

MPRI
Game Theory Techniques in Computer Science

Olivier Serre
olivier.serre@irif.fr
`www.irif.fr/~serre`

September 17, 2019

Two-player games on finite graphs: basic definitions

Contents

1.1 Graph, arena, game	3
1.2 Winning conditions	4
1.3 Strategies et winning positions	6
1.3.1 Winning positions	7
1.3.2 Some specific winning strategies	7

In this chapter, we introduce the basic objects. We start with the notion of two-player games on graphs, introducing the notion of graphs, arenas, plays and winning conditions. We then introduce the main winning condition that we consider in this course. We finally conclude by giving the notion of strategies (together with special cases) and of winning positions.

1.1 Graph, arena, game

Let Σ be a possibly infinite set. We denote by Σ^* the set of finite sequences (or words depending on the context) over Σ and by Σ^ω the set of infinite sequences indexed by \mathbb{N} (or infinite words depending on the context). Finally, $\Sigma^\infty = \Sigma^* \cup \Sigma^\omega$. For a word $u \in \Sigma^*$, we let u^ω denote the infinite word obtained by repeating ad infinitum the word u : $u^\omega = u \cdots u \cdots u \cdots$. Finally, we let ε denote the empty word.

DEFINITION 1.1 A **graph** is a pair $G = (V, E)$ where V is a set of vertices and $E \subseteq V \times V$ is a set of edges. A graph is finite iff V is finite

The graphical representation of a graph is given in the example below.

EXAMPLE 1.1 Figure 1.1 gives the graphical representation of the graph $G = (V, E)$ where:

- $V = \{1, 2, 3, 4\}$.
- $E = \{(1, 2), (1, 2), (1, 4), (2, 1), (2, 2), (4, 2)\}$.

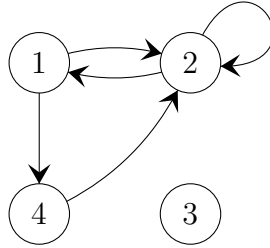


Figure 1.1: Graphical representation of a graph

The two players of our games will be called Eve and Adam (Other terminology are 0 and 1, Éloïse and Abelard, existential player and universal player...).

We fix a graph $G = (V, E)$ and a partition of the vertices $V = V_E \uplus V_A$ among the two players : the vertices (positions) in V_E belong to (are controlled by) Eve while those in V_A belong to Adam.

A triple $\mathcal{G} = (G, V_E, V_A)$ is called an **arena** (alternatively a **game graph**). The graphical representation of an arena is the same as a graph except that we represent by circles those vertices that belong to Eve and by squares those vertices that belong to Adam. Figure 1.2 provide such a representation for $\mathcal{G} = (G, V_E, V_A)$ where $V_E = \{2, 4, 7, 8\}$ and $V_A = \{1, 3, 5, 6\}$.

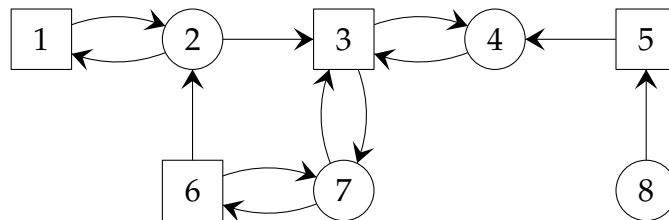


Figure 1.2: Example of an arena

1.2 Winning conditions

A **winning condition** on \mathcal{G} is a subset Ω of V^ω .

Finally, a **two-player (perfect information) game** on an arena \mathcal{G} is a pair $\mathbb{G} = (\mathcal{G}, \Omega)$, where \mathcal{G} is an arena and Ω is a winning condition on \mathcal{G} .

A **play** in a game $\mathbb{G} = (\mathcal{G}, \Omega)$ starting in v_0 is defined as follows. The players move a token along edges of the graph, and at the beginning the player who controls the initial vertex v_0 chooses, if exists, an edge $e_1 = (v_0, v_1)$ with origin v_0 . If such an edge does not exists, the play ends. Otherwise, the token goes in v_1 and the player who controls v_1 chooses the next edge (v_1, v_2) , if exists, and so on. Hence a play is a maximal (finite or infinite) path $v_0 v_1 v_2 \dots$ in G . A **partial play** is a prefix of a play, *i.e.* a finite path in G . In the following we use λ to denote (partial) plays.

The sequence $12376(21)^\omega$ is an infinite play in the arena given in Figure 1.2.

REMARK 1.1 In the following we should use terminology *game* without precisizing that there are two-player. It is however important to note that there are many other games: some with more than two players (multiplayer games), randomized/stochastic games, imperfect information games (where the player may not know about the exact vertex of the play), concurrent games (where the players play simultaneously an action and where the new vertex depends on the previous one and on the actions chosen by the players). We may discuss some of these variants much later in this course

Eve **win** an infinite play iff it belongs to Ω , otherwise it is Adam who wins. We may say that the play is **winning** for Eve (of for Adam otherwise). In the case of a finite play (hence ending in a dead-end vertex) the player that cannot move loses the play.

For some given winning condition Ω , we consider the **dual winning condition** $\bar{\Omega}$ as the one $\bar{\Omega} = V^\omega \setminus \Omega$.

We now give some example of winning conditions. We start with reachability and safety winning conditions.

DEFINITION 1.2 Let $F \subseteq V$ be a subset of **final states**. The winning condition V^*FV^ω is said to be a **reachability condition**. Hence, Eve wins a play iff a final state is eventually visited.

The dual winning condition $(V \setminus F)^\omega$ of a reachability condition is said to be a **safety condition**. In this setting we say that those vertices in F are the **forbidden vertices**. Hence, Eve wins a play iff it never visits a forbidden state.

We define now the Büchi, co-Büchi, parity and Muller winning conditions.

DEFINITION 1.3 Let $F \subseteq V$ be a subset of **final vertices**. The winning condition $(V^*F)^\omega$ is said to be a **Büchi condition**. Hence, Eve wins a play iff it visits infinitely often a final state.

The dual condition $\bigcup_{i \geq 0} (V^*F)^i (V \setminus F)^\omega$ of a Büchi condition is a **co-Büchi condition**. In this setting we say that those vertices in F are the **forbidden vertices**. Hence, Eve wins a play iff it visits only finitely often forbidden states.

DEFINITION 1.4 Let $C \subseteq \mathbb{N}$ be a finite subset of integers, often called **colours** and let ρ be an application from V into C , called **colouring function**. The winning condition

$$\{v_0v_1v_2 \cdots \in V^\omega \mid \limsup(\rho(v_i))_{i \geq 0} \text{ is even}\}$$

is said to be a **parity condition**. Hence, Eve wins a play iff the largest colour infinitely often visited is even.

The dual condition

$$\{v_0v_1v_2 \cdots \in V^\omega \mid \limsup(\rho(v_i))_{i \geq 0} \text{ is odd}\}$$

is a **co-parity winning condition**. Hence, Eve wins a play iff the largest colour infinitely often visited is odd.

REMARK 1.2 Consider a co-parity winning condition Ω given by a colouring function ρ . Consider the colouring function ρ' defined by $\rho'(v) = \rho(v) + 1$. One easily check that the parity condition induced by ρ' is the same as Ω .

REMARK 1.3 A variant of the parity condition consists in considering the largest infinitely often visited colour. Such a condition is known as a **min-parity** condition by opposition with **max-parity** (that we simply refer to as parity condition in this document). It is easily seen that in the context of finite arenas those conditions are equi-expressive.

EXERCISE 1.1 Prove that any min-parity condition can be expressed as a max-parity condition.

EXERCISE 1.2 Prove that any parity condition can be expressed as a Boolean combination of Büchi conditions.

DEFINITION 1.5 Let $C \subseteq \mathbb{N}$ be a set of integers, and let $\rho : V \rightarrow C$, be a **colouring function**. Let \mathcal{F} be a collection of subsets of C , called **final sets**. The winning condition

$$\{v_0v_1v_2 \cdots \in V^\omega \mid \{c \mid \exists^\infty i, \rho(v_i) = c\} \in \mathcal{F}\}$$

is a **Muller condition**. Hence, Eve wins a play iff the set of colours infinitely often visited belongs to \mathcal{F} .

The dual of a Muller condition \mathcal{F} is again a Muller condition given by $\mathcal{P}(C) \setminus \mathcal{F}$.

1.3 Strategies et winning positions

For the rest of this section, we fix an arena $\mathcal{G} = (G, V_E, V_A)$, where $G = (V, E)$.

Strategies

A **strategy** for Eve is a partial function $\varphi : V^* \rightarrow V$ such that, for all partial play λ ending in some vertex $v \in V_E$, $(v, \varphi(\lambda)) \in E$, i.e. a strategy should provide a valid move. Strategies for Adam are defined as functions $\psi : V^* \rightarrow V$ such that, for all partial play λ ending in some vertex $v \in V_A$, $(v, \psi(\lambda)) \in E$

Given a strategy φ for Eve, she can **respect it** by always moving as indicated by φ . Formally, we shall say that Eve respects φ during a play $\lambda = v_0v_1v_2 \cdots$ if, for all $0 \leq i < |\lambda|$, $v_{i+1} = \varphi(v_0 \cdots v_i)$ whenever $v_i \in V_E$. In a similar way we define the fact that Adam respects some strategy ψ .

