almost surely reachable from every state of $\mathcal{G}$, and hence $\mathcal{G}$ is CM. □

Appendix C.8. Proof of Lemma 12

Lemma 35 below ensures that we can safely interchange the quantification over σ, t, and N used to define $\vec{\mathsf{z}}^{\mathcal{E}}$. That means that, for every ε, there exists an N such that $\frac{\mathsf{rew}^{N-1}(\vec{r})}{N}$ stays above the threshold $\vec{\mathsf{z}}^{\mathcal{E}} - \varepsilon$, independently of the Player □ strategy and of the state considered as starting state.

We first show two following technical lemmas.

Lemma 33. *Let $\mathcal{D}$ be a DTMC, let $b \geq 0$, let $(c_K)_{K\in\mathbb{N}}$ be a sequence of positive reals, and let $(X_K)_{K\in\mathbb{N}}$, $(Y_K)_{K\in\mathbb{N}}$, $(Z_K)_{K\in\mathbb{N}}$ be sequences of real-valued random variables on $\Omega_{\mathcal{D}}$ such that $Z_K \geq 0$, $|X_K| \leq b \cdot c_K$, and $|Y_K| \leq b \cdot Z_K$. Then*

$$\left|\mathbb{E}_{\mathcal{D}}\left[\frac{X_K + Y_K}{c_K + Z_K}\right] - \mathbb{E}_{\mathcal{D}}\left[\frac{X_K}{c_K}\right]\right| \leq \frac{2b}{c_K}\mathbb{E}_{\mathcal{D}}[Z_K].$$

Proof. From the assumptions of the lemma, we obtain

$$\begin{aligned}
\left|\mathbb{E}_{\mathcal{D}}\left[\frac{X_K + Y_K}{c_K + Z_k}\right] - \mathbb{E}_{\mathcal{D}}\left[\frac{X_K}{c_K}\right]\right| &= \left|\mathbb{E}_{\mathcal{D}}\left[\frac{Y_K}{c_K + Z_K}\right] - \mathbb{E}_{\mathcal{D}}\left[\frac{X_K \cdot Z_K}{c_K(c_K + Z_K)}\right]\right| \\
&\leq \mathbb{E}_{\mathcal{D}}\left[\frac{|Y_K|}{c_K + Z_K}\right] + \mathbb{E}_{\mathcal{D}}\left[\frac{|X_K| \cdot Z_K}{c_K(c_K + Z_K)}\right] \\
&\leq \mathbb{E}_{\mathcal{D}}\left[\frac{b \cdot Z_K}{c_K}\right] + \mathbb{E}_{\mathcal{D}}\left[\frac{b \cdot c_K \cdot Z_K}{c_K^2}\right] \\
&\leq \frac{2b}{c_K}\mathbb{E}_{\mathcal{D}}[Z_K].
\end{aligned}$$

□

Lemma 34. *Let $\mathcal{G}$ be a game with states S and with minimum non-zero probability $p_{\min}$. For any $s, t \in S$ such that t is reachable from s almost surely, the expected number of steps to reach t from s with an MD strategy is bounded from above by $|S| \cdot p_{\min}^{-|S|}$.*

Proof. After $|S|$ steps, t is reached from s with probability at least $p^* \stackrel{\text{def}}{=} p_{\min}^{|S|}$, Thus, the expected number of steps to reach $S_{\mathcal{H}}$ from s is upper bounded by $N_{\text{trans}} \stackrel{\text{def}}{=} |S|p^* + 2|S|p^*(1-p^*) + 3|S|p^*(1-p^*)^2 + \cdots = |S|/p^*$. □

Lemma 35. *For every MEC $\mathcal{E}$ of a finite PA with rewards $\vec{r}$, it holds that*

$$\underline{\lim}_{N\to\infty} \min_{t\in S_{\mathcal{E}}} \inf_{\sigma} \mathbb{E}^{\sigma}_{\mathcal{E},t}\left[\frac{\mathsf{rew}^{N-1}(\vec{r})}{N}\right] \geq \vec{\mathsf{z}}^{\mathcal{E}}.$$

