Fair adversaries and randomization in two-player games

Eugene Asarin[1], Raphaël Chane-Yack-Fa[2*], and Daniele Varacca[3]

[1] LIAFA - CNRS & Univ. Paris Diderot, France
[2] Univ. de Sherbrooke - Département d'informatique - Quebec, Canada
[3] PPS - CNRS & Univ. Paris Diderot, France

Abstract. two-player games are used to model open systems. One player models the system, trying to respect some specification, while the other player models the environment. In classical model checking, the objective is to verify that the system can respect its specification, whatever the environment does.

In this article, we consider a more realistic scenario when the environment is supposed to be fair. We define a notion of fair player in two-player games. Our solution is inspired by Banach-Mazur games, and leads to a definition of a novel class of 3-player games called ABM-games. For ω-regular specifications on finite arenas, we explore the properties of ABM-games and devise an algorithm for solving them. As the main result, we show that winning in an ABM-game (i.e. winning against a fair player) is equivalent to winning with probability one against the randomized adversary.

Key words: Games, Markov decision processes, fairness.

1 Introduction

Two-player games are used to model open systems. One player (sometimes called Adam) models the system, trying to achieve some goal, while the other player (sometimes called Eve) models the environment. In classical model checking, one wants to verify that the system can achieve the goal in *any* kind of environment. Therefore, one can assume a very evil Eve, able to exploit the smallest weakness of Adam. In many situations, the environment is not that evil. It can make choices against the system, but sometimes it can play in its favour. There are different ways to model such an environment. One possible way is to suppose that the environment makes random choices. This leads to the notion of $1\,{}^{1}\!/_{2}$-player games, or Markov decision processes (MDP). In such models, one wishes to verify whether the system can reach its goal with probability 1.

Another point of view is to suppose that the environment is *fair*. Fairness assumptions are well known in the context of closed systems. In most cases [2, 14], fairness assumptions are of the form "if an action is allowed infinitely often during a run, then it is done infinitely often". But what does it mean for a *player*

* During this research the second author was with LIAFA and PPS.

to be fair? A general definition of fairness for closed systems has been proposed in [15]. It is based on a different notion of game: the Banach-Mazur game. A property is defined to be a fairness property if the good player has a winning strategy in the game. This definition has a topological characterisation in terms of comeager sets. In [14], a comparison between the general notion of fairness and Markov chains is performed. It is shown that for ω-regular specifications, a finite Markov chain satisfies the specification with probability 1, if and only if the specification is a fairness property in the sense of [15].

In this work, we propose a similar definition of fair player. The idea is to split Eve into two "sub-players", Banach (good) and Mazur (evil), playing the Banach-Mazur game between themselves. Eve is thus not always playing against Adam, but sometimes she shares his goals. However Adam does not know when this is the case, he only knows that Eve is split. To show that our notion of 2-player game with fair Eve is correct, we compare it with Markov Decision Processes. Similarly to the result of [14], we show that for games with ω-regular conditions, Adam can reach his goal with probability 1 in an MDP if and only if he can win the game against fair Eve.

The characterisation in terms of Banach-Mazur games gives a different point of view of qualitative probabilistic model checking. In [13], this point of view is used to simplify and modify the classic algorithm by Courcoubetis and Yannakakis [7]. Also it helps providing a notion of counterexample for probabilistic model checking [12]. We hope that our proposal can be the starting point for similar results in the model checking of MDPs.

Structure of the paper: In Section 2, we introduce the known notions of two-player game, Markov decision process, Banach-Mazur game, etc. We present the theorem that links Banach-Mazur games on graphs with Markov chains. In Section 3, we introduce our new game, the ABM game. We show that for parity winning condition, if player Adam wins, then he has a memoryless winning strategy, and for ω-regular condition, he has a finite memory strategy. This is the main technical result of the paper, and we present the complete proof. This proof is indeed constructive and it generates an algorithm to decide whether Adam has a winning strategy, and to produce the winning strategy when it exists. We also show that, contrary to two-player games, the ABM game is not determined, by showing a game where no player has a winning strategy. Finally, in Section 4, we show that, for parity and ω-regular goals, Adam wins the ABM game, if and only if he almost surely wins in the corresponding Markov decision process. This result uses the existence of memoryless and finite memory strategies.

2 Infinite games on finite graphs

2.1 Preliminaries

A *(directed) graph* is a pair $G = (V, T)$ where V is a set of *vertices* (also called states) and T a set of *edges* (or transitions) such that $T \subseteq V \times V$. In this article, we assume that the graphs are finite, i.e. V is always a finite set. We also consider only graphs without dead-ends, i.e. for all $v \in V$ there exists a vertex $w \in V$ such that $(v, w) \in T$. We assume the reader is familiar with the notion of *strongly connected component.* A *bottom strongly connected component* U is a strongly connected component such that for all $v \in U$, $(v, v') \in T \implies v' \in U$. A *path* of a graph is an infinite sequence $x = (v_0, v_1, \dots)$ such that $(v_k, v_{k+1}) \in T$ for each $k \in \mathbb{N}$. A *path fragment* is a finite prefix $\alpha = (v_0, v_1, \dots, v_n)$ of some path.

An *initialized graph* is a pair (G, v_0), where $v_0 \in V$. An *arena* is a 4-tuple $\mathscr{G} = (G, V_A, V_E, v_0)$ where (G, v_0) is an initialized graph and $\{V_A, V_E\}$ is a partition of V . A vertex of V_A is typically represented by a square and a vertex of V_E by a circle. In case that one of the two sets V_A, V_E is empty, we identify the arena with the underlying initialized graph. A *winning condition* Ω on $\mathscr{G}$ is a subset of the set V^ω of infinite sequences on V. For $x \in V^\omega$, we denote by $\inf(x)$ the set of elements of V which appear infinitely often in x.