Interesting strategies are those that can be used all along a play. We shall say that a strategy is **usable** from a vertex v if, for all partial play λ starting from v where Eve respects φ , and ending in a vertex in V_E , $\varphi(\lambda)$ is defined.

Consider now a game $\mathbb{G} = (\mathcal{G}, \Omega)$, an initial position v_0 and a strategy φ for Eve. We say that φ is a **winning strategy for Eve in \mathbb{G} from v_0** if any play starting from v_0 where Eve respects φ is won by Eve. We defined in a similar way winning strategies for Adam. The game \mathbb{G} is **determined** if, for all initial vertex, one of the two player has a winning strategy from that position.

REMARK 1.4 In any game, there is, for any initial vertex, at most one player who has a winning strategy. Indeed, if both had one, the play where they follow their respective winning strategy would be both inside and outside the winning condition.

In this course, all the game we shall considered are determined.

1.3.1 Winning positions

A vertex v in a game \mathbb{G} is a **winning position** for Eve if she has a winning strategy in \mathbb{G} from v . The set of winning positions for Eve will be often denoted $W_{\mathbf{E}}$. Similarly we define the set $W_{\mathbf{A}}$ of winning vertices for Adam. A game is determined iff $V = W_{\mathbf{E}} \cup W_{\mathbf{A}}$.

1.3.2 Some specific winning strategies

We fix an arena $\mathcal{G} = ((V, E), V_{\mathbf{E}}, V_{\mathbf{A}})$ and a game $\mathbb{G} = (\mathcal{G}, \Omega)$. In the following we discuss special kind of strategies.

We start with *memoryless* or *positional* strategies. A strategy is memoryless if the move it provides only depend on the current vertex, not on the past.

DEFINITION 1.6 *A strategy φ is **memoryless** (or **positional**) if for all partial play λ and λ' ending in the same vertex, $\varphi(\lambda) = \varphi(\lambda')$. Hence, memoryless strategies for Eve (resp. for Adam) can be identified with those maps φ from $V_{\mathbf{E}}$ (resp. $V_{\mathbf{A}}$) such that for all $v \in V_{\mathbf{E}}$ $(v, \varphi(v)) \in E$.*

In the case of a finite arena, a memoryless strategy admits a finite description. This is no longer true if the arena is infinite.

An intermediate notion is the one of finite memory strategy.

DEFINITION 1.7 *Let M be a set called **memory**. A **strategy with memory** M for Eve is a tuple (m_0, φ, up) made of an element $m_0 \in M$ called initial memory, of a map φ from $V \times M$ into V and such that for all (v, m) , $(v, \varphi(v, m)) \in E$, and of an application up from $M \times V$ into M called update function. Eve respects φ during a play $\lambda = v_0v_1v_2, \dots$ if for all $i < |\lambda|$, $v_{i+1} = \varphi(v_i, m_i)$, where we let $m_i = up(m_{i-1}, v_i)$ for all $i \geq 1$. In other words, when Eve has to play she considers the value of the memory and the current vertex to chose her move (thanks to φ); and afeter every move (made either by Eve or Adam), Eve updates the memory thanks to the function up by considering the former value and the new vertex reached.*

When M is a singleton, we obtain memoryless strategies.

Let φ be a (general) strategy and let v_0 be an initial position. Consider the strategy with memory M , (m_0, ψ, up) where we let $M = V^*$, $m_0 = v_0$, $\psi(v, m) = \varphi(m)$ and $up(v_0 \dots v_n, v_{n+1}) = (v_0 \dots v_n v_{n+1})$. This strategy save in the memory the history of the play, hence it is straightforward to prove that the set of plays where Eve respects strategy (m_0, ψ, up) are exactly those where Eve respects φ .

A very interesting case is the one of **finite memory strategies**, that is those strategy with memory M where M is finite. When the arena is finite, those strategies can be implemented using a finite transducer (finite automaton with output).

Two-player games on finite graphs: ω -regular winning conditions

Contents

2.1	Reachability games	9
2.2	Büchi Games	11
2.3	Parity games	12
2.3.1	Basic definitions	12
2.3.2	Proof of Theorem 2.3	13
2.4	Jeux de Muller	15
2.4.1	Les stratégies gagnantes peuvent requérir de la mémoire	15
2.4.2	Résolution des jeux de Muller: le last appearance record	16

In this chapter we consider two-player games on finite graphs equipped with an ω -regular winning conditions: reachability, safety, Büchi, co-Büchi, parity and Muller. We study them from an algorithmic point of view, looking from algorithms to compute the winning regions. We will also prove that we can construct memoryless strategies for all winning condition but Muller conditions (which require finite memory).

2.1 Reachability games

We fix a graph $G = (V, E)$ without dead-ends, a partition $V_E \uplus V_A$ of V defining an arena \mathcal{G} . We fix a subset $F \subseteq V$ of final vertices and we denote by \mathbb{G} the corresponding reachability game

We have the following result.

THEOREM 2.1 *One can compute the winning positions in \mathbb{G} and construct memoryless winning strategies for both players.*

The rest of the section is devoted to the proof of this result. We defined by induction the following increasing sequence of subsets of V .

$$\begin{cases} Attr_0^{\mathbf{E}}(F) = F \\ Attr_{i+1}^{\mathbf{E}}(F) = Attr_i^{\mathbf{E}}(F) \cup \{v \in V_{\mathbf{E}} \mid \exists v' \in Attr_i^{\mathbf{E}}(F) \text{ s.t. } (v, v') \in E\} \\ \quad \cup \{v \in V_{\mathbf{A}} \mid \forall v' \in V, (v, v') \in E \Rightarrow v' \in Attr_i^{\mathbf{E}}(F)\} \end{cases}$$

As the previous sequence is increasing and bounded, it converges: denote by $Attr^{\mathbf{E}}(F)$ its limit. We will show that $Attr^{\mathbf{E}}(F) = W_{\mathbf{E}}$ is the winning region for Eve. The set $Attr^{\mathbf{E}}(F)$ is called **Attractor for Eve to F** and its complement is said to be a **trap for Eve**.

We define the function $rk : V \rightarrow \mathbb{N}$ by letting $rk(v) = \min\{i \mid v \in Attr_i^{\mathbf{E}}(F)\}$ (with the convention that $\min \emptyset = \infty$). Then we have the following property.

PROPERTY 2.1 *For all $v \in Attr^{\mathbf{E}}(F)$, let $v \notin F$. Then the following holds*

- *Either $v \in V_{\mathbf{E}}$ and v has a successor v' in $Attr^{\mathbf{E}}(F)$ such that $rk(v') < rk(v)$;*
- *Or $v \in V_{\mathbf{A}}$ and for all successor v' of v , v' belongs to $Attr^{\mathbf{E}}(F)$ and $rk(v') < rk(v)$.*

We finally define a memoryless strategy φ for Eve by letting for all $v \in Attr^{\mathbf{E}}(F) \cap V_{\mathbf{E}}$, $\varphi(v) = v'$ for some v' such that $rk(v') < rk(v)$ (such a v' exists thanks to Property 2.1). For those vertices outside of $Attr^{\mathbf{E}}(F)$ or in F , φ can take any value.

We verify that if Eve respect φ during a play starting from some vertex in $Attr^{\mathbf{E}}(F)$, then the play stays in $Attr^{\mathbf{E}}(F)$ until it reaches F . Indeed, consider a play $\lambda = v_0v_1 \dots$ where she respects φ . By definition, one has, for all $i \geq 0$, that $v_i \in Attr^{\mathbf{E}}(F) \setminus F \Rightarrow v_{i+1} \in Attr^{\mathbf{E}}(F)$. Indeed, we have two possible cases: either λ visits F (hence is winning), or it never visits F and thus $v_i \in Attr^{\mathbf{E}}(F)$ for all $i \geq 0$. In this latter case, one has $rk(v_0) < rk(v_1) < rk(v_2) < \dots$ which leads an infinite decreasing sequence in \mathbb{N} , leading a contradiction. The strategy φ is therefore winning from any vertex in $Attr^{\mathbf{E}}(F)$.

It remains to prove that for any vertex in $V \setminus Attr^{\mathbf{E}}(F)$, Adam has a memoryless winning strategy. We have a dual version of Property 2.1

PROPERTY 2.2 *For all $v \notin Attr^{\mathbf{E}}(F)$,*

- *if $v \in V_{\mathbf{E}}$, for all successor v' of v , $v' \notin Attr^{\mathbf{E}}(F)$, i.e. $rk(v') = \infty$.*
- *if $v \in V_{\mathbf{A}}$, v has a successor $v' \notin Attr^{\mathbf{E}}(F)$, i.e. $rk(v') = \infty$;*

We define a memoryless strategy ψ for Adam by letting for all $v \notin Attr^{\mathbf{E}}(F) \cap V_{\mathbf{A}}$, $\psi(v) = v'$ for some v' such that $v' \notin Attr^{\mathbf{E}}(F)$ (such a v' exists from Property 2.2).

We verify that if Adam respects ψ in a play starting from a vertex not in $Attr^{\mathbf{E}}(F)$, then it stays outside of $Attr^{\mathbf{E}}(F)$ forever. Indeed, let $\lambda = v_0v_1 \dots$ be a play where Adam respects ψ and starting from $v_0 \notin Attr^{\mathbf{E}}(F)$. Property 2.2 and definition of ψ , imply directly that $v_i \notin Attr^{\mathbf{E}}(F)$ for all i . This concludes the proof.