Proof. Fix a MEC $\mathcal{E} = (S_{\mathcal{E}}, \Delta_{\mathcal{E}})$ of a finite PA $\mathcal{M} = \langle S, (S_{\Box}, S_{\bigcirc}), \varsigma, \mathcal{A}, \chi, \Delta\rangle$. Denote by $p_{\min}$ the minimum non-zero probability in $\mathcal{M}$, and let $\rho^* \stackrel{\text{def}}{=} \max_{s\in S,i} |r_i(s)|$. Assume toward a contradiction that there exists $\delta > 0$ and i such that

$$\underline{\lim}_{N\to\infty} \min_{t\in S_{\mathcal{E}}} \inf_{\sigma} \mathbb{E}^{\sigma}_{\mathcal{E},t}\left[\frac{\mathsf{rew}^{N-1}(r_i)}{N}\right] < \mathsf{z}_i^{\mathcal{E}} - \delta.$$

In particular, we can fix $N \geq \lfloor 2\rho^*|S_{\mathcal{E}}|p_{\min}^{-|S_{\mathcal{E}}|}\delta^{-1}\rfloor$, $t \in S_{\mathcal{E}}$, and σ, such that

$$\mathbb{E}^{\sigma}_{\mathcal{E},t}\left[\frac{\mathsf{rew}^{N-1}(r)}{N}\right] < \mathsf{z}_i^{\mathcal{E}} - \delta.$$

We show that there exists a strategy σ' such that $\mathbb{E}^{\sigma'}_{\mathcal{E},t}[\mathsf{mp}(\vec{r}_i)] < \mathsf{z}_i^{\mathcal{E}}$, that is, it contradicts the definition of $\mathsf{z}_i^{\mathcal{E}}$. From Lemma 34, we have that $|S_{\mathcal{E}}| \cdot p_{\min}^{-|S_{\mathcal{E}}|}$ is an upper bound for the expected number of steps to reach t from s for MD strategies. We construct the strategy σ' as follows. Starting from t, σ' plays in the first phase the first N steps of σ, then plays in the second phase an MD strategy to reach t, and then repeats ad infinitum the two previous phases. For a path λ, we let $N^{(K)}(\lambda)$ be the index of the beginning of the Kth loop, and $+\infty$ if λ contains no loops. We have

$$\begin{aligned}
\mathbb{E}^{\sigma'}_{\mathcal{E},t}[\mathsf{mp}(r_i)] &= \mathbb{E}^{\sigma'}_{\mathcal{E},t}\left[\underline{\lim}_{k\to\infty}\frac{1}{k+1}\mathsf{rew}^k(r_i)\right] && \text{(definition)}\\
&\leq \mathbb{E}^{\sigma'}_{\mathcal{E},t}\left[\underline{\lim}_{K\to\infty}\frac{1}{N^{(K)}+1}\mathsf{rew}^{N^{(K)}}(r_i)\right] && \text{(sub-sequence)}\\
&\leq \underline{\lim}_{K\to\infty}\mathbb{E}^{\sigma'}_{\mathcal{E},t}\left[\frac{1}{N^{(K)}+1}\mathsf{rew}^{N^{(K)}}(r_i)\right] && \text{(Fatou's Lemma)}
\end{aligned}$$

For a path λ, we denote by $c_K(\lambda) - 1 \stackrel{\text{def}}{=} NK$ (resp. $Z_K(\lambda)$) the total cumulated steps in the first phase (resp. second phase) during the first K loops. We denote by $X_K(\lambda)$ (resp. $Y_K(\lambda)$) the respective cumulated reward of r_i. We have $\mathbb{E}^{\sigma'}_{\mathcal{E},t}\left[\frac{1}{N^{(K)}+1}\mathsf{rew}^{N^{(K)}}(r_i)\right] \stackrel{\text{def}}{=} \mathbb{E}^{\sigma'}_{\mathcal{E},t}\left[\frac{X_K+Y_K}{c_K+Z_K}\right]$, and so from Lemma 33 we obtain

$$\mathbb{E}^{\sigma'}_{\mathcal{E},t}\left[\frac{1}{N^{(K)}+1}\mathsf{rew}^{N^{(K)}}(r_i)\right] \leq \mathbb{E}^{\sigma'}_{\mathcal{E},t}\left[\frac{X_K}{c_K}\right] + \frac{2\rho^*}{c_K}\mathbb{E}^{\sigma'}_{\mathcal{E},t}[Z_K]. \tag{C.2}$$