2.2 Two-player games

We define first classical 2-player games on arenas [9]. In this kind of games, each player plays successively on the arena and the game never stops.

Definition 1. *A* 2-player game *on an arena $\mathscr{G}$ is a pair $\mathbb{G}^2 = (\mathscr{G}, \Omega)$ where Ω a winning condition on $\mathscr{G}$. A* play *on $\mathscr{G}$ is a path on G starting at v_0.*

We call the two players Adam and Eve. Intuitively, the players play the game by moving on vertices a token initially placed in v_0. In each vertex v_i, if $v_i \in V_A$ then Adam moves the token to some vertex v_{i+1} so that $(v_i, v_{i+1}) \in T$. If $v_i \in V_E$ then Eve chooses a successor. Adam wins a play x if $x \in \Omega$, otherwise Eve wins.

Definition 2. *A* strategy *for Adam is a mapping $\phi : V^* \to V$ which, for each path fragment α ending in $v_i \in V_A$, returns a successor vertex $\phi(\alpha) = v_{i+1}$ such that $(v_i, v_{i+1}) \in T$. Adam* follows a strategy *ϕ during a play $x = (v_0, v_1, \dots)$ if for all $i \in \mathbb{N}$ such that $v_i \in V_A$, $\phi(v_0, \dots, v_i) = v_{i+1}$. We say that ϕ is a* winning strategy *for Adam in $\mathbb{G}^2$, if Adam wins each play beginning in v_0 following the strategy ϕ. We say also that Adam wins the game if he has a winning strategy.*

The definition is analogous for Eve. To choose the strategy, a player may just need a memory of bounded size, or no memory at all.

Definition 3. *A* finite memory *strategy for Adam is a mapping $\phi : V \times M \to V$ where M is a finite set, together with an update function $up : V \times M \to M$, and an initial state memory $m_0 \in M$. Adam follows a finite memory strategy ϕ during a play $x = (v_0, v_1, \dots)$ if there is a sequence of memory states $(m_0, m_1, \dots)$ such that for all $i \in \mathbb{N}$, $up(v_i, m_i) = m_{i+1}$ and whenever $v_i \in V_A$, then $\phi(v_i, m_i) = v_{i+1}$. A* memoryless *(or* positional*) strategy for Adam is a mapping $\phi : V \to V$, which for all vertices $v_i \in V_A$ gives a successor vertex $\phi(v_i) = v_{i+1}$.*

Positional strategies can be seen as finite memory strategies where M is a singleton.

2.3 Winning conditions

We can consider several classes of winning conditions on arenas [9], such as reachability conditions, Büchi conditions, Muller conditions or parity conditions.

Definition 4. *A* parity game *on an arena* $\mathscr{G}$ *is a 2-player game together with a colouring function* $c : V \to \mathbb{N}$*, such that the winning condition* Ω *is the set of plays* x *such that the number* $\min(c(\inf(x)))$ *is even.*

A classical theorem is the following:

Theorem 5 ([8, 16]). *If Adam has a winning strategy on a parity game, then he has a positional winning strategy.*

We can define a more general class of winning conditions, using parity automata. The definition of parity automata is standard, see for instance [9]. A winning condition Ω is said to be ω-regular if it is accepted by some parity automaton. One can transform a game with ω-regular winning condition into a parity game by making the synchronised product of the game with the automaton. In this way, one can prove the following well known fact:

Theorem 6. *If Adam has a winning strategy for an* ω*-regular winning condition, then Adam has a finite memory winning strategy.*

The memory needed by Adam is essentially the automaton recognizing the winning condition.

2.4 Probabilistic models

In certain cases, we want to model unpredictable events in systems. Therefore, we want to be able to take transitions in a probabilistic way.

Definition 7. *A* $1\,{}^1\!/_2$-player game *is a triple* $\mathbb{G}^{1{}^1\!/_2} = (\mathscr{G}, p, \Omega)$ *where* $\mathscr{G} = (G, V_A, V_E, v_0)$ *is an arena,* Ω *a winning condition like in 2-player games, while* p *is a probabilistic transition function defined as* $p : V_E \times V \to [0, 1]$ *such that* $p(v_e, v) = 0$ *iff* $(v_e, v) \notin T$ *and for all* $v_e \in V_E$, $\sum_{v \in V} p(v_e, v) = 1$*. If the graph* G *is bipartite then the game is also called Markov Decision Process (MDP). If* $V_A = \emptyset$*, then the game is a* ${}^1\!/_2$-player *game* $\mathbb{G}^{1/2}$ *and it is also known as Markov Chain (MC).*

In $1\,{}^1\!/_2$-player games, Eve is the ${}^1\!/_2$ player because she does not really make any choices. So we will not talk about strategies for Eve. However, strategies for Adam are defined in the same way as in 2-player games. They can be finite memory or memoryless. In this article, we will not need randomized strategies that can be found in the literature [6].

Given a strategy for the full player Adam, one generates a (possibly infinite state) Markov chain by taking the "execution tree" of the graph G and by pruning out all the choices Adam does not take. If the strategy is finite memory the corresponding Markov chain has finitely many states. One can calculate with standard techniques the probability of measurable sets of infinite paths. See for instance [6] for more details. Here we are interested in the following definition:

Definition 8. *We say that Adam has an* almost sure winning strategy *if in the Markov chain generated by the strategy, the winning condition has probability 1. Adam wins almost surely if he has an almost sure winning strategy.*

Note that ω-regular winning conditions are always measurable.