REMARK 2.1 The set $V \setminus Attr^{\mathbf{E}}(F)$ is said to be a **trap for Eve**. This terminology comes from the fact that Adam has a strategy to *trap* Eve in $V \setminus Attr^{\mathbf{E}}(F)$, i.e. to maintain the play in the set.

2.2 Büchi Games

We fix a graph $G = (V, E)$ without dead-end, a partition $V_{\mathbf{E}} \uplus V_{\mathbf{A}}$ of V that defines an arena \mathcal{G} . We fix a subset $F \subseteq V$ of final states and we denote by \mathbb{G} the associated Büchi game.

We have the following result.

THEOREM 2.2 *One can compute the winning positions in \mathbb{G} and construct memoryless winning strategies for both players.*

We first start by defining a variant of the attractor. Let S be a subset of states. We define the following increasing sequence

$$\begin{cases} Attr_0^{+\mathbf{E}}(S) = \{v \in V_{\mathbf{E}} \mid \exists v' \in S \text{ s.t. } (v, v') \in E\} \cup \{v \in V_{\mathbf{A}} \mid (v, v') \in E \Rightarrow v' \in S\} \\ Attr_{i+1}^{+\mathbf{E}}(S) = Attr_i^{+\mathbf{E}}(S) \cup \{v \in V_{\mathbf{E}} \mid \exists v' \in Attr_i^{+\mathbf{E}}(S) \text{ s.t. } (v, v') \in E\} \\ \quad \cup \{v \in V_{\mathbf{A}} \mid \forall v', (v, v') \in E \Rightarrow v' \in Attr_i^{+\mathbf{E}}(S) \cup S\} \end{cases}$$

And we denote $Attr^{+\mathbf{E}}(S)$ its limit (we refer to this set as the strict attractor to S for Eve).

An easy adaptation of the proof for reachability games allows us to prove that a vertex belongs to $Attr^{+\mathbf{E}}(S)$ iff Eve has a strategy that ensures her to reach in **at least one move** S .

We now consider the following (decreasing¹) sequence:

$$Z_i = \begin{cases} F & \text{if } i = 1 \\ Attr^{+\mathbf{E}}(Z_{i-1}) \cap F & \text{if } i > 1 \end{cases}$$

We denote by Z_∞ its limit and in particular we have that $Z_\infty = Attr^{+\mathbf{E}}(Z_\infty) \cap F$ and $Z_\infty \subseteq F$. We deduce from equality $Z_\infty = Attr^{+\mathbf{E}}(Z_\infty) \cap F$ that Eve has a positional strategy from Z_∞ to come back to Z_∞ in at least one move. We denote by φ_1 this strategy.

The intuitive idea is that Z_i consists of those final states from which Eve can force at least i visits to F . Finally, Z_∞ is the subset of F from which Eve can force infinitely many visits to F . It is therefore natural to consider the set $Attr^{\mathbf{E}}(Z_\infty)$ of those states from which Eve can force to reach Z_∞ .

We will prove that $W_{\mathbf{E}} = Attr^{\mathbf{E}}(Z_\infty)$ and for this we let φ_0 be a positional strategy of Eve in the reachability game to Z_∞ . We finally define the following positional strategy φ for Eve

$$\varphi(v) = \begin{cases} \varphi_0(v) & \text{if } v \in (Attr^{\mathbf{E}}(Z_\infty) \setminus Z_\infty) \cap V_{\mathbf{E}} \\ \varphi_1(v) & \text{if } v \in Z_\infty \cap V_{\mathbf{E}} \end{cases}$$

Therefore, strategy φ leads the play in Z_∞ (mimicking φ_0) and then once Z_∞ is reached, leave it by taking care to have the possibility to come back to it (this property is ensured by strategy φ_1).

¹The sequence is decreasing. Indeed, one has $Z_1 \subseteq Z_0 = F$ by definition; now assuming that $Z_{i+1} \subseteq Z_i$ one has by monotonicity of the $Attr^{+\mathbf{E}}$ operator that $Attr^{+\mathbf{E}}(Z_{i+1}) \subseteq Attr^{+\mathbf{E}}(Z_i)$ hence that $Z_{i+2} = Attr^{+\mathbf{E}}(Z_{i+1}) \cap F \subseteq Attr^{+\mathbf{E}}(Z_i) \cap F = Z_{i+1}$.

Also note that it implies by induction that an alternative definition of Z_i can be $Attr^{+\mathbf{E}}(Z_{i-1}) \cap Z_{i-1}$ as $Attr^{+\mathbf{E}}(Z_{i-1}) \cap F = Attr^{+\mathbf{E}}(Z_{i-1}) \cap Z_{i-1}$ because of the inclusion $Z_i \subseteq Z_{i-1} \subseteq F$.

Consider a play starting from $\text{Attr}^{\mathbf{E}}(Z_\infty)$ and where Eve respects φ : such a play visits infinitely often $Z_\infty \subseteq F$ and is therefore winning. Indeed, any vertex visited along such a play belongs to $\text{Attr}^{\mathbf{E}}(Z_\infty)$ and she systematically plays an attractor strategy to Z_∞ : hence one can factorize such a play into partial plays that are all ending in Z_∞ .

This proves that $\text{Attr}^{\mathbf{E}}(Z_\infty) \subseteq W_{\mathbf{E}}$. To conclude the proof, it remains to prove that $V \setminus (\text{Attr}^{\mathbf{E}}(Z_\infty)) \subseteq W_{\mathbf{A}}$ and give a positional strategy for Adam from those vertices.

Let $v \in V \setminus (\text{Attr}^{\mathbf{E}}(Z_\infty))$: there exists some i such that $v \notin \text{Attr}^{\mathbf{E}}(Z_i)$ (consider i such that $Z_i = Z_\infty$). We chose i such that $v \in \text{Attr}^{\mathbf{E}}(Z_i) \setminus \text{Attr}^{\mathbf{E}}(Z_{i+1})$ (with the convention that $Z_{-1} = V$) and we note $rk(v) = i$.

We then have the following property.

PROPERTY 2.3 *For all $v \in (V \setminus (\text{Attr}^{\mathbf{E}}(Z_\infty)))$, one has:*

- if $v \in V_{\mathbf{A}}$, v has a successor v' such that $rk(v') \leq rk(v)$ and this inequality is strict iff $v \in F$;
- if $v \in V_{\mathbf{E}}$, for all successor v' of v one has $rk(v') \leq rk(v)$ and this inequality is strict iff $v \in F$.

Proof. Let $i = rk(v)$. As $v \notin \text{Attr}^{\mathbf{E}}(Z_{i+1})$, Adam can force to stay outside of $\text{Attr}^{\mathbf{E}}(Z_{i+1})$, i.e. to stay inside vertices with $rk \leq i$. This is how we define v' . If $v \in F$, assume that $rk(v') = rk(v) = i$ for all v' if $v \in V_{\mathbf{A}}$ or for some v' if $v \in V_{\mathbf{E}}$: then by definition we would have $v \in \text{Attr}_{\mathbf{E}}^+(Z_i) \cap F = Z_{i+1}$ thus $rk(v) = i + 1$. ■

We define a positional strategy ψ for Adam directly from Property 2.3: $\psi(v) = v'$ where $rk(v') \leq rk(v)$ for all $v \in (V \setminus (\text{Attr}^{\mathbf{E}}(Z_\infty))) \cap V_{\mathbf{A}}$. During a play where Adam respects ψ the value rk of the vertices that are visited decreases and this decreasing is strict whenever a final vertex is visited. As there is no infinitely decreasing sequence in the positive integers, we conclude that F is visited finitely often (more precisely, strategy ψ ensures that from a vertex v , there are at most $rk(v)$ visits to F).

2.3 Parity games

We fix a finite graph $G = (V, E)$ without dead-end, a partition $V_{\mathbf{E}} \cup V_{\mathbf{A}}$ of V that induces an arena \mathcal{G} . We take a finite set of colours $C = \{0, \dots, d\}$ and a colouring function $\rho : V \rightarrow C$. We denote by \mathbb{G} the induced parity game

We have the following result.

THEOREM 2.3 *One can compute the winning positions in \mathbb{G} and construct memoryless winning strategies for both players.*

For some player σ (Eve ou Adam), we denote by $\bar{\sigma}$ the other player.

2.3.1 Basic definitions

We start with some basic definitions, useful in the proof later.

DEFINITION 2.1 (Sub-arena) *Let $\mathcal{G} = (G, V_{\mathbf{E}}, V_{\mathbf{A}})$ be an arena with $G = (V, E)$, and let $U \subseteq V$. Let $col : V \rightarrow C$ be a colouring function on G . We consider the coloured sub-graph (one restrict col to U) of \mathcal{G} induced by U , $\mathcal{G}[U] = (G[U], V_{\mathbf{E}} \cap U, V_{\mathbf{A}} \cap U)$ où $G[U] = (U, E \cap (U \times U))$.*

We say that $\mathcal{G}[U]$ is a **sub-arena** iff it has no dead-end, i.e. for all $u \in U$, there is some $u' \in U$ such that $(u, u') \in E$.

FACT 2.1 A sub-arena of a sub-arena is a sub-arena.