We now consider the two terms on the right-hand side of (C.2). By definition of σ' in the first phase, the first term equals $\frac{K}{1+KN}\mathbb{E}^{\sigma}_{\mathcal{E},t}\left[\mathsf{rew}^{N-1}(r_i)\right]$. The second term is upper-bounded by δ, since $(2\rho^*/c_K)\mathbb{E}^{\sigma'}_{\mathcal{E},t}[Z_K] \leq (2\rho^*/KN)K|S_{\mathcal{E}}|p_{\min}^{-|S_{\mathcal{E}}|} = 2\rho^*|S_{\mathcal{E}}|p_{\min}^{-|S_{\mathcal{E}}|}/N \leq \delta$. We can now conclude

$$\mathbb{E}^{\sigma'}_{\mathcal{E},t}[\mathsf{mp}(r_i)] \leq \underline{\lim}_{K\to\infty}\frac{K}{1+KN}\mathbb{E}^{\sigma}_{\mathcal{E},t}[\mathsf{rew}^{N-1}] + \delta = \frac{1}{N}\mathbb{E}^{\sigma}_{\mathcal{E},t}[\mathsf{rew}^{N-1}] + \delta < u_i^{\mathcal{E}}.$$

This contradicts the definition of $u_i^{\mathcal{E}}$ and the proof is complete. □

We can now prove Lemma 12.

Proof. Let $\mathfrak{E}$ be the set of L MECs $\mathcal{E}_l$ of $\mathcal{G}^{\pi^l}$, indexed by l. We show that the strategy π constructed in Definition 7, with appropriately chosen step counts N_l, satisfies the lemma, that is, it approximates γ. Throughout the proof, we refer to the strategy π, keeping the step counts as parameters. From Lemma 11, every MEC is almost surely reachable in $\mathcal{G}$ from any state s. Thus, we have an upper bound $N_{\triangleright} = |S| \cdot p^*$ on the mean time spent between two MECs. For every l, we define A_l such that, for every $N_l \geq A_l$, $\min_{t\in S_{\mathcal{E}_l}} \inf_\sigma \mathbb{E}^{\sigma}_{\mathcal{E}_l,t}\left[\mathsf{rew}^{N_l-1}(\vec{r})\right] \geq N_l(\vec{\mathsf{z}}^{\mathcal{E}_l} - \varepsilon/3)$, which exists by virtue of Lemma 35. We now define the step counts for π by $N_l \stackrel{\text{def}}{=} \lfloor h\gamma(\mathcal{E}_l)\rfloor$, and let $N \stackrel{\text{def}}{=} \sum_{l=1}^{L} N_l$ with h chosen such that

(h1) for every l, $N_l \geq A_l$;

(h2) $1/h \leq \varepsilon/(3\sum_{l=1}^{L}\|\vec{\mathsf{z}}^{\mathcal{E}_l}\|_\infty)$;

(h3) $(L\gamma(\mathcal{E}_l)+1)/(h-L) \leq \varepsilon/(3\sum_l \|\vec{\mathsf{z}}^{\mathcal{E}_l}\|_\infty)$; and

(h4) $\frac{1}{N}2\rho^* L N_\triangleright \leq \varepsilon/3$.

For an infinite path λ, we let $N^{(K)}(\lambda)$ be the index of the beginning of the Kth loop, or $+\infty$ if λ has fewer than K loops. For every finite DU strategy σ, it holds for almost every path λ that $N^{(K)}(\lambda)$ is finite for all K, and thus $\lim_{k\to\infty}\frac{1}{k+1}\mathsf{rew}^k(\vec{r})(\lambda) = \lim_{K\to\infty}\frac{1}{N^{(K)}+1}\mathsf{rew}^{N^{(K)}}(\vec{r})(\lambda)$. Hence,

$$\begin{aligned}
\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\mathsf{mp}(\vec{r})] &= \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left[\lim_{k\to\infty}\frac{1}{k+1}\mathsf{rew}^k(\vec{r})\right] && \text{(definition)}\\
&= \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left[\lim_{K\to\infty}\frac{1}{N^{(K)}+1}\mathsf{rew}^{N^{(K)}}(\vec{r})\right] && \text{(almost sure equality)}\\
&= \lim_{K\to\infty}\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left[\frac{1}{N^{(K)}+1}\mathsf{rew}^{N^{(K)}}(\vec{r})\right]. && \text{(Lebesgue's theorem)}
\end{aligned}$$