2.5 Banach-Mazur games

A different kind of 2-player game is a game where players do not play only one transition but a sequence of transitions successively, and the alternation is not decided by the arena, but by the players themselves. This game is called a Banach-Mazur game or a path game. It is played on a graph.

Definition 9. *A* Banach-Mazur game $\mathbb{G}^{BM}$ *is given by an initialized graph* (G, v_0)*, and a winning condition* Ω*.*

The two players are B (Banach) and M (Mazur). M begins in v_0 and chooses a path fragment $(v_0^0, v_1^0, \ldots, v_{n_0}^0)$ of size n_0 (also chosen). Then player B does the same from the vertex $v_{n_0}^0$. The game goes infinitely alternating player B and player M turns, so we get an infinite sequence x. Banach wins if $x \in \Omega$, otherwise Mazur wins.

Definition 10. *A play z of a Banach-Mazur game $\mathbb{G}^{BM}$ is an infinite sequence of path fragments $z = (\beta_0\beta_1 \ldots)$ where $\beta_k = (v_0^k, \ldots, v_{n_k}^k)$. The* flattening *of a play z is the corresponding infinite path. Banach wins a play z, if its flattening belongs to Ω, otherwise Mazur wins. A strategy for Banach is a mapping $\psi : (V^*)^* \to V^+$ which for each finite sequence $\gamma = (\beta_0 \ldots \beta_i)$ of path fragments, gives a feasible path fragment $\psi(\gamma) = \beta_{i+1} = (v_0^{i+1}, \ldots, v_{n_{(i+1)}}^{i+1})$. Banach follows a strategy ψ during a play $x = (\beta_0\beta_1 \ldots)$ if for all $i \in \mathbb{N}$ such that i is even, $\psi(\beta_0 \ldots \beta_i) = \beta_{i+1}$. The strategy ψ for Banach is winning in $\mathbb{G}^{BM}$ from v_0, if Banach wins each play starting in v_0 following the strategy ψ. The strategy for Mazur is defined analogously.*

The Banach-Mazur game was proposed by Mazur (see [11], problem 43) as a way of characterising the topological notion of co-meagerness of subsets of the unit interval. Mazur conjectured that the second player has a winning strategy if and only if the winning condition is co-meager in the standard topology. Banach proved this (and he won, as prize, a bottle of wine). Banach-Mazur games were adapted later on graphs (see [4]). Völzer, Varacca and Kindler [15] used them to give a definition of fairness in Kripke structures (equivalent to closed systems). They argued that this game generalizes the known notions of fairness in systems, and they proposed the following definition:

Definition 11. *$\Omega \subseteq V^\omega$ is a fairness property on (G, v_0) if B has a winning strategy in the Banach-Mazur game on (G, v_0) with winning condition Ω.*

We will adapt the game in order to express fairness in open systems, i.e. classical 2-player games.

An important result is the link between the Banach-Mazur games and Markov chains, shown by the following theorem.

Theorem 12 ([14]). *If (G, v_0) is an initialized graph, Ω an ω-regular winning condition and p any probabilistic transition function on V, then Ω has probability 1 in the Markov chain generated on (G, v_0) by p iff Banach wins the Banach-Mazur game on (G, v_0, Ω).*

3 ABM games

3.1 Definitions

After having recalled the known notions of 2-player games and Banach-Mazur games, we now propose a new kind of game that somehow combines those two. In this new game, Adam plays as usual, but Eve is split in two. The two halves are called Banach and Mazur. Intuitively Banach helps Adam (he is good or "Bon" in French), while the real adversary is Mazur (evil, or "Mauvais" in French).

Definition 13. *An* ABM game *is given by an arena $\mathscr{G} = (G, V_A, V_E, v_0)$, and a winning condition Ω.*

The game $\mathbb{G}^{ABM} = (\mathscr{G}, \Omega)^{ABM}$ is played by players A, E_B and E_M. At the beginning of the game, if $v_0 \in V_A$ then A chooses a transition $(v_0, v_1) \in T$. If $v_0 \in V_E$ then E_M chooses the transition. The game goes on in the same way as in 2-player games. States that belong to V_A are controlled by player A and those in V_E by player E_M. After a while, player E_M has to let the control of V_E states to player E_B. Then, it is this one's turn to play against player A before passing the lead to E_M again and so on. The following definition formalizes the rules of play for Banach and Mazur:

Definition 14. *A* move tree λ *of player E_B (or E_M) from a state $v_k \in V_E$, is a finite, prefix closed set of path fragments starting at v_k, and verifying the following conditions:*

- *for each path fragment $\alpha \in V^*$ and each vertex $v \in V_E$, if $\alpha v \in \lambda$ then there is at most one vertex w such that $(v, w) \in T$ and $\alpha v w \in \lambda$;*
- *for each path fragment $\alpha \in V^*$ and each vertex $v \in V_A$, $\alpha v w \in \lambda$ if and only if $(v, w) \in T$.*

In a state of V_A, player A chooses a transition as in the classical 2-player game. But in a state $v_i \in V_E$ the token is moved according to the move tree λ given by E_M or E_B. In fact, if E_M (or E_B) gets the lead in v_k then $(v_k, \ldots, v_i)$ is a prefix of a branch of λ. If v_i has a successor, then E_M (E_B resp.) plays v_{i+1}, which is the unique successor of v_i in the tree. Else, E_M (E_B resp.) passes the lead.