DEFINITION 2.2 (Trap) Let σ be a player and let $\mathcal{G} = (G, V_{\mathbf{E}}, V_{\mathbf{A}})$ be an arena with $G = (V, E)$. A **trap for σ** (or σ -trap) in \mathcal{G} is a subset $U \subseteq V$ such that

- for all $v \in U \cap V_{\sigma}$, $(v, v') \in E \Rightarrow v' \in U$ (σ cannot exit U);
- for all $v \in U \cap V_{\bar{\sigma}}$, there exists $v' \in U$ tel que $(v, v') \in E$ ($\bar{\sigma}$ can stay in U);

FACT 2.2 The player $\bar{\sigma}$ has a positional strategy to keep the play in a σ -trap.

Proof. Staying in a trap is the dual as exiting the trap, which is a reachability condition. The result follows from the fact that reachability games admit positional strategies for both players. ■

FACT 2.3 Let U be a trap for σ in arena \mathcal{G} . Then $\mathcal{G}[U]$ is a sub-arena .

Proof. Immediate.

DEFINITION 2.3 (Attractor) Let σ be a player and let $\mathcal{G} = (G, V_{\mathbf{E}}, V_{\mathbf{A}})$ be an arena with $G = (V, E)$. The **attractor** of $U \subseteq V$ for σ (σ -attractor), denoted $\text{Attr}^{\sigma}(\mathcal{G}, U)$ (we omit \mathcal{G} if it is clear from the context), is defined as the limit of the following (increasing and bounded) sequence:

- $\text{Attr}_0^{\sigma}(U) = U$;
- $\forall i \geq 0, \text{Attr}_{i+1}^{\sigma}(U) = \text{Attr}_i^{\sigma}(U) \cup \{v \in V_{\sigma} \mid \exists v' \in \text{Attr}_i^{\sigma}(U) \text{ s.t. } (v, v') \in E\} \cup \{v \in V_{\bar{\sigma}} \mid (v, v') \in E \Rightarrow v' \in \text{Attr}_i^{\sigma}(U)\}$

FACT 2.4 The set $\text{Attr}^{\sigma}(\mathcal{G}, U)$ is the set of winning positions for σ in the reachability game to U .

The complement of an attractor for σ is a trap for σ .

The attractor for σ of a trap for $\bar{\sigma}$ is a trap for $\bar{\sigma}$

Proof. Direct consequences of the results on reachability games or immediate from the definitions.

2.3.2 Proof of Theorem 2.3

We now turn to the proof of Theorem 2.3. In addition of the announced result, we will prove that W_{σ} is a trap for $\bar{\sigma}$.

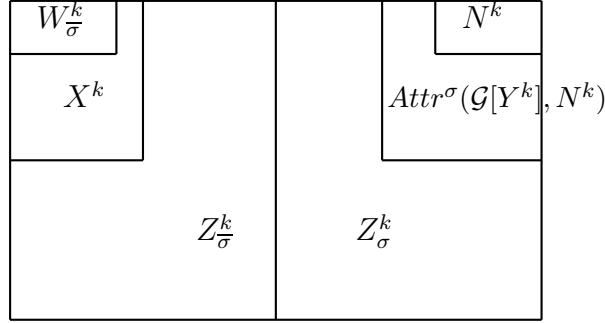
The proof is by induction on the number n of colours. The base case is the one where $n = 1$ and the result is immediate: depending on the parity of the unique colour, Eve (even) or Adam (odd) wins everywhere and any strategy is winning (in particular one can choose it positional).

We assume that we proved the result up to some $n - 1 \geq 1$ and we consider the case where we have n colours. For the rest of the proof, σ denotes the player that wins if the largest colour c is infinitely often visiter (i.e. Eve if c is even, Adam otherwise).

We construct a sequence of subsets of V , that we denote $(W_{\bar{\sigma}}^k)$ together with a sequence of positional strategies $\varphi_{\bar{\sigma}}^k$ for $\bar{\sigma}$ such that:

1. For all k , $W_{\bar{\sigma}}^k$ is a trap for σ and $\varphi_{\bar{\sigma}}^k$ is a winning strategy for $\bar{\sigma}$ in $W_{\bar{\sigma}}^k$. Moreover $\varphi_{\bar{\sigma}}^k$ traps the play in $W_{\bar{\sigma}}^k$.
2. The sequence $(W_{\bar{\sigma}}^k)_k$ is increasing and each $\varphi_{\bar{\sigma}}^{k+1}$ extend $\varphi_{\bar{\sigma}}^k$

We start by letting $W_{\bar{\sigma}}^0 = \emptyset$ and the properties are then trivially verified. Now assume that $W_{\bar{\sigma}}^k$ and $\varphi_{\bar{\sigma}}^k$ have been constructed and define $W_{\bar{\sigma}}^{k+1}$ and $\varphi_{\bar{\sigma}}^{k+1}$.



We start by letting $X^k = Attr^{\bar{\sigma}}(\mathcal{G}, W_{\bar{\sigma}}^k)$. It is a trap for σ as it is the attractor of a trap for σ (induction hypothesis).

The complement $Y^k = V \setminus X^k$ of X^k is a trap for $\bar{\sigma}$ as it is the complement of a $\bar{\sigma}$ -attractor. In particular it is a subarena that we denote by $\mathcal{G}[Y^k]$.

Consider now the set of vertices in Y^k that have colour c : $N^k = \{v \in Y^k \mid \rho(v) = c\}$ and we remove from Y^k the attractor for σ of N^k in the subarena:

$$Z^k = Y^k \setminus Attr^{\sigma}(\mathcal{G}[Y^k], N^k)$$

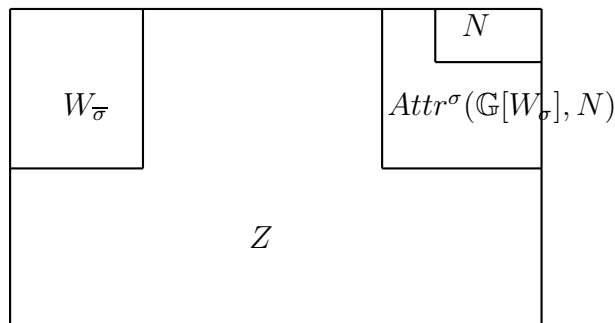
The set Z^k is a trap for σ (complement of a σ attractor) and defines a subarena (subarena of a sub-arena). Moreover all vertices in Z^k have a colour at most equal to $c - 1$.

Using the induction hypothesis, we define $Z_{\bar{\sigma}}^k$ and $Z_{\bar{\sigma}}^k$. Finally, we let $W_{\bar{\sigma}}^{k+1} = X^k \cup Z_{\bar{\sigma}}^k$, and we define $\varphi_{\bar{\sigma}}^{k+1}$ as the union of the (positional) strategies on these two disjoint sets. Clearly, such a strategy is winning on $W_{\bar{\sigma}}^{k+1}$ and extend $\varphi_{\bar{\sigma}}^{k+1}$.

The fact that $W_{\bar{\sigma}}^{k+1}$ is a trap for σ is immediate.

Now, we take for $W_{\bar{\sigma}}$ the limit of the previous sequence and we do the same to define $\varphi_{\bar{\sigma}}$. It is clear that points (1) and (2) are true for $\bar{\sigma}$.

We let $W_{\sigma} = V \setminus W_{\bar{\sigma}}$. The fact that W_{σ} is a trap for $\bar{\sigma}$ is a consequence of the fact that it is the complement of a $\bar{\sigma}$ -attractor as $W_{\bar{\sigma}} = Attr^{\bar{\sigma}}(W_{\bar{\sigma}})$. It remains to prove that σ has a winning positional strategy.



For this, we let $N = \{v \in W_\sigma \mid \rho(v) = c\}$ and

$$Z = W_\sigma \setminus \text{Attr}^\sigma(\mathcal{G}[W_\sigma], N)$$

By definition of $W_{\bar{\sigma}}$, $\bar{\sigma}$ cannot win in Z (otherwise $W_{\bar{\sigma}}$ would not be the limit of the sequence $W_{\bar{\sigma}}^k$). Thus σ has a positional strategy on Z (induction hypothesis). We define the strategy φ_σ as follows:

- Follow the positional winning strategy when being in Z .
- In $\text{Attr}^\sigma(\mathcal{G}[W_\sigma], N)$ follows the positional attractor strategy.
- In N stay in the winning region.

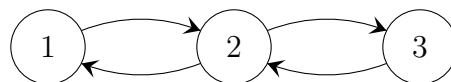
We easily check that this strategy is positional and winning. Indeed, in a play where σ follows this strategy, either the play eventually stay forever in Z (hence is winning for σ by induction hypothesis), or it visits infinitely often N and is therefore winning for σ as well.

2.4 Jeux de Muller

On se fixe un graphe non étiqueté $G = (V, E)$ sans cul-de-sac, une partition $V_E \cup V_A$ de V qui définit un graphe de jeu \mathcal{G} . On se donne un ensemble de sous-ensembles d'états $\mathcal{F} = \{F_1, \dots, F_i\}$, $F_i \subseteq V$ pour tout i . On appelle alors \mathbb{G} le **jeu de Muller** induit: Eve gagne une partie ssi l'ensemble des sommets visités infiniment souvent au cours de la partie est dans \mathcal{F} .