For a path $\lambda \in \Omega_{\mathcal{D}}$, we denote by $c_K - 1 \stackrel{\text{def}}{=} NK$ (resp. $Z_k(\lambda)$), the total cumulated time spent on the MEC phase (resp. inter-MEC phase) during the first K loops. We denote by $X_K(\lambda)$ (resp. $Y_K(\lambda)$) the respective cumulated reward. We are interested in the limit when $K\to\infty$ of

$$\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left[\frac{1}{N^{(K)}+1}\mathsf{rew}^{N^{(K)}}\right] = \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left(\frac{X_K+Y_K}{c_K+Z_K}\right),$$

and from Lemma 33 we therefore get that

$$\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left[\frac{1}{N^{(K)}+1}\mathsf{rew}^{N^{(K)}}\right] \geq \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left(\frac{X_K}{c_K}\right) - \frac{2\rho^*}{c_K}\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}(Z_K). \tag{C.3}$$

We let $X_{l,k}(\lambda)$ be the reward accumulated in the lth MEC phase during the kth loop, and thus have $X_K = \sum_{k=0}^{K-1}\sum_{l=1}^{L} X_{l,k}$. By virtue of (h1), $N_l \geq A_l$, and hence it holds that $\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[X_{l,k}] \geq N_l(\vec{\mathsf{z}}^{\mathcal{E}_l} - \frac{1}{3}\varepsilon)$. Therefore,

$$\begin{aligned}
\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left(\frac{X_K}{c_K}\right) &= \frac{1}{1+KN}\sum_{k=0}^{K-1}\sum_{l=1}^{L}\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[X_{l,k}]\\
&\geq \frac{1}{1+KN}\sum_{k=0}^{K-1}\sum_{l=1}^{L}N_l(\vec{\mathsf{z}}^{\mathcal{E}_l} - \frac{1}{3}\varepsilon)\\
&\geq \frac{K}{1+KN}\sum_{l=1}^{L}N_l\vec{\mathsf{z}}^{\mathcal{E}_l} - \frac{1}{3}\varepsilon.
\end{aligned}$$

Taking the limit, we get

$$\begin{aligned}
\lim_{K\to\infty}\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left(\frac{X_K}{c_K}\right) &\geq \sum_{l=1}^{L}\frac{N_l}{N}\vec{\mathsf{z}}^{\mathcal{E}_l} - \frac{1}{3}\varepsilon\\
&\geq \sum_{l=1}^{L}\gamma(\mathcal{E}_l)\vec{\mathsf{z}}^{\mathcal{E}_l} - \sum_{l=1}^{L}\left|\gamma(\mathcal{E}_l) - \frac{N_l}{N}\right|\|\vec{\mathsf{z}}^{\mathcal{E}_l}\|_\infty - \frac{1}{3}\varepsilon
\end{aligned}$$

Note that

$$\frac{N_l}{N} \geq \frac{h\gamma(\mathcal{E}_l) - 1}{\sum_{l'=1}^{L} h\gamma(\mathcal{E}_{l'})} \geq \gamma(\mathcal{E}_l) - \frac{1}{h},$$

and that

$$\frac{N_l}{N} \leq \frac{h\gamma(\mathcal{E}_l) + 1}{\sum_{l'=1}^{L}(h\gamma(\mathcal{E}_{l'}) - 1)} = \frac{h\gamma(\mathcal{E}_l) + 1}{h - L} = \gamma(\mathcal{E}_i) + \frac{1}{h-L}(L\gamma(\mathcal{E}_l) + 1).$$

Using condition (h2) and (h3) on h, we get $\left|\gamma(\mathcal{E}_l) - \frac{N_l}{N}\right| \leq \varepsilon/(3\sum_{l'=1}^{L} \|\vec{u}^{\mathcal{E}_{l'}}\|)$, and hence

$$\lim_{K\to\infty} \mathbb{E}_{\mathcal{G}}^{\pi,\sigma}\left(\frac{X_K}{c_K}\right) \geq \sum_{l=1}^{L} \gamma(\mathcal{E}_l)\vec{\mathsf{z}}^{\mathcal{E}_l} - \frac{2}{3}\varepsilon. \tag{C.4}$$