Intuitively, in open systems, player A represents the system while players E_B and E_M the environment. When we want to verify a specification, we assume that the system always makes the right choice but the environment is not necessarily always against. Sometimes the environment helps A to satisfy the property, sometimes the environment plays against. In this way, ABM games allow us to model a *fair* environment.

Like in Banach-Mazur games, we can see an infinite play x as an infinite sequence of tree branches $x = (\beta_0\beta_1 \dots)$ where $\beta_k = (v_0^k, \dots, v_{n_k}^k)$ is a path fragment where E_B is leading if k is odd and E_M is leading if k is even. Players A and E_B win the play x if the flattening of $x \in \Omega$, else E_M wins. In the following sections we will sometimes identify a play with its flattening.

Definition 15. *A strategy ϕ for player A is defined as in 2-player games. A strategy for player E_B or E_M is a mapping $\psi : V^* \to \mathscr{T}$ where $\mathscr{T}$ is the set of move trees. Player A follows a strategy ϕ during a play $x = (\beta_0\beta_1 \dots)$ with flattening $x = (v_0v_1 \dots)$ if for all $i \in \mathbb{N}$ such that $v_i \in V_A$, $\phi(v_0, \dots, v_i) = v_{i+1}$. E_B (E_M resp.) follows a strategy ψ during a play $x = (\beta_0\beta_1 \dots)$ if for all $i \in \mathbb{N}$ such that i is even (odd resp.), $\phi(\beta_0 \dots \beta_i) = \lambda_i$ and β_{i+1} is a branch of λ_i.*

A pair of strategies (ϕ, ψ) (for A and E_B respectively) is said to be winning if A and E_B win the play following their respective strategies. We say that A has a winning strategy ϕ if there exists a winning pair of strategies (ϕ, ψ) for A and E_B. Player A wins the game if he has a winning strategy.

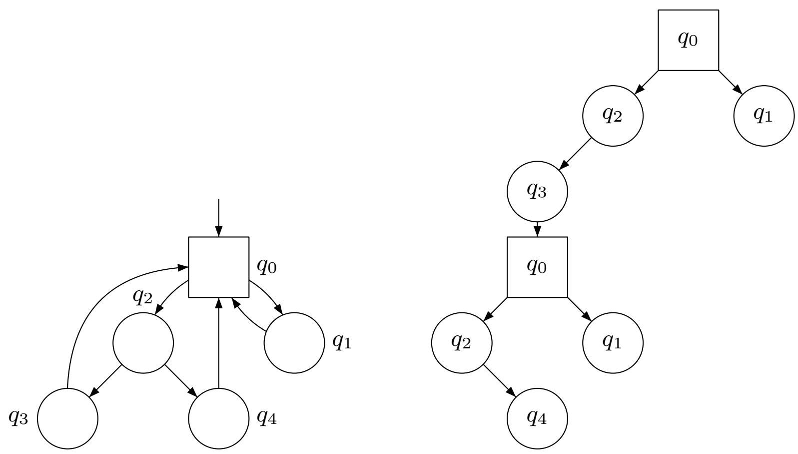

Fig. 1. Example of an arena and a move tree in this arena

If E_M plays the move tree on Fig. 1 at the beginning of the game, he will reach q_3 then q_4 before passing the initiative or will pass his turn if player A chooses the transition to q_1.

Remark. In the definition of ABM games, it is important to notice that a strategy for player A depends only on the sequence of the previously visited vertices. Therefore, A never knows whether he is playing against Banach or Mazur at any given time. That is why there are three players and not only two. This is important for the main result (Theorem 32) to hold.

In the example of Fig. 2, consider the following winning condition: $\Box\Diamond q_3 \wedge \Box\neg q_4$ (infinitely often q_3 but never q_4). Consider the ABM game with this winning condition. If we suppose that player A always knows who is leading in states of V_E, we can construct a winning strategy for players A and E_B. While player E_M is leading, player A takes the transition going to q_1. Then he takes transition to q_2 when E_B gets the lead. Player E_B always takes transition to q_3. Thus, state q_3 is visited infinitely often but state q_4 never. However, if Eve plays randomly (with any Markovian distribution), the winning condition has probability 0 according to the theory of MDPs because state q_4 will be reached almost surely. This contradicts Theorem 32.

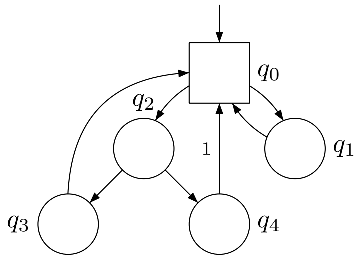

Fig. 2. A 1 1/2-player game

This is also the reason why we cannot code our game in terms of classic 2-player games. A comparison with games with imperfect information (see e.g. [5, 3]) remains to be explored.

3.2 Traps and attractors

Before presenting the main theorems, we need to study some properties of ABM arenas. The following notions were often used in proofs in classical 2-player games [9, 16].

Definition 16. *Let $\mathscr{G} = ((V, T), V_A, V_E, v_0)$ be an arena. A trap for Eve (or E-trap) in $\mathscr{G}$ is a subset of vertices $U \subseteq V$ such that:*

- *for all $v \in U \cap V_E$, $(v, v') \in T \implies v' \in U$,*
- *for all $v \in U \cap V_A$, there exists $v' \in U$ such that $(v, v') \in T$.*

Trap for Adam can be defined in the same way.

The idea of the E-trap is to consider the set of vertices in which Adam can keep the token no matter what does Eve. The following easy proposition was expressed in [16] for 2-player games. It depends only on Adam, and thus it holds also for ABM games.

