Games and logic (Parity games)

Igor Walukiewicz

CNRS, LaBRI Bordeaux

Journée "Théorie des Jeux et Informatique"
February 2009

Plan

Plan

- Parity games from verification problems.
- Parity games from the satisfiability problem.
- Properties of parity games (memoryless strategies).
- Some extensions.

Omitted:

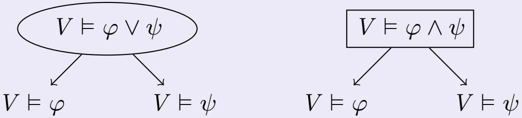
- Ehrenfeucht-Fraïssé games
- Wedge games

Games from verification

PROPOSITIONAL FORMULAS WITHOUT NEGATION OPERATION

$$P \mid \neg P \mid \varphi \vee \psi \mid \varphi \wedge \psi$$

CHECKING IF φ IS SATISFIED IN A VALUATION $V : \mathrm{Prop} \to \{0,1\}$

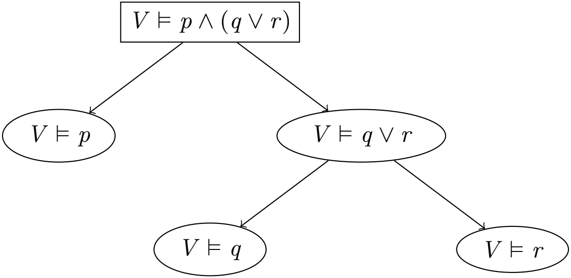


$V \vDash P$ Eve wins if $V(P) = 1$

$V \vDash \neg P$ Eve wins if $V(P) = 0$

FACT

Eve has a winning strategy from ($V \vDash \varphi$) iff φ is true in V


$V \models p \wedge (q \vee r)$
$V \models p$
$V \models q \vee r$
$V \models q$
$V \models r$

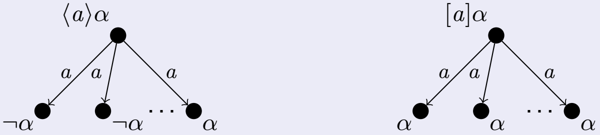
Models

Transition systems: graph with labelled edges.
In each node there is a valuation of propositions.

Modal logic

$$P \mid \neg P \mid \alpha \vee \beta \mid \alpha \wedge \beta \mid \langle a \rangle \alpha \mid [a]\alpha$$

Semantics



Verification as a game

Verification (Model Checking)

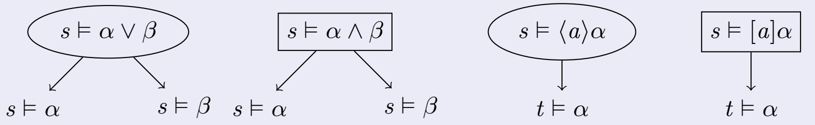
Given a transition system $\mathcal{M}$ and a property α, check if $\mathcal{M} \vDash \alpha$.

Reformulation

Construct a game $G(\mathcal{M}, \alpha)$ of two players: Adam and Eve.
Fix the rules in such a way that

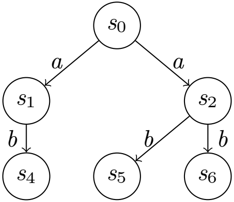
$$\text{Eve wins from the initial position of } G(\mathcal{M}, \alpha) \text{ iff } \mathcal{M} \vDash \alpha$$