We now upper-bound the absolute value of the second term of (C.3) using

$$\frac{2\rho^*}{c_K}\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}(Z_K) \leq \frac{2\rho^*}{KN}KLN_{\triangleright} = \frac{1}{N}2\rho^* LN_{\triangleright} \leq \varepsilon/3, \tag{C.5}$$

where the last inequality comes from condition (h4) on h, and hence on N. Applying the bounds (C.4) and (C.5) to (C.3), we obtain

$$\mathbb{E}_{\mathcal{G}}^{\pi,\sigma}[\mathsf{mp}(\vec{r})] \geq \sum_{l=1}^{L} \gamma(\mathcal{E}_l)\vec{\mathsf{z}}^{\mathcal{E}_l} - \varepsilon. \qquad \square$$

Appendix D. Proofs of results of Section 5

Appendix D.1. Proof of Lemma 13

Proof. Let $\mathcal{M} = \langle S, (S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta\rangle$, $\mathcal{M}' = \langle S', (S'_\Box, S'_\bigcirc), \varsigma', \mathcal{A}', \chi', \Delta'\rangle$, and $\sigma = \langle \mathfrak{M}, \sigma_{\mathsf{c}}, \sigma_{\mathsf{u}}, \sigma_{\mathsf{d}}\rangle$. We construct an SU strategy σ' that simulates σ applied to $\mathcal{M}$ by keeping the current state in $\mathcal{M}$ and the memory of σ in its own memory. The functional simulation ensures that every path of $\mathcal{M}^\sigma$ corresponds to a path in $(\mathcal{M}')^{\sigma'}$, and so after seeing memory $(s, \mathfrak{m})$ the strategy σ' picks the next move that σ would pick in state s with memory $\mathfrak{m}$. Our aim is to show that the trace distributions of $(\mathcal{M}')^{\sigma'}$ and $\mathcal{M}^\sigma$ are equivalent. We formally let $\sigma' \stackrel{\text{def}}{=} \langle \mathfrak{M}', \sigma'_{\mathsf{c}}, \sigma'_{\mathsf{u}}, \sigma'_{\mathsf{d}}\rangle$, where we define $\mathfrak{M}' \stackrel{\text{def}}{=} \mathfrak{M} \times S$, and where, for all $(\mathfrak{m}, s), (\mathfrak{n}, (a, \mu)), (\mathfrak{o}, t) \in \mathfrak{M}'$ and all $s' \xrightarrow{a}{}' \mu'$ in $\mathcal{M}'$, such that $s' = \mathcal{F}(s)$, $\mu' = \overline{\mathcal{F}}(\mu)$, $t' = \mathcal{F}(t) \in \mathsf{supp}(\mu')$, we define

$$\sigma'_{\mathsf{d}}(s')((\mathfrak{m}, s)) \stackrel{\text{def}}{=} \sigma_{\mathsf{d}}(s)(\mathfrak{m}) \cdot \frac{\varsigma(s)}{\varsigma'(s')}$$

$$\sigma'_{\mathsf{u}}((\mathfrak{m}, s), (a, \mu'))((\mathfrak{n}, (a, \mu))) \stackrel{\text{def}}{=} \frac{\sigma_{\mathsf{u}}(\mathfrak{m}, (a, \mu))(\mathfrak{n})}{\sigma'_{\mathsf{c}}(s', (\mathfrak{m}, s))(a, \mu')} \tag{D.1}$$

$$\sigma'_{\mathsf{u}}((\mathfrak{n}, (a, \mu)), t')((\mathfrak{o}, t)) \stackrel{\text{def}}{=} \sigma_{\mathsf{u}}(\mathfrak{n}, t)(\mathfrak{o}) \cdot \frac{\mu(t)}{\mu'(t')} \tag{D.2}$$

$$\sigma'_{\mathsf{c}}(s', (\mathfrak{m}, s))(a, \mu') \stackrel{\text{def}}{=} \textstyle\sum_{\overline{\mathcal{F}}(\mu) = \mu'} \sigma_{\mathsf{c}}(s, \mathfrak{m})(a, \mu).$$