Proposition 17. *In an ABM game, player A has a memoryless strategy to keep the play in an E-trap.*

Definition 18. *Let $\mathscr{G} = ((V, T), V_A, V_E, v_0)$ be an arena. The attractor of $U \subseteq V$ for Eve (or E-attractor) written $Attr_E(U)$, is the limit of the sequence defined as:*

- $Attr^0_E(U) = U$,
- $\forall i \geq 0, Attr^{i+1}_E(U) = Attr^i_E(U) \cup \{v \in V_E \mid \exists v' \in Attr^i_E(U)$ *such that* $(v, v') \in T\} \cup \{v \in V_A \mid (v, v') \in T \implies v' \in Attr^i_E(U)\}$.

Attractor for Adam can be defined in the same way.

In 2-player games, an E-attractor of a set U induces a strategy for Eve to reach U. In ABM games, vertices of V_E are alternately controlled by E_B and E_M and thus there is no strategy for any of these 2 players to reach U. However, we have the following property on the complement anyway.

Proposition 19 ([16]). *The complement of an E-attractor in an arena is an E-trap.*

3.3 Positional strategies

In the rest of this section, we will show that in the case of ABM games with parity winning conditions, winning strategies for Adam can be memoryless. A classical method to prove such a result is to compute the set of winning positions and show that from these states there exists a positional winning strategy. This technique was used for the 2-player games [16]. In doing so, the set of states is partitioned into winning and losing states. Later, we will observe that ABM games are not determined. However, we still can compute winning and non-winning positions. In the following, G denotes a graph, $\mathscr{G} = (G, V_A, V_E, v_0)$ an arena and $c : V \to \mathbb{N}$ a colouring mapping defining a parity winning condition as in 2-player games.

Theorem 20. *If player A wins the parity game $(\mathscr{G}, c)^{ABM}$ then A has a memoryless winning strategy.*

To start proving this result, let $\mathscr{C}_G^M = \{C_0^M, \dots, C_i^M \dots, C_n^M\}$ be the set of bottom strongly connected components of graph G that have the following property: for all $i < n$, for all $U \subseteq C_i^M$, if U is an E-trap and U is strongly connected then $min(c(U))$ is odd. We write $\mathscr{C}_G^B = \{C_0^B, \dots, C_i^B \dots, C_m^B\}$ the set of bottom strongly connected components of G that do not have that property.

This construction draws its inspiration from the proof [6] of the existence of memoryless strategies in MDP with parity winning condition. In fact, the sets $\mathscr{C}_G^M$ and $\mathscr{C}_G^B$ are inspired from the concept of *controllably win recurrent vertices* [7, 6].

Lemma 21. *For all strategies ϕ for A and ψ for E_B, there exists a play x following ϕ and ψ such that $\inf(x)$ is an E-trap.*

Proof. Fix a pair of strategies (ϕ, ψ) for A and E_B. We will show that E_M can play in order to make x an E-trap. Therefore, we suppose that E_M knows players A and E_B's strategies and chooses his moves according to these. Anytime during the play where E_M has the initiative, we will say that a vertex $q \in V_E$ is *explored* if E_M has already taken all outgoing edges from q since the beginning of his turn. We will say that q is *exhausted* if there is no play that follows ϕ and reaches q. Player E_M will play in this way: when he gets the lead, if there exists a vertex $q \in V_E$ not *explored* and not *exhausted* then E_M tries to reach it. This is possible because q is not *exhausted*. There exists a play following ϕ that allows this. So in this visited state q, he chooses a transition that has not yet been taken since he is leading. E_M repeats this process until all vertices of V_E are marked *explored* or *exhausted*, and then passes the initiative. We notice that an *exhausted* vertex will stay forever *exhausted*. Formally, a move tree λ of E_M is a tree that owns at most one branch λ_0 such that each node $n \in V_E$ of the branch is not a leaf. Indeed, that branch represents the revealed strategy of player A. To agree with the definition of a move tree, E_M passes the lead if A does not follow his strategy ϕ. The branch λ_0 is finite. Suppose that λ_0 is infinite, then there always exists a vertex that is not *explored* and not *exhausted*. That contradicts the finiteness of the graph. Each vertex is eventually either *explored* or *exhausted*. So the tree λ is finite and it is a proper move tree.

Suppose now that for a play x obtained in that way, $\inf(x)$ is not an E-trap. Then there exist vertices $s \in V_E$ and $t \in V$ such that $(s,t) \in T$, $s \in \inf(x)$ and $t \notin \inf(x)$. $s \in \inf(x)$ then s will never be *exhausted* and each time E_M gets the initiative, he will be able to visit its successor t. So we visit t infinitely often. But $t \notin \inf(x)$, contradiction. We can conclude that $\inf(x)$ is an E-trap. □

Lemma 22. *Player A has no winning strategy in the parity game on $\mathscr{C}_G^M$.*

Proof. Let $C_i^M \in \mathscr{C}_G^M$. Assume that A and E_B have a winning pair of strategies (ϕ, ψ) on C_i^M. Then for each play x following the strategies (ϕ, ψ), $U = \inf(x)$ is a strongly connected set such that $min(c(U))$ is even. Thus, by definition of $\mathscr{C}_G^M$, U is not an E-trap. That is a contradiction regarding to previous lemma. As a consequence, A and E_B do not have winning strategies on C_i^M. So A has no winning strategy on $\mathscr{C}_G^M$. □

Lemma 23. *Player A has a memoryless winning strategy in the parity game on $\mathscr{C}_G^B$.*