Game rules



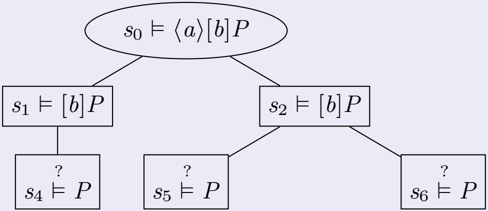
where $s \xrightarrow{a} t$

Example

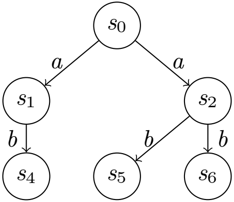


$\overset{?}{\models} \langle a \rangle [b] P$

Game

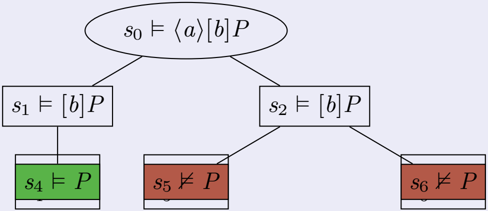


Example

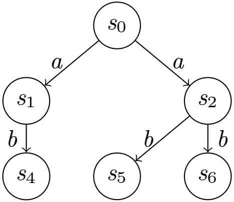


$\overset{?}{\models} \langle a \rangle [b] P$

Game

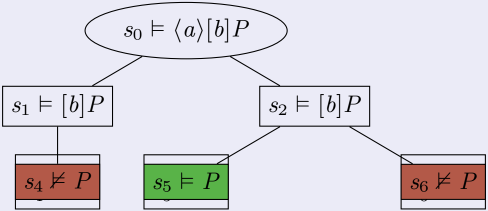


Example



$\overset{?}{\models} \langle a \rangle [b] P$

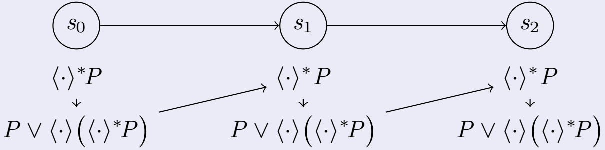
Game



Game rules: reachability

$$\langle\cdot\rangle^* P \equiv P \vee \langle\cdot\rangle\big(\langle\cdot\rangle^* P\big)$$ there is a path ending in P

Reachability: $\langle\cdot\rangle^* P$



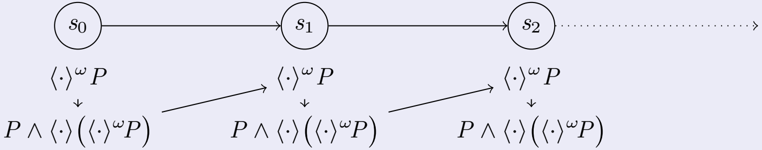
Who wins?

Eve wins if the game ends.

Game rules: safety

$\langle\cdot\rangle^\omega P \equiv P \wedge \langle\cdot\rangle\big(\langle\cdot\rangle^\omega P\big)$ there is an ω-path where P is always true

Safety: $\langle\cdot\rangle^\omega P$



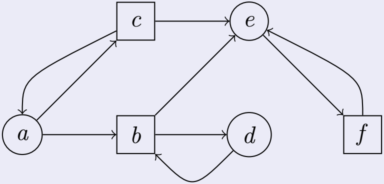
Who wins?

Eve wins if the game continues forever.

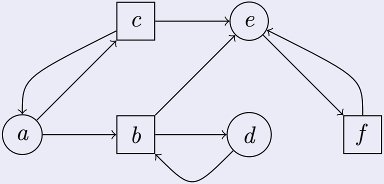
$\mathcal{M} \vDash \alpha$	modal logic	reachability	safety
↓	↓	↓	↓
$G(\mathcal{M}, \alpha)$	finite duration games	termination	non-termination

Parity games

Definition (Game $\mathcal{G} = \langle V_E, V_A, R, \lambda : V \to C, Acc \subseteq C^\omega \rangle$)



DEFINITION (GAME $\mathcal{G} = \langle V_E, V_A, R, \lambda : V \to C, Acc \subseteq C^\omega \rangle$)



DEFINITION (WINNING A PLAY)

Eve wins a play $v_0 v_1 \ldots$ iff the sequence is in Acc.

DEFINITION (WINNING POSITION)

A strategy for Eve is $\sigma_E : V^* \times V_E \to V$. A strategy is winning from a given position iff all the plays starting in this position and respecting the strategy are winning. A position is winning if there is a winning strategy from it.

What kind of winning conditions

Properties

- reachability
- safety
- etc.

Winning conditions

- reachability: $Acc =$(sequences passing through a position from F),
- safety: $Acc =$(sequences of elements of F),
- repeated reachability: $Acc =$(sequences with infinitely many elements from F).
- ultimately safe: $Acc =$(almost all elements from F).