Denote by $\mathbb{P}_{\mathcal{D}}(\mathfrak{m}, \lambda) \stackrel{\text{def}}{=} \mathbb{P}_{\mathcal{D}}(\lambda) \cdot \mathfrak{d}_{\lambda}(\mathfrak{m})$ the probability of the path λ and the memory $\mathfrak{m}$ after seeing λ. A functional simulation $\mathcal{F}$ must be defined for the reachable states of $\mathcal{M}$, and so it extends inductively to a total function on paths of $\mathcal{M}$ by defining $\mathcal{F}(\lambda(a,\mu)s) \stackrel{\text{def}}{=} \mathcal{F}(\lambda)(a, \overline{\mathcal{F}}(\mu))\mathcal{F}(s)$. We now show by induction on the length of paths that $\mathbb{P}^{\sigma'}_{\mathcal{M}'}((\mathfrak{m}, s), \lambda') = \mathbb{P}^{\sigma}_{\mathcal{M}}(\mathfrak{m}, \lambda)$ if $\mathcal{F}(\lambda) = \lambda'$, and $\mathbb{P}^{\sigma'}_{\mathcal{M}'}((\mathfrak{m}, s), \lambda') = 0$ otherwise.

For the base case, for any $(\mathfrak{m}, s) \in \mathfrak{N}'$ and $s' \in S'$ such that $s' = \mathcal{F}(s)$, we have that $\mathbb{P}^{\sigma'}_{\mathcal{M}'}((\mathfrak{m}, s), s') = \varsigma'(s') \cdot \sigma'_{\mathsf{d}}(s', (\mathfrak{m}, s)) = \sigma_{\mathsf{d}}(s)(\mathfrak{m}) \cdot \varsigma(s) = \mathbb{P}^{\sigma}_{\mathcal{M}}(\mathfrak{m}, s)$; if, on the other hand, $s' \neq \mathcal{F}(s)$ then $\sigma'_{\mathsf{d}}(s', (\mathfrak{m}, s)) = 0$, and so $\mathbb{P}^{\sigma'}_{\mathcal{M}'}((\mathfrak{m}, s), s') = 0$.

For the induction step, assume the equality holds for $\lambda \in \Omega^{\mathsf{fin}}_{\mathcal{M}}$ and $\lambda' \in \Omega^{\mathsf{fin}}_{\mathcal{M}'}$, and we consider paths $\lambda(a,\mu)t \in \Omega^{\mathsf{fin}}_{\mathcal{M}}$ and $\lambda'(a,\mu')t' \in \Omega^{\mathsf{fin}}_{\mathcal{M}'}$. We have that

$$\mathbb{P}^{\sigma'}_{\mathcal{M}'}((\mathfrak{o}, t), \lambda'(a,\mu')t') = \sum_{(\mathfrak{m}, \mathsf{last}(\lambda)), (\mathfrak{n}, (a,\mu)) \in \mathfrak{N}'} \mathbb{P}^{\sigma'}_{\mathcal{M}'}((\mathfrak{m}, s), \lambda') \cdot p_1 \cdot p_2,$$

where

$$p_1 = \sigma'_{\mathsf{c}}(\mathsf{last}(\lambda'), (\mathfrak{m}, \mathsf{last}(\lambda)))(a,\mu') \cdot \sigma'_{\mathsf{u}}((\mathfrak{m}, \mathsf{last}(\lambda)), (a,\mu'))((\mathfrak{n}, (a,\mu)))$$
$$p_2 = \mu'(t') \cdot \sigma'_{\mathsf{u}}((\mathfrak{n}, (a,\mu)), t')((\mathfrak{o}, t')).$$

We consider first the case where $\mathcal{F}(\lambda(a,\mu)t) \neq \lambda'(a,\mu')t'$: if $\mathcal{F}(\lambda) \neq \lambda'$, then from the induction hypothesis $\mathbb{P}^{\sigma'}_{\mathcal{M}'}((\mathfrak{m}, s), \lambda') = 0$; and if $\mathcal{F}((a,\mu)t) \neq (a,\mu')t'$, then $p_2 = 0$ from (D.2). Now suppose that $\mathcal{F}(\lambda(a,\mu)t) = \lambda'(a,\mu')t'$. From (D.1) we have that $p_1 = \sigma_{\mathsf{u}}(\mathfrak{m}, (a,\mu))(\mathfrak{n})$ and from (D.2) we have that $p_2 = \mu(t) \cdot \sigma_{\mathsf{u}}(\mathfrak{n}, t)(\mathfrak{o})$. Applying the induction hypothesis, we conclude the induction, since

$$\begin{aligned}\mathbb{P}^{\sigma'}_{\mathcal{M}'}((\mathfrak{o}, t), \lambda'(a,\mu')t') &= \sum_{\mathfrak{m}, \mathfrak{n} \in \mathfrak{N}} \mathbb{P}^{\sigma}_{\mathcal{M}}(\mathfrak{m}, \lambda) \cdot \sigma_{\mathsf{u}}(\mathfrak{m}, (a,\mu))(\mathfrak{n}) \cdot \mu(t) \cdot \sigma_{\mathsf{u}}(\mathfrak{n}, t)(\mathfrak{o}) \\ &= \mathbb{P}^{\sigma}_{\mathcal{M}}(\mathfrak{o}, \lambda(a,\mu)t).\end{aligned}$$