Proof. Let $C_i^B \in \mathscr{C}_G^B$. By definition of $\mathscr{C}_G^B$, we know that there exists an E-trap: $U \subseteq C_i^B$ such that $min(c(U))$ is even. We write m a vertex, which has minimal colour in U. The strategy ϕ of player A is the following: for all $u \in U$ such that $u \in V_A$, A chooses a successor $v \in U$ such that for all others successors w of u, the distance between v and m is shorter than the one between w and m. And for all $u \notin U$ such that $u \in V_A$, A will choose similarly the successor with shortest distance to reach the set U. The strategy ψ for player E_B consists in using a similar strategy of shortest distance when it is his turn and passing the lead to player E_M each time the vertex m is reached. (In the same way that in Lemma 21, we do not consider the case where player A does not follow his strategy. So the move tree is finite.) Remark that distances to the vertex m and to the set U are well-defined for each vertex of the component C_i^B because of its strong connection. Let x be a play following strategies ϕ and ψ. Then $\inf(x) \subseteq U$ because the strategies allow to reach the set U and to stay in it forever. Thus the minimal vertex infinitely often reached is $min(c(\inf(x))) = m$. We can conclude that the pair of strategies (ϕ, ψ) is winning. Moreover, ϕ is clearly memoryless. So A has a memoryless winning strategy on $\mathscr{C}_G^B$. □

Now we will show that we can partition the set of vertices V of the graph into a winning region for A written W and a non-winning region for A written L. We construct L by induction:

– $G_0 = G$,
– $L_0 = Attr_E(\mathscr{C}^M_{G_0})$,
– $G_{i+1} = G_i$ restricted to $V \setminus L_i$,
– $L_{i+1} = L_i \cup Attr_E(\mathscr{C}^M_{G_{i+1}})$.

G is a finite graph so the sequences $(G_i)_i$ and $(L_i)_i$ converge. We write G_W the limit of the sequence $(G_i)_i$, L the limit of $(L_i)_i$ and $W = V \setminus L$.

Lemma 24. *For all $i \in \mathbb{N}$, player A has no winning strategy on L_i if E_M is leading the states of V_E.*

Proof. Player A has no winning strategy on $\mathscr{C}^M_G$ so he does not have one on L_0 either. Indeed, player E_M can play the strategy induced by the E-attractor to reach $\mathscr{C}^M_G$ where A has no winning strategy. Let $n \in \mathbb{N}$, suppose that A has no winning strategy on L_n if E_M is leading. According to Lemma 22, we know that A has no winning strategy on $\mathscr{C}^M_{G_{n+1}}$ if the play is restricted to the graph G_{n+1}. The only way for A to win would be to always reach L_n when player E_B has the initiative in L_n. By induction hypothesis, if E_M leads in L_n then A has no winning strategy. But for each play where we can reach L_n with E_B, there exists a play where we can reach L_N with E_M. Player E_M only needs to simulate the moves of E_B until he arrives at L_N. So A has no winning strategy on $\mathscr{C}^M_{G_{n+1}}$. As shown previously, player A does not have a strategy in its E-attractor either. Thus A has no winning strategy in $L_{n+1} = L_n \cup Attr_E(\mathscr{C}^M_{G_{n+1}})$ if E_M has the initiative. □

Lemma 25. *W is an E-trap.*

Proof. For all $i \in \mathbb{N}$, $V \setminus L_i$ is an E-trap because it is the complement of an E-attractor. So W is an E-trap. □

Lemma 26. *Player A has a memoryless winning strategy on W.*

Proof. When the token is in $W \setminus \mathscr{C}^B_{G_W}$, strategies for A and E_B consist in strategies of shortest paths to the set $\mathscr{C}^B_{G_W}$ in W similar to previously described strategies. According to Lemma 25, W is an E-trap, so A has a strategy to prevent E_M from reaching L. The shortest distance strategy also allows to stay in W. Furthermore, we can always reach $\mathscr{C}^B_{G_W}$. Indeed, the only bottom strongly connected components reachable from a state of W belong to $\mathscr{C}^B_{G_W}$ by construction of L and W. The strategy of distance is then a winning and memoryless strategy. When $\mathscr{C}^B_{G_W}$ is reached, we can use the strategy described in Lemma 23, which is also winning and memoryless. □

Proof of Theorem 20. At the beginning of the game, E_M leads, so A has no winning strategy on L according to Lemma 24. Lemma 26 says that A has a memoryless winning strategy on $W = V \setminus L$. In consequence, if A has a winning strategy in the initial state then he has a memoryless winning strategy. □

We observe that the proof of Theorem 20 is constructive. It provides explicitly the winning region of Adam. Also, for each state of this region, the winning strategy is explicitly given by Lemma 23 and Lemma 26.

3.4 ABM games are not determined

Lemma 22 says that players A and E_B have no winning strategy in the parity game on $\mathscr{C}_G^M$. In general, player E_M has no winning strategy either on $\mathscr{C}_G^M$. We can infer that the game is not determined.

Consider the parity game represented on Fig. 3. Lemma 22 says that players A and E_B do not have any strategies to win the game. However, we can observe here that player E_M does not have a strategy either. If he wants to win, E_M has either to reach infinitely often state q_4 or reach infinitely often q_1 and a finite number times q_3. But E_M does not know player A's strategy. If A were to take the transition going to q_2 a finite number of times, then E_M could just pass to E_B without making any moves. But since he does not know actually if player A intends to take this transition infinitely often or not, there are two cases.