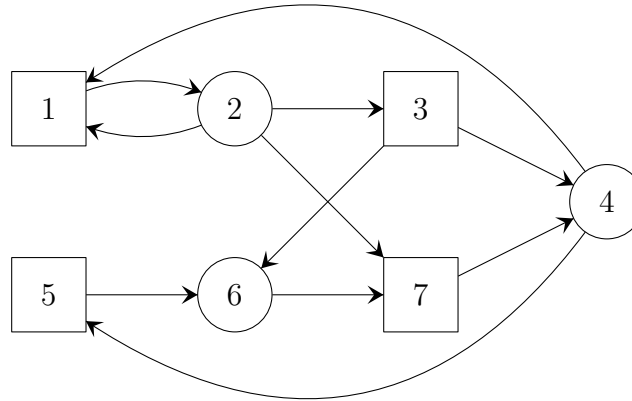
2.4.1 Les stratégies gagnantes peuvent requérir de la mémoire

On commence par un exemple très simple où il n'y a qu'un seul joueur. On considère le graphe suivant et on prend $\mathcal{F} = \{\{1, 2, 3\}\}$.



Le seul sommet où Eve a un choix est le sommet 2. Si elle joue avec une stratégie sans mémoire, on va visiter infiniment souvent soit $\{1, 2\}$ soit $\{2, 3\}$. Une stratégie gagnante avec mémoire consiste à jouer alternativement vers 1 et 3 depuis 2.

Pour un exemple plus explicite pour la suite, on considère maintenant le graphe de jeu suivant et on prend $\mathcal{F} = \{\{1, 2, 3, 4\}, \{1, 2, 3, 4, 5, 6, 7\}\}$.



A nouveau, il est facile de vérifier qu'une stratégie sans mémoire pour Eve ne peut être gagnante: depuis 2, Eve doit aller vers 3 (qui doit dans tous les cas être infiniment souvent visité), et si elle choisit d'aller depuis 4 vers 1, Adam peut jouer de sorte à ce que tous les sommets soient infiniment souvent visités sauf 5, et si elle choisit d'aller depuis 4 en 5, Adam peut jouer de sorte à ce que les sommets infiniment souvent visités soient $\{4, 5, 6, 7\}$.

Une stratégie gagnante avec mémoire pour Eve est la suivante:

- Depuis 2 toujours aller vers 3
- Depuis 4 si l'on vient de 3, aller vers 1.
- Depuis 4, si l'on vient de 7 via 3 et 6, aller vers 5 puis 6 puis 7 puis 4 puis 1.

En d'autres termes, Eve regarde en 4 quels sont les deux derniers états impairs visités:

- (1, 3) ou (5, 7): aller vers 1.
- (3, 7): aller vers 5

Il est à notre que cette information est facilement constructible par un automate fini et qu'une telle machine peut donc réaliser une stratégie gagnante pour Eve.

On généralise ces idées dans la suite.

2.4.2 Résolution des jeux de Muller: le last appearance record

On identifie dans la suite V avec $\{1, \dots, n\}$. On appelle **last appearance record (LAR)** une permutation de V avec un entier dans $\{1, \dots, n\}$ (le **hit**). On représentera cela par un n -uplet dont le j -ème élément est souligné si le hit vaut j , et on note $LAR(V)$ l'ensemble des LAR sur V . Le LAR est défini par induction pour toute suite (finie) de sommets (en particulier toute partie partielle) en posant: $LAR(\varepsilon) = (1, \dots, \underline{n})$ et en posant si $LAR(\lambda) = (v_1, \dots, v_n)$, $LAR(\lambda \cdot v) = (v_1, \dots, v_{j-1}, \underline{v_{j+1}}, v_{j+2}, \dots, v_n, v)$ si $v = v_j$: v est mis au fond et la position qu'il occupait devient le hit. Ainsi $LAR(\lambda \cdot v)$ ne dépend que de v et de $LAR(\lambda)$.

Il est facile de voir que l'ensemble des sommets infiniment souvent répétés est F au cours d'une partie **ssi** au bout d'un certain moment

1. le LAR a toujours un hit supérieur ou égal à $n - |F| + 1$;
2. infiniment souvent le hit vaut $n - |F| + 1$;
3. les $|F|$ derniers sommets du LAR forment une permutation de F .

Preuve. Supposons que l'on ait 1, 2 et 3 vraies à partir d'un certain moment. Il est clair que l'ensemble infiniment souvent visité est F (à partir d'un moment les sommets visités sont toujours dans les $|F|$ derniers, et ils sont donc dans F ; de plus on n'en laisse pas de côté grâce au point 2).

On considère une partie $\lambda = u_1 u_2 \dots$ de sommets infiniment souvent visités F . Il existe un indice k tel que l'on ne voit plus que des éléments de F après u_k , et il existe $h > k$ tel que $F = \{u_k, \dots, u_h\}$. Par définition du LAR, il est clair que les $|F|$ dernier sommets du LAR en u_h forment une permutation de F . On montre par récurrence que le LAR vérifie les trois propriétés voulues pour v_l avec $l \geq h + 1$. Pour $l = h + 1$, comme $u_l \in F$ et comme les sommets de F apparaissent dans les $|F|$ dernières positions du LAR en u_h , le hit en u_l est supérieur ou égal à $n - |F| + 1$ et de plus on a toujours une permutation de F en queue du LAR. En raisonnant de la même façon on établit donc 1 et 3. Supposons que l'on n'ai pas 2: il existe $m > h$ tel qu'après m le hit soit toujours plus grand que $n + 1 - |F|$: ainsi le $(n + 1 - |F|)$ -ième sommet du LAR en u_m n'est plus jamais visité, et n'est donc que finiment souvent visité alors qu'il appartient à F , ce qui est contradictoire. ■

On considère un nouveau graphe de jeu dans lequel les sommets vont en plus contenir une information sur le LAR. Plus formellement, on considère le graphe $G' = (V \times LAR(V), E')$ où $((v, \tau), (v', \tau')) \in E'$ ssi $(v, v') \in E$ et τ' est le LAR obtenu en allant vers v' avec le LAR τ . On considère la partition de V' donnée par $V'_E = V_E \times LAR(V)$, et on appelle \mathcal{G}' le graphe de jeu associé. Enfin, on définit une condition de parité pour \mathcal{G}' : un sommet (v, ℓ) tel que ℓ a pour hit j , a pour couleur $2j$ si les j derniers éléments de ℓ forment une permutation d'un ensemble de \mathcal{F} et a pour couleur $2j + 1$ si les j derniers éléments de ℓ ne forment pas une permutation d'un ensemble de \mathcal{F} .

Appelons \mathbb{G}' le jeu défini sur \mathcal{G}' équipé de la nouvelle condition de parité que l'on vient de définir. On a alors le résultat suivant qui permet de conclure:

LEMMA 2.1 *Un sommet v est gagnant pour Eve dans \mathbb{G} ssi le sommet $(v, LAR(\varepsilon))$ est gagnant pour Eve dans \mathbb{G}'*

Preuve. Considérons une partie $\lambda = v_0 v_1 \dots$ dans \mathbb{G} . On considère alors la partie $\lambda' = v'_0 v'_1 \dots$ dans \mathbb{G} définie par $v'_i = (v_i, LAR(v_0 \dots v_{i-1})) \dots$ pour tout i . Il est alors facile de voir que λ est gagnante pour Eve dans \mathbb{G} ssi λ' est gagnante pour Eve dans \mathbb{G}' . Ceci vient de la caractérisation donnée des sommets infiniment souvent répétés en terme de LAR, et de la définition des couleurs.

On note τ la fonction (bijection en fait) telle que $\tau(\lambda) = \lambda'$ et π la fonction inverse qui projette λ' en λ , et on considère les versions naturelles de ces fonctions définie sur les parties partielles. Maintenant, si φ' est une stratégie d'Eve dans \mathbb{G}' on définit la stratégie φ pour Eve dans \mathbb{G} en posant $\varphi(\lambda) = last(\pi(\tau(\lambda) \cdot \varphi'(\tau(\lambda))))$, où $last$ est la fonction qui associe à une partie son dernier sommets. Il est alors facile de vérifier que φ est gagnante depuis v si φ' est gagnante depuis $(v, LAR(\varepsilon))$: en effet toute partie λ dans \mathbb{G} où Eve respecte φ est telle que $\tau(\lambda)$ est une partie dans laquelle Eve respecte la stratégie (gagnante) φ' ; comme les parties gagnantes dans \mathbb{G} sont exactement celles dont l'image par τ est une partie gagnante dans \mathbb{G}' , cela termine l'argument.

Comme cette construction peut être également faite pour une stratégie gagnante d'Adam, cela conclut la preuve. ■

On a donc en résumé le théorème suivant:

THEOREM 2.4 *On peut calculer l'ensemble des positions gagnantes dans un jeu de Muller ainsi que des stratégies gagnantes qui nécessitent une mémoire de taille $k.k!$ où k est le nombre de sommets du graphe de jeu considéré.*

Preuve. La première partie du résultat est une conséquence directe des deux lemmes précédents. La seconde partie (taille de la mémoire) est laissée en exercice. ■

Jeux et automates d'arbres

Contents

3.1 Définitions	19
3.1.1 Rappel: automates de mots	19
3.1.2 Automates d'arbres	20
3.2 Le problème du vide	23
3.3 Lemme de Rabin	25

L'objet de ce chapitre est l'étude des automates acceptants des arbres infinis. Dans une première partie, on donnera les définitions de bases et quelques exemples. Dans une seconde partie on montrera que le problème du test du vide est intimement lié (polynomialement équivalent) au problème de décision du gagnant dans un jeu de parité sur un graphe fini. On terminera enfin en donnant une preuve du théorème de complémentation des langages réguliers d'arbres infinis (lemme de Rabin).