We thus have

$$\tilde{\mathbb{P}}^{\sigma'}_{\mathcal{M}'}(w) = \sum_{\substack{\lambda' \in \mathsf{paths}(w) \\ (\mathfrak{m}, s) \in \mathfrak{N}'}} \mathbb{P}^{\sigma'}_{\mathcal{M}'}((\mathfrak{m}, s), \lambda') = \sum_{\substack{\lambda' \in \mathsf{paths}(w) \\ \mathcal{F}(\lambda) = \lambda' \\ \mathfrak{m} \in \mathfrak{N}}} \mathbb{P}^{\sigma}_{\mathcal{M}}(\mathfrak{m}, \lambda) \stackrel{*}{=} \sum_{\lambda \in \mathsf{paths}(w)} \mathbb{P}^{\sigma}_{\mathcal{M}}(\lambda) \stackrel{\text{def}}{=} \tilde{\mathbb{P}}^{\sigma}_{\mathcal{M}}(w).$$

where the equation marked with $*$ is a consequence of $\mathsf{trace}(\lambda) = \mathsf{trace}(\mathcal{F}(\lambda))$. Thus, σ' and σ induce the same trace distribution, and φ, which is defined on traces, satisfies $(\mathcal{M}')^{\sigma'} \models \varphi \Leftrightarrow \mathcal{M}^{\sigma} \models \varphi$. □

Appendix D.2. Proof of Lemma 14

Proof. Following Remark 3, we assume w.l.o.g. that the strategies are DU strategies. We construct a functional simulation by viewing states in the induced PA $\mathcal{M} = (\|_{i \in I}\ \mathcal{G}^i)^{\|_{i \in I} \pi^i}$ as derived from the paths of the composed game $\mathcal{G} = (\|_{i \in I}\ \mathcal{G}^i)$. These paths are projected to components $\mathcal{G}^i$ and then assigned a corresponding state in the induced PA $(\mathcal{G}^i)^{\pi^i}$. Due to the structure imposed by compatibility, moves chosen at Player $\Diamond$ states in $\mathcal{G}^i$ can be translated to moves in the composition $\mathcal{M}' = \|_{i\in I} (\mathcal{G}^i)^{\pi^i}$.

Denote the induced PA by $\mathcal{M} = \langle S, (S_\Box, S_\bigcirc), \varsigma, \mathcal{A}, \chi, \Delta\rangle$, and the composition of induced PAs by $\mathcal{M}' = \langle S', (S'_\Box, S'_\bigcirc), \varsigma', \mathcal{A}, \chi', \Delta'\rangle$. We define a partial function $\mathcal{F} : S \to S'$, and then show that it is a functional simulation. We use $\vec{\gamma}$ to stand for both Player $\Box$ states $\vec{s}$ and Player $\Diamond$ state-move tuples $(\vec{s}, (a, \vec{\mu}))$ of the game $\mathcal{G}$, as occurring in the induced PA $\mathcal{M}$ (see Definition 6). We write

$$[\vec{\gamma}]^i = \begin{cases} s^i & \text{if } \vec{\gamma} = \vec{s} \in S_\Box \\ (s^i, (a, \mu^i)) & \text{if } \vec{\gamma} = (\vec{s}, (a, \vec{\mu})) \text{ and } \mathcal{G}^i \text{ is involved in } \vec{s} \xrightarrow{a} \vec{\mu} \\ s^i & \text{if } \vec{\gamma} = (\vec{s}, (a, \vec{\mu})) \text{ and } \mathcal{G}^i \text{ is not involved in } \vec{s} \xrightarrow{a} \vec{\mu}, \end{cases}$$

We define $\mathcal{F}$ by $[\mathcal{F}(\vec{\gamma}, \vec{\mathfrak{d}})]_i = (\gamma^i, \mathfrak{d}^i)$ for all reachable states $(\vec{\gamma}, \vec{\mathfrak{d}}) \in S$ of $\mathcal{M}$, and all $i \in I$. We now show that $\mathcal{F}$ is a functional simulation.