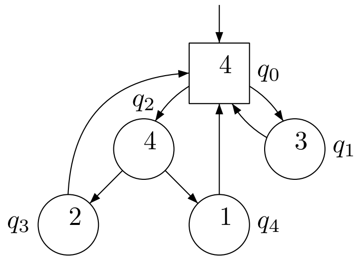

Fig. 3. Example of parity game

- If player E_M supposes that player A will reach q_2 infinitely often, then his strategy has to wait for this move and reach state q_4 before passing the lead to player E_B. However, if eventually A never reaches q_2, then E_M would never pass his turn. As a consequence, this is not a strategy because the move tree chosen by E_M must be finite.
- If E_M supposes that A will never reach q_2 again at a certain point, then he will pass his turn in q_1. But we can imagine a scenario where each time E_M passes the initiative, A take the transition to q_2 and let E_B reach q_3.

In any case, we cannot define any winning strategy for player E_M in this game.

3.5 Finite memory

We conclude the section by extending Theorem 20 to ω-regular winning conditions, similarly to Theorem 6.

Theorem 27. *Let $\mathscr{G}$ be an arena and $\Omega \subseteq V^\omega$ an ω-regular condition. If player A has a winning strategy in the game $(\mathscr{G}, \Omega)^{ABM}$ then he has a finite-memory winning strategy.*

The proof technique is similar to the one used for Theorem 6. One makes the product with a deterministic parity automaton recognising the winning condition. The automaton is essentially the memory needed by Adam.

4 Fairness as randomization

In this section, we intend to demonstrate a theorem similar to Theorem 12 in the context of open systems. That is we want to build a connection between

ABM games and 1 1/2-player games. We first do this for the parity case, and then extend to all ω-regular conditions.

We start by noting that the existence of memoryless strategies for ABM games is mirrored in MDP.

Theorem 28 ([7, 6]). *Let $\mathscr{G} = (G, V_A, V_E, v_0)$ be an arena, $p : V_E \times V \to [0, 1]$ a probabilistic transition function on V_E such that $p(v_e, v) = 0$ iff $(v_e, v) \notin T$ and for all $v_e \in V_E$, $\sum_{v \in V} p(v_e, v) = 1$, and $c : V \to \mathbb{N}$ a colouring mapping. If Adam wins almost surely the parity 1 1/2-player game $(\mathscr{G}, p, c)$ then Adam has a memoryless almost sure winning strategy.*

4.1 The parity case

Theorem 29. *Let $\mathscr{G} = (G, V_A, V_E, v_0)$ be an arena, c a colouring, p a probabilistic transition function on V_E and $\Omega \subseteq V^\omega$ the parity winning condition defined by c. If A has a winning strategy in the game $(\mathscr{G}, \Omega)^{ABM}$ then Adam has an almost-sure winning strategy in the 1 1/2-player game $(\mathscr{G}, p, \Omega)$.*

Proof. Assume that player A has a winning strategy in the game $(\mathscr{G}, \Omega)^{ABM}$. Then according to Theorem 20, A has a memoryless winning strategy ϕ. For each state of V_A, we keep only the outgoing edge provided by ϕ. Let $\mathscr{K} = ((V, T^K), v_0)$ be the initialized graph where $T^K = T \setminus \{(v_a, v) \in T \mid v_a \in V_A \text{ and } \phi(v_a) \neq v\}$. We can easily see that, if player A wins in $(\mathscr{G}, \Omega')^{ABM}$ then Banach wins in $(\mathscr{K}, \Omega')^{BM}$. Indeed, each play that is winning for A on $\mathscr{G}$ can be simulated on $\mathscr{K}$. It does not matter if Banach or Mazur has the initiative in a state of V_A because there is only one outgoing edge.

Let p' be the probabilistic transition function on V such that for all $v_e \in V_E$, $v \in V$, $p'(v_e, v) = p(v_e, v)$ and for all $v_a \in V_A$, $(v_a, v) \in T \implies p'(v_a, v) = 1$. This defines a Markov chain on $((V, T^K), v_0)$. Thanks to Theorem 12, we know that the winning condition Ω has probability 1 in this Markov chain. But this shows that ϕ is an almost sure winning strategy for Adam in the 1 1/2-player game $(\mathscr{G}, p, \Omega)$. □

Theorem 30. *Let $\mathscr{G} = (G, V_A, V_E, v_0)$ be an arena, c a colouring, p a probabilistic transition function on V_E and Ω the parity winning condition defined by c. If Adam has an almost sure winning strategy in the 1 1/2-player game $(\mathscr{G}, p, \Omega)$, then A has a winning strategy in the game $(\mathscr{G}, \Omega)^{ABM}$.*

Proof. Suppose that Adam has an almost sure winning strategy in the game $(\mathscr{G}, p, \Omega)$. Then according to Theorem 28, Adam has a memoryless almost sure winning strategy. Let $\mathscr{K} = ((V, T^K), v_0)$ be the initialized graph where $T^K = T \setminus \{(v_a, v) \in T \mid v_a \in V_A \text{ and } \phi(v_a) \neq v\}$. Let p' be the probabilistic transition function on V such that for all $v_e \in V_E$, $v \in V$, $p'(v_e, v) = p(v_e, v)$ and for all $v_a \in V_A$, $(v_a, v) \in T \implies p'(v_a, v) = 1$. This generates a Markov chain. As Adam wins almost surely in $(\mathscr{G}, p, \Omega)$ then Ω has probability 1 in the Markov chain. By Theorem 12, Banach has a winning strategy in the game $(\mathscr{K}, \Omega)^{BM}$. This means that ϕ is a winning strategy for Adam in the game $(\mathscr{G}, \Omega)^{ABM}$.