3.1 Définitions

3.1.1 Rappel: automates de mots

DEFINITION 3.1 *Un Automate de mots est un quintuplé $\mathcal{A} = (Q, A, I, \Delta, Acc)$ où Q est un ensemble fini d'états de contrôle, A est un alphabet fini d'entrée, $I \subseteq Q$ est un ensemble d'états initiaux, $\Delta \subseteq Q \times A \times Q$ est une relation de transition et Acc est une condition d'acceptation d'un des types suivants:*

- *$Acc = F \subseteq Q$ est un ensemble d'états finaux, et l'on parle alors de condition d'acceptation par état final.*
- *$Acc = R \subseteq Q$ est un ensemble d'états répétés, et l'on parle alors de condition de Büchi.*
- *$Acc = \rho : Q \rightarrow \mathbb{N}$ est une fonction de coloriage, et l'on parle alors de condition de parité.*

Un calcul de \mathcal{A} sur un mot (potentiellement infini) $u = a_0a_1 \dots$ est une suite $q_0q_1 \dots$ telle que

- *$q_0 \in I$ est un état initial;*

- pour tout $i \leq |u|$, $(q_{i-1}, a_{i-1}, q_i) \in \Delta$.

Si \mathcal{A} est muni d'une condition d'acceptation par état final, un mot fini u est accepté par \mathcal{A} si et seulement s'il existe un calcul de \mathcal{A} sur u qui se termine dans un état final.

Si \mathcal{A} est muni d'une condition de Büchi, un mot infini u est accepté par \mathcal{A} si et seulement s'il existe un calcul $q_0q_1q_2 \dots$ de \mathcal{A} qui visite infiniment souvent R , i.e. tel que $|\{i \mid q_i \in R\}| = \infty$.

Enfin, si \mathcal{A} est muni d'une condition de parité, un mot infini u est accepté par \mathcal{A} si et seulement s'il existe un calcul $q_0q_1q_2 \dots$ de \mathcal{A} tel que $\liminf(\rho(q_i))_{i \geq 0}$ est paire.

Le langage reconnu (ou accepté) par un automate est l'ensemble des mots sur lesquels il existe un calcul acceptant de l'automate. On note généralement $L(\mathcal{A})$ le langage accepté par \mathcal{A} .

Enfin, un automate est dit **déterministe** si Δ est fonctionnelle, i.e. pour tout $(q, a) \in Q \times A$, $|\{q' \mid (q, a, q') \in \Delta\}| \leq 1$ et si I est un singleton.

Les résultats suivants sont des (grands) classiques:

THEOREM 3.1 *Sur les mots finis, les modèles déterministe et non-déterministe reconnaissent les mêmes langages.*

THEOREM 3.2 *Sur les mots infinis, les automates déterministes de Büchi sont strictement moins expressifs que les automates non-déterministes de Büchi. Par contre, les automates déterministes de parités, les automates non-déterministes de parité et les automates non-déterministes de Büchi ont le même pouvoir d'expression.*

Les **langages réguliers de mots finis** sont les langages acceptés par des automates finis. Les **langages réguliers de mots infinis** sont les langages acceptés par des automates finis déterministes de parité (de façon équivalente non-déterministes de Büchi/parité).

Concernant les propriétés de clôture, on a les résultats (classiques) suivants.

THEOREM 3.3 *La classe des langages réguliers de mots finis est une algèbre de Boole. La classe des langages réguliers de mots infinis est une algèbre de Boole.*

3.1.2 Automates d'arbres

DEFINITION 3.2 *Un arbre binaire fini sur un alphabet A est une fonction $t : D \rightarrow A$ où $D \subseteq \{0, 1\}^*$ est un sous-ensemble clos par préfixe appelé **domaine**. De plus on demandera que pour tout $u \in D$, $u0 \in D \Leftrightarrow u1 \in D$ (i.e. un nœud a deux fils ou est une feuille). Un nœud u de t est un élément de D et son étiquette est $t(u)$. La racine est le nœud ε (où ε est le mot vide).*

DEFINITION 3.3 *Un automate d'arbres finis est un quintuplé $\mathcal{A} = (Q, A, I, \Delta, Acc)$ où Q est un ensemble fini d'états de contrôle, A est un alphabet fini d'entrée, $I \subseteq Q$ est un ensemble d'états initiaux, $\Delta \subseteq Q \times A \times Q \times Q$ est une relation de transition et $Acc \subseteq Q \times A$ est une condition d'acceptation.*

Un calcul de \mathcal{A} sur un arbre fini t d'alphabet A et de domaine D est un arbre fini t' d'alphabet $Q \times A$ et de domaine D tel que

- pour tout $u \in D$, $t'(u) = (q, t(u))$ pour un certain $q \in Q$ (i.e. un calcul est une Q -annotation de l'arbre en entrée);
- $t'(\varepsilon) = (q_i, t(\varepsilon))$ avec $q_i \in I$;

- pour tout $u \in D$, si $u0, u1 \in D$ alors $t'(u0) = (q_0, t(u0))$ et $t'(u1) = (q_1, t(u1))$ avec $(q, t(u), q_0, q_1) \in \Delta$.

Un arbre fini t est accepté par \mathcal{A} si et seulement s'il existe un calcul t' de \mathcal{A} sur t tel que toutes les feuilles de t' sont étiquetées par des éléments de Acc .

Le langage reconnu (ou accepté) par un automate est l'ensemble des arbres finis sur lesquels il existe un calcul acceptant de l'automate. On note généralement $L(\mathcal{A})$ le langage accepté par \mathcal{A} .

Enfin, un automate est dit **déterministe** si Δ est fonctionnelle, i.e. pour tout $(q, a) \in Q \times A$, $|\{(q_0, q_1) \mid (q, a, q_0, q_1) \in \Delta\}| \leq 1$ et si I est un singleton.

EXERCISE 3.1 Donner un automate reconnaissant l'ensemble des arbres possédant une branche faite uniquement de a . Montrer qu'un automate déterministe ne peut pas reconnaître ce langage.

EXERCISE 3.2 Soit L un langage régulier de mots finis sur un alphabet A . La frontière d'un arbre est le mot obtenu en lisant les étiquettes des feuilles de la gauche vers la droite. Montrer que le langage des arbres dont la frontière est dans L est reconnu par un automate d'arbres non-déterministe.

On a le résultat suivant.

PROPOSITION 3.1 La classe des langages réguliers d'arbres finis est une algèbre de Boole.

Preuve. Les clôtures par intersection et unions sont triviales. La clôture par complément utilise un modèle équivalent d'automates d'arbre qui évalue l'arbre des feuilles vers la racine. Ce dernier modèle pouvant être facilement déterminisé, on obtient alors la clôture par complément. ■

Concernant le modèle déterministe, on a le résultat suivant.

PROPOSITION 3.2 Les automates d'arbres finis déterministes sont strictement moins expressifs que les automates d'arbres finis non-déterministes.

Preuve. Il suffit, par exemple, de considérer le langage des arbres qui contiennent exactement une feuille étiquetée par a . ■

DEFINITION 3.4 Un **arbre binaire infini** sur un alphabet A est une fonction $t : \{0, 1\}^* \rightarrow A$. Un nœud u de t est un élément de $\{0, 1\}^*$ et son étiquette est $t(u)$. La racine est le nœud ε .

DEFINITION 3.5 Un **automate d'arbres infinis** est un quintuplé $\mathcal{A} = (Q, A, I, \Delta, Acc)$ où Q est un ensemble fini d'états de contrôle, A est un alphabet fini d'entrée, $I \subseteq Q$ est un ensemble d'états initiaux, $\delta : Q \times A \times Q \times Q$ est une relation de transition et Acc est une condition d'acceptation d'un des types suivants:

- $Acc = F \subseteq Q$ est un ensemble d'états répétés, et l'on parle alors de condition de **Büchi**.
- $Acc = \rho : Q \rightarrow \mathbb{N}$ est une fonction de coloriage, et l'on parle alors de condition de **parité**.

Un calcul de \mathcal{A} sur un arbre infini t d'alphabet A est un arbre infini t' d'alphabet $Q \times A$ tel que

- pour tout $u \in D$, $t'(u) = (q, t(u))$ pour un certain $q \in Q$ (i.e. un calcul est une Q -annotation de l'arbre en entrée);

- $t'(\varepsilon) = (q_i, t\varepsilon)$ avec $q_i \in I$;
- pour tout $u \in D$, $t'(u0) = (q_0, t(u0))$ et $t'(u1) = (q_1, t(u1))$ avec $(q_0, q_1) \in \Delta(q, t(u))$.