Case (F1). We show that $\overline{\mathcal{F}}(\varsigma) = \varsigma'$. Note that, due to the normal form, the initial distribution ς of $\mathcal{M}$ only maps to states of the form $S_\Box \times \mathfrak{M}$, and the initial distribution ς' of $\mathcal{M}'$ only maps to states of the form $\prod_{i\in I} S^i_\Box \times \mathfrak{M}^i$. For such states $(\vec{s}, \vec{\mathfrak{d}}) \in S_\Box \times \mathfrak{M}$, we have $[\mathcal{F}(\vec{s}, \vec{\mathfrak{d}})]_i = (s^i, \mathfrak{d}^i)$, and so $\overline{\mathcal{F}}(\varsigma)((s^1, \mathfrak{d}^1), (s^2, \mathfrak{d}^2), \ldots) = \varsigma(\vec{s}, \vec{\mathfrak{d}}) = \varsigma'((s^1, \mathfrak{d}^1), (s^2, \mathfrak{d}^2), \ldots)$.

Case (F2). Consider a transition $(\vec{\gamma}, \vec{\mathfrak{d}}) \xrightarrow{a} \mu_{\vec{\gamma},\vec{\mathfrak{d}}}$ of the induced PA, $\mathcal{M} = \mathcal{G}^{\|_{i\in I}\pi^i}$ where $\mu_{\vec{\gamma},\vec{\mathfrak{d}}}(\vec{\gamma}', \vec{\mathfrak{d}}') \stackrel{\text{def}}{=} \Delta^\pi((\vec{\gamma}, \vec{\mathfrak{d}}), (\vec{\gamma}', \vec{\mathfrak{d}}'))$. It is induced from a transition $\vec{s} \xrightarrow{a} \vec{\mu}$ of the game composition $\mathcal{G}$. For each involved component $\mathcal{G}^i$, we apply the strategy π^i separately, and obtain that, for each transition $s^i \xrightarrow{a}{}^i \mu^i$ in $\mathcal{G}^i$, the transition $(\gamma^i, \mathfrak{d}^i) \xrightarrow{a} \mu_{\gamma^i,\mathfrak{d}^i}$ (where $\mu_{\gamma^i,\mathfrak{d}^i}(\gamma', \mathfrak{d}') = \Delta^{\pi^i}((\gamma^i, \mathfrak{d}^i), (\gamma, \mathfrak{d}')))$ is in the induced PA $(\mathcal{G}^i)^{\pi^i}$. Then, composing the induced PAs $(\mathcal{G}^i)^{\pi^i}$ yields a transition $\mathcal{F}(\vec{\gamma}, \vec{\mathfrak{d}}) \xrightarrow{a}{}' \nu$ in $\mathcal{M}'$, where ν is not null on element $\mathcal{F}(\gamma_+, \mathfrak{d}_+)$ only if $\gamma^i = \gamma^i_+$ and $\mathfrak{d}^i = \mathfrak{d}^i_+$ for the component not involved and $\mathfrak{d}^i_+ = \pi^i_{\mathsf{u}}(\mathfrak{d}^i, (a, \mu^i))$ for the involved component. On such elements it holds that

$$\begin{aligned} \nu(\mathcal{F}(\vec{\gamma_+}, \vec{\mathfrak{d}_+})) &= \prod_{i\in\Gamma(\vec{\mathfrak{d}},\vec{\gamma_+})} \mu^{\pi^i}_{\gamma^i,\mathfrak{d}^i}(\gamma^i_+, \mathfrak{d}^i_+) \qquad &&\text{(Definition 9)} \\ &= \overline{\mathcal{F}}(\mu_{\vec{\gamma},\vec{\mathfrak{d}}})(\mathcal{F}(\vec{\gamma_+}, \vec{\mathfrak{d}_+})). &&\text{(definition of } \overline{\mathcal{F}}) \end{aligned}$$

We thus have that $\mathcal{F}(\vec{\gamma}, \vec{\mathfrak{d}}) \xrightarrow{a}{}' \overline{\mathcal{F}}(\mu_{\vec{\gamma},\vec{\mathfrak{d}}})$ is in $\mathcal{M}'$, concluding the proof of (F2). $\Box$