The winning strategy of player E_B is to simulate Banach winning strategy in $(\mathscr{K}, \Omega)^{BM}$. Thus Adam and Banach have also a winning strategy in the game $(\mathscr{G}, \Omega)^{ABM}$. □

The following theorem results from Theorem 30 and Theorem 29.

Theorem 31. *Let $\mathscr{G} = (G, V_A, V_E, v_0)$ be an arena, c a colouring, p a probabilistic transition function on V_E and Ω the parity winning condition defined by c. Adam has an almost-sure winning strategy in the $1\,1/2$-player game $(\mathscr{G}, p, \Omega)$ if and only if A has a winning strategy in the game $(\mathscr{G}, \Omega)^{ABM}$.*

4.2 ω-regular conditions

The key fact in order to exploit Theorem 12 is that the graph obtained after applying the memoryless strategy of Adam is finite, as Theorem 12 applies only to finite graphs. We notice thus that Theorem 20 on the existence of memoryless strategies is essential in the proof of Theorem 32. In the case of ω-regular winning conditions, the strategy of Adam is finite memory. As in the memoryless case, the key observation is that the graph one gets by applying the strategy is finite (though larger than the original graph). Thus, it is still possible to apply Theorem 12. We omit the straightforward details of the proof.

Theorem 32. *Let $\mathscr{G} = (G, V_A, V_E, v_0)$ be an arena, p a probabilistic transition function on V_E and $\Omega \subseteq V^\omega$ an ω-regular condition. Adam has an almost-sure winning strategy in the $1\,1/2$-player game $(\mathscr{G}, p, \Omega)$ if and only if A has a winning strategy in the game $(\mathscr{G}, \Omega)^{ABM}$.*

We can notice that if $V_A = \emptyset$ then we have the special case of Theorem 12.

Thus, we showed that playing against a fair player is equivalent to playing against a probabilistic player in the case of ω-regular properties.

5 Related and Future Work

The Banach-Mazur game is one possible definition of fairness in closed systems. An equivalent topological definition can be given in terms of co-meagerness. In [1], the topological definition is used to prove the equivalence between probabilistic and fair semantics of timed automata. Interestingly, this equivalence holds only for one-clock automata, but it breaks down once we allow more than one clock. Another equivalent definition is in terms of α-fairness [10]. Of the three definitions, this is the one that most resembles the intuitive notion of fairness "if something is often possible, it will be often performed". It would be interesting to define fair strategies for Eve in terms of α-fairness. We also expect that this game-theoretic point of view can be applied to improve existing algorithms, or to find new ones, in the qualitative model checking of MDPs.

In this paper, we have applied a definition of fairness to one of the players of 2-player games. In general, we could study what happens to other players and other games. For instance, a $1\,1/2$-player game where Adam plays fairly should be equivalent to a Markov chain.

References

1. C. Baier, N. Bertrand, P. Bouyer, T. Brihaye, and M. Größer. Almost-sure model checking of infinite paths in one-clock timed automata. In *LICS'08*, pages 217–226. IEEE Computer Society, 2008.
2. C. Baier and M. Kwiatkowska. Model checking for a probabilistic branching time logic with fairness. *Distributed Computing*, 11(3):125–155, 1998.
3. N. Bertrand, B. Genest, and H. Gimbert. Qualitative determinacy and decidability of stochastic games with signals. In *LICS'09*, pages 319–328. IEEE Computer Society, 2009.
4. D. Berwanger, E. Grädel, and S. Kreutzer. Once upon a time in the West. Determinacy, complexity and definability of path games. In *LPAR'03*, LNCS 2850, pages 226–240. Springer, 2003.
5. K. Chatterjee, L. Doyen, T. A. Henzinger, and J.-F. Raskin. Algorithms for omega-regular games with imperfect information. *Logical Methods in Computer Science*, 3(3), 2007.
6. K. Chatterjee, M. Jurdziński, and T. A. Henzinger. Quantitative stochastic parity games. In *SODA'04*, pages 114–123. ACM/SIAM, 2004.
7. C. Courcoubetis and M. Yannakakis. The complexity of probabilistic verification. *Journal of the ACM*, 42(4):857–907, 1995.
8. E. A. Emerson and C. Jutla. Tree automata, mu-calculus and determinacy (extended abstract). In *FOCS'91*, pages 368–377. IEEE, 1991.
9. E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games*. LNCS 2500. Springer, 2002.
10. O. Lichtenstein, A. Pnueli, and L. D. Zuck. The glory of the past. In *Logic of Programs*, LNCS 193, pages 196–218. Springer, 1985.
11. D. Mauldin, editor. *The Scottish Book*. Birkhäuser, 1981.
12. M. Schmalz, D. Varacca, and H. Völzer. Counterexamples in probabilistic LTL model checking for Markov chains. In *CONCUR'09*, LNCS 5710, pages 587–602. Springer, 2009.
13. M. Schmalz, H. Völzer, and D. Varacca. Model checking almost all paths can be less expensive than checking all paths. In *FSTTCS'07*, LNCS 4855, pages 532–543. Springer, 2007.
14. D. Varacca and H. Völzer. Temporal logics and model checking for fairly correct systems. In *LICS'06*, pages 389–398. IEEE Computer Society, 2006.
15. H. Völzer, D. Varacca, and E. Kindler. Defining fairness. In *CONCUR'05*, LNCS 3653, pages 458–472. Springer, 2005.
16. W. Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theor. Computer Science*, 200:135–183, 1998.