Étant donnée une branche d'un calcul t' , c'est à dire un mot infini $\alpha = \alpha_1\alpha_2\alpha_3 \cdots \in \{0, 1\}^\omega$ on lui associe de façon naturelle une suite d'états $\tau(\alpha) = q_0q_1q_2 \cdots$ où $(q_i, a_i) = t'(\alpha_1 \cdots \alpha_i)$. La branche α est acceptante ssi:

- Acc est une condition de Büchi et $\tau(\alpha)$ visite infiniment souvent R , i.e. $|\{i \mid q_i \in R\}| = \infty$;
- ou Acc est une condition de parité ρ et $\liminf(\rho(q_i))_{i \geq 0}$ est paire.

Un calcul t' est acceptant si toutes ses branches sont acceptantes.

Un arbre infini t est accepté par \mathcal{A} si et seulement s'il existe un calcul acceptant de \mathcal{A} sur t .

Le langage reconnu (ou accepté) par un automate est l'ensemble des arbres infinis sur lesquels il existe un calcul acceptant de l'automate. On note généralement $L(\mathcal{A})$ le langage accepté par \mathcal{A} .

Enfin, un automate est dit **déterministe** si Δ est fonctionnelle, i.e. pour tout $(q, a) \in Q \times A$, $|\{(q_0, q_1) \mid (q, a, q_0, q_1) \in \Delta\}| \leq 1$ et si I est un singleton.

EXERCISE 3.3 Montrer que les automates d'arbres de Büchi déterministes sont strictement moins expressifs que leur variante non-déterministe.

EXERCISE 3.4 Donner un automate de Büchi reconnaissant les arbres possédant une branche qui contient une infinité de b . Montrer que ce langage n'est pas reconnaissable par un automate déterministe de parité.

EXERCISE 3.5 Donner un automate déterministe de parité reconnaissant les arbres dont chaque branche possède un nombre fini de b . Montrer que ce langage ne peut pas être reconnu par un automate de Büchi (même non-déterministe).

■

Les exercices précédents permettent d'établir les résultats suivants.

PROPOSITION 3.3 On a:

1. Les automates d'arbres déterministes de Büchi sont strictement moins expressifs que les automates d'arbres non-déterministes de Büchi.
2. Les automates d'arbres non-déterministes de Büchi sont strictement moins expressifs que les automates d'arbres non-déterministes de Parité.
3. Les automates d'arbres déterministes de Büchi sont strictement moins expressifs que les automates d'arbres déterministes de Parité.
4. Les automates de Büchi non-déterministes et les automates de parité déterministes sont incomparables.

La classe des **langages réguliers d'arbres infinis** est celle des langages reconnus par des automates non-déterministes de parité.

On a alors le résultat (dur) suivant.

THEOREM 3.4 La classe des langages réguliers d'arbres infinis est une algèbre de Boole.

3.2 Le problème du vide

Dans ce qui suit on s'intéresse au problème du vide pour les automates d'arbres infinis de parité: étant donné un automate \mathcal{A} a-t-on $L(\mathcal{A}) = \emptyset$? La solution de ce problème se transpose alors facilement à toutes les sous-classes considérées ci-dessous (que ce soit en terme de déterminisme, de condition d'acceptation ou de modèle — mots finis, mots infinis ou arbres finis).

On se fixe un automate de parité $\mathcal{A} = (Q, A, I, \Delta, \rho)$. Dans ce qui suit on construit un jeu de parité \mathbb{G} sur un graphe fini \mathcal{G} tel qu'Eve remporte \mathbb{G} (pour un sommet d'un type particulier) si et seulement si $L(\mathcal{A}) \neq \emptyset$. Dans ce jeu Eve va décrire un calcul de l'automate et Adam va le vérifier. Plus précisément, Eve commence par donner un état initial. Ensuite, elle donne l'étiquette de la racine et choisit la transition que l'automate effectue à la racine et Adam choisit quel sous-arbre du calcul il souhaite vérifier (comme Eve doit gagner dans tous les cas le calcul devra être bon des deux côtés...). Eve doit alors choisir l'étiquette du nœud choisi par Adam ainsi que la transition. Adam choisit alors quel côté vérifier et ainsi de suite.

Plus formellement, le graphe de jeu est le suivant:

- $V_E = Q$ et $V_A = \Delta$
- depuis un sommet q , il y a une transition vers tout sommet de la forme $(q, a, q_0, q_1) \in \Delta$.
- depuis un sommet (q, a, q_0, q_1) , il y a deux transitions une vers q_0 et une vers q_1 .

Seuls les sommets de Q sont colorés et leur couleur est donnée par ρ (on peut toujours colorer les autres sommets en leur attribuant la couleur maximale, ce qui n'affectera pas le gagnant d'une partie).

On a alors le résultat suivant:

THEOREM 3.5 *Le langage $L(\mathcal{A})$ est non vide si et seulement si Eve possède une stratégie gagnante dans \mathbb{G} depuis un sommet de I .*

Preuve. Supposons $L(\mathcal{A})$ non vide et considérons t un arbre dans le langage. Soit t' un calcul acceptant de \mathcal{A} sur t . On définit une stratégie φ pour Eve dans le jeu \mathbb{G} qui est gagnante depuis le sommet $q \in I$ où l'on note $(q, a) = t'(\varepsilon)$. La stratégie φ utilise une mémoire qui est un nœud de t' , c'est à dire un mot $u \in \{0, 1\}^*$. Au départ $u = \varepsilon$ et à tout moment on aura que la partie est dans un sommet q si et seulement si $t'(u) = (q, a)$ pour un certain a (la propriété sera vérifiée trivialement par induction). Depuis un sommet q , Eve joue vers (q, a, q_0, q_1) , où $t'(u) = (q, a)$ et où $t'(u \cdot 0) = (q_0, a_0)$ et $t'(u \cdot 1) = (q_1, a_1)$. La mémoire u est mise à jour après chaque coup d'Adam: si ce dernier va de (q, a, q_0, q_1) vers q_0 , la mémoire devient $u \cdot 0$, et sinon la mémoire devient $u \cdot 1$.

Considérons maintenant une partie λ dans le jeu \mathbb{G} au cours de laquelle Eve respecte φ . Tout d'abord les choix d'Adam (0 ou 1: 0 s'il choisit q_0 et 1 s'il choisit q_1) au cours de λ permettent de lui associer de façon naturelle une branche infinie β de t' (c'est à dire un mot de $\{0, 1\}^\omega$). Par ailleurs, les choix d'Eve permettent d'associer à λ une suite infinie α d'éléments de $Q \times A$ qui correspond exactement à la suite des étiquettes des éléments de β dans t' . Dès lors la plus petite couleur des états qui apparaissent infiniment dans α est la même que la plus petite couleur apparaissant infiniment souvent sur la branche β et est donc paire car t' est acceptant. La stratégie φ est donc gagnante pour Eve.

Réciproquement, considérons une stratégie gagnante φ pour Eve depuis un sommet $q_i \in I$, et construisons un arbre t et un calcul associé t' acceptant. On construit t' (et donc t) en définissant $t'(u)$ par induction sur u . Pour $u = \varepsilon$, $t'(u) = (q_i, a)$, où $\varphi(q_i) = (q_i, a, q_0, q_1)$. Supposons $t'(u) = (q_n, a_n)$ défini et notons $u = u_1 u_2 \cdots u_n$. On considère

$$\lambda = q_0 \cdot (q_0, a_0, q_0^g, q_0^d) \cdot q_1 \cdot (q_1, a_1, q_1^g, q_1^d) q_2 \cdots q_n$$

une partie débutant en $q_0 = q_i$, où Eve respecte φ et où $q_{i+1} = q_i^g$ si $u_{i+1} = 0$ et $q_{i+1} = q_i^d$ si $u_{i+1} = 1$. On note

- $(q_n, a_n, q_n^g, q_n^d) = \varphi(\lambda)$;
- $(q_n^g, a_{n+1}^g, q_{n+1}^{g,g}, q_{n+1}^{g,d}) = \varphi(\lambda \cdot (q_n, a_n, q_n^g, q_n^d) \cdot q_n^g)$;
- $(q_n^d, a_{n+1}^d, q_{n+1}^{d,g}, q_{n+1}^{d,d}) = \varphi(\lambda \cdot (q_n, a_n, q_n^g, q_n^d) \cdot q_n^d)$.

et on pose

$$t'(u0) = (q_n^g, a_{n+1}^g) \quad \text{et} \quad t'(u1) = (q_n^d, a_{n+1}^d)$$

On a ainsi défini de façon unique t' . De plus toute branche de t' est associée à une partie infinie dans \mathbb{G} où Eve respecte φ en considérant simplement la construction inductive précédente. En particulier, la suite des couleurs associée à une branche est la même que la suite des couleurs dans la partie associée à la branche. Comme cette dernière partie est remportée par Eve, la branche vérifie donc la condition de parité. Cette propriété étant vraie pour toutes les branches on conclut que t' est un calcul acceptant et que le langage $L(\mathcal{A})$ est donc non vide. ■

La construction précédente montre que l'on peut réduire polynomialement le problème du vide pour les automates d'arbres infinis de parité au problème de décision du gagnant dans un jeu de parité sur un graphe fini. En fait la construction marche de la même façon dans l'autre sens: on peut réduire polynomialement le problème de décision du gagnant dans un jeu de parité sur un graphe fini au problème du vide pour les automates d'arbres infinis de parité.

Pour cela, on considère un graphe de jeu $\mathcal{G} = (V_E, V_A, E)$ et une fonction de coloriage dessus que l'on note ρ . On suppose de plus que chaque sommet v de \mathcal{G} possède exactement deux successeurs que l'on notera $\tau_0(v)$ et $\tau_1(v)$. On appelle \mathbb{G} le jeu de parité associé, et on se donne un sommet initial v_i dans V .

On définit l'automate d'arbres infinis déterministe de parité suivant: $\mathcal{A} = (Q, A, I, \Delta, \rho)$ où

- $Q = V \cup \{f\}$, où f sera un état dans lequel on boucle et où l'on accepte tout arbre;
- $A = \{0, 1, \perp\}$;
- $I = \{v_i\}$;
- ρ est étendue à f en posant $\rho(f) = 0$
- Δ est définie par:
 - $(f, x, f, f) \in \Delta$ pour tout $x \in A$;
 - $(v, 0, \tau_0(v), f) \in \Delta$ pour tout $v \in V_E$

- $(v, 1, f, \tau_1(v)) \in \Delta$ pour tout $v \in V_E$
- $(v, \perp, \tau_0(v), \tau_1(v))$ pour tout $v \in V_A$

On a alors le résultat escompté:

THEOREM 3.6 *Eve gagne dans \mathbb{G} depuis v_i si et seulement si $L(\mathcal{A}) \neq \emptyset$.*

Preuve. $L(\mathcal{A}) \neq \emptyset$ ssi Eve gagne depuis v_i dans le jeu \mathbb{G}' donné par le Théorème 3.5. On construit donc \mathbb{G}' et (miracle :-)) on se rend compte qu'on retrouve (à quelques sommets intermédiaires inutiles) le jeu \mathbb{G} , ce qui termine la preuve. ■

3.3 Lemme de Rabin

On prouve enfin le résultat suivant dû à Rabin:

LEMMA 3.1 (Rabin'72) *La classe des langages réguliers d'arbres infinis est fermée par complémentaire.*

On se fixe pour le reste de la preuve un automate à parité $\mathcal{A} = (Q, A, I, \Delta, \rho)$ que l'on va compléter. On peut toujours se ramener au cas où I est un singleton $\{q_i\}$.

On considère un arbre t et on lui associe un jeu $\mathbb{G}_{\mathcal{A},t}$ dans lequel Eve gagne ssi $t \in L(\mathcal{A})$. Le jeu $\mathbb{G}_{\mathcal{A},t}$ est une variante du jeu \mathbb{G} que l'on avait utilisé pour le test du vide: la différence est qu'Eve ne choisit plus t mais se contente de choisir les transitions de l'automates \mathcal{A} . Dans ce jeu, Eve choisit une transition à appliquer à la racine, puis Adam choisit de vérifier le fils gauche ou le fils droit. Eve choisit alors la nouvelle transition puis Adam la direction et ainsi de suite.

Ainsi, les sommets d'Eve dans ce jeu sont des triplets de la forme $(u, t(u), q)$ où u est un nœud et $q \in Q$, et les sommets d'Adam sont de la forme $(u, t(u), (q, t(u), q_0, q_1))$ où $(q, t(u), q_0, q_1) \in \Delta$. Depuis un sommet $(u, t(u), q)$ Eve peut aller dans n'importe quel sommet $(u, t(u), (q, t(u), q_0, q_1))$ et depuis un sommet $(u, t(u), (q, t(u), q_0, q_1))$, Adam peut aller soit en $(u0, t(u0), q_0)$ soit en $(u1, t(u1), q_1)$. Seuls les sommets d'Eve sont colorés et leur couleur est donnée par la valeur de ρ sur leur troisième composante. Il est alors facile de voir que l'on a le résultat suivant.

PROPOSITION 3.4 *Eve gagne dans $\mathbb{G}_{\mathcal{A},t}$ depuis $(\varepsilon, t(\varepsilon), q_i)$ ssi $t \in L(\mathcal{A})$.*

Ainsi, compléter un automate d'arbres infinis revient à montrer que l'on peut traduire la non-existence d'une stratégie gagnante pour Eve dans un jeu $\mathbb{G}_{\mathcal{A},t}$ comme l'existence d'une stratégie pour Eve dans un jeu $\mathbb{G}_{\mathcal{B},t}$ où \mathcal{B} ne dépend que de \mathcal{A} et surtout pas de t .

La preuve se fait en deux étapes:

1. On montre que si Eve n'a pas de stratégie gagnante dans $\mathbb{G}_{\mathcal{A},t}$ alors Adam en a une. De plus on montre que cette stratégie peut être choisie sans mémoire.
2. On convertit une stratégie pour Adam en une stratégie pour Eve dans un nouveau jeu

Le premier point revient à dire que les jeux de parité sur des graphes infinis sont positionnellement déterminés. Nous avons établi dans le premier chapitre de ce cours ce résultat pour les graphes finis et il s'étend sans problème au cadre infini (modulo une induction transfinie). On se concentre donc sur la seconde partie de la preuve.

On a l'équivalence suivante: $t \notin L(\mathcal{A})$ ssi Adam possède une stratégie gagnante sans mémoire dans $\mathbb{G}_{\mathcal{A},t}$ depuis $(\varepsilon, t(\varepsilon), q_i)$. Une telle stratégie est une fonction $\varphi : \{0, 1\}^* \times A \times \Delta \rightarrow \{0, 1\}$. On peut décomposer φ en $(\varphi_w)_{w \in \{0,1\}^*}$ avec $\varphi_w : A \times \Delta \rightarrow \{0, 1\}$. L'ensemble $I = A \times \Delta \rightarrow \{0, 1\}$ des instructions locales est fini, et on peut donc voir φ comme un arbre I -étiqueté s avec $s(w) = \varphi_w$ pour tout $w \in \{0, 1\}^*$. Appelons $s \odot t$ l'arbre $(I \times A)$ -étiqueté défini par $s \odot t(w) = (s(w), t(w))$.

Ainsi $t \notin L(\mathcal{A})$ équivaut à

Il existe un arbre I -étiqueté s tel que pour toute suite $\tau = \tau_0 \tau_1 \dots \in \Delta^\omega$ de transitions choisies par Eve et pour toute (en fait pour l'unique) branche $\beta \in \{0, 1\}^\omega$ déterminée par τ via la stratégie codée dans s , la suite d'états associée viole la condition de parité.

On introduit la notation suivante pour tout arbre C -étiqueté T et tout mot infini $\beta = \beta_1 \beta_2 \dots \in \{0, 1\}^\omega$: on note $T|\beta \in C^\omega$ le mot $c_0 c_1 c_2 \dots$ où $c_i = T(\beta_1 \dots \beta_i)$. Ainsi, $T|\beta$ est le mot lu le long de la branche β dans l'arbre T .

On peut alors reformuler la propriété précédente:

(1) Il existe un arbre I -étiqueté s tel que $s \odot t$ satisfasse:

(2) pour toute branche $\beta \in \{0, 1\}^\omega$,

(3) pour toute suite $\tau = \tau_0 \tau_1 \dots \in \Delta^\omega$,

(4) si le mot $s|\beta \in (I = A \times \Delta \rightarrow \{0, 1\})^\omega$ appliqué au mot $t|\beta \in A^\omega$ et au mot $\tau \in \Delta^\omega$ produit β alors la suite d'états induite par τ ne vérifie pas la condition de parité.

La propriété (4) est une propriété de mots infinis sur l'alphabet $I \times A \times \Delta \times \{0, 1\}$ qui peut être trivialement testée par un automate déterministe de parité. Ce dernier est de plus indépendant de t . La propriété (3) est une propriété de mots infinis sur l'alphabet $I \times A \times \{0, 1\}$ obtenue par une quantification universelle sur la propriété (4), et qui est donc reconnaissable par un automate déterministe de parité (via la déterminisation de ces automates). La propriété (2) est une propriété des arbres $(I \times A)$ -étiquetés qui peut être testée par un automate d'arbre déterministe de parité qui teste la propriété (3) sur chaque branche (ceci est possible car on peut tester de façon déterministe la propriété (3)). Enfin, la propriété (1) peut être décidée par un automate d'arbres non déterministe de parité \mathcal{B} qui devine s , construit $s \odot t$ à la volée et simule le test de la propriété (2) dessus.

Comme la propriété (4) est indépendante de t , il en va de même de \mathcal{B} . Enfin, \mathcal{B} accepte un arbre t ssi $t \notin L(\mathcal{A})$, ce qui conclut la preuve.

Contents

1	Two-player games on finite graphs: basic definitions	3
1.1	Graph, arena, game	3
1.2	Winning conditions	4
1.3	Strategies et winning positions	6
1.3.1	Winning positions	7
1.3.2	Some specific winning strategies	7
2	Two-player games on finite graphs: ω-regular winning conditions	9
2.1	Reachability games	9
2.2	Büchi Games	11
2.3	Parity games	12
2.3.1	Basic definitions	12
2.3.2	Proof of Theorem 2.3	13
2.4	Jeux de Muller	15
2.4.1	Les stratégies gagnantes peuvent requérir de la mémoire	15
2.4.2	Résolution des jeux de Muller: le last appearance record	16
3	Jeux et automates d'arbres	19
3.1	Définitions	19
3.1.1	Rappel: automates de mots	19
3.1.2	Automates d'arbres	20
3.2	Le problème du vide	23
3.3	Lemme de Rabin	25