The parity condition

Definition (Parity condition: $\Omega : V \to \{0, \dots, d\}$)

$$(v_0, v_1, \dots) \in Acc \quad \text{iff} \quad \liminf_{n\to\infty} \Omega(v_n) \text{ is even}$$

Examples

0, 1, 0, 1, 0, 1, 0, 1, 0, 1... liminf is even
0, 1, 0, 1, 2, 1, 2, 1, 2, 1... liminf is odd

Other conditions in terms of parity condition

- Infinitely often states from $F \subseteq V$.
  $\Omega : V \to \{0, 1\}$ such that $\Omega(v) = 0$ iff $v \in F$.
- Almost always states from $F \subseteq V$.
  $\Omega : V \to \{1, 2\}$ such that $\Omega(v) = 2$ iff $v \in F$.
- Reachability for F.
  Arrange so that each state from F is winning.
- Safety for F.
  $\Omega(v) = 0$ for $v \in F$ and arrange so that all states not in F are loosing.

Parity games $\equiv$ μ-calculus model checking

SYNTAX

$P \mid \neg P \mid X \mid \alpha \mid \alpha \vee \beta \mid \alpha \wedge \beta \mid \langle a \rangle \alpha \mid [a]\alpha \mid \mu X.\alpha \mid \nu X.\alpha$

SEMANTICS

Given $\mathcal{M} = \langle V, \{E_a\}_{a \in Act}, P^{\mathcal{M}}, \dots \rangle$ and $Val : Var \to \mathcal{P}(V)$ we define $[\![\alpha]\!]^{\mathcal{M}}_{Val} \subseteq \mathcal{P}(V)$.

$$[\![P]\!]^{\mathcal{M}}_{Val} = P^{\mathcal{M}}$$

$$[\![X]\!]^{\mathcal{M}}_{Val} = Val(X)$$

$$[\![\langle a \rangle \alpha]\!]^{\mathcal{M}}_{Val} = \{v : \exists v'.\ E_a(v, v') \wedge v' \in [\![\alpha]\!]^{\mathcal{M}}_{Val}\}$$

$$[\![\mu X.\alpha(X)]\!]^{\mathcal{M}}_{Val} = \bigcap \{S \subseteq V : [\![\alpha(S)]\!]^{\mathcal{M}}_{Val} \subseteq S\}$$

Notation: $\mathcal{M}, s \vDash \alpha$ for $s \in [\![\alpha]\!]^{\mathcal{M}}_{Val}$, where Val will be clear from the context.

Operator

$\alpha(X) : P(V) \to P(V)$

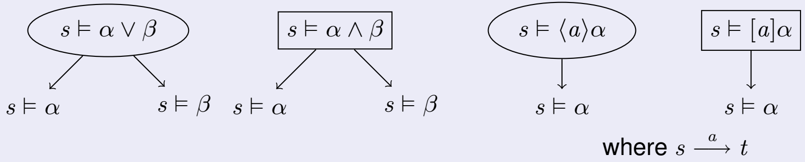
We will give a characterization of the semantics in terms of games

Games for the mu-calculus

Setup

- We are given a transition system $\mathcal{M}$ and a formula α_0.
- We define a game $G(\mathcal{M}, s_0, \alpha_0)$ where Eve wins from $(s_0 \vDash \alpha_0)$ iff $\mathcal{M}, s_0 \vDash \alpha_0$.

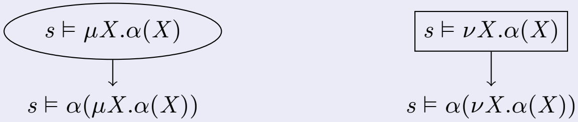
Game rules



In $(s \vDash P)$ Eve wins iff $s \in P^{\mathcal{M}}$ In $(s \vDash \neg P)$ Eve wins iff $s \notin P^{\mathcal{M}}$

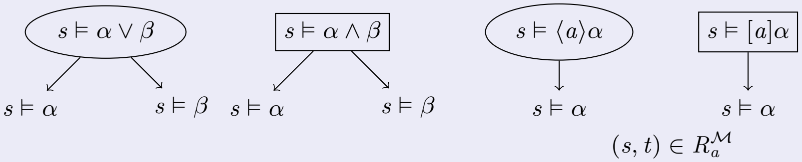
What to do with $\mu X.\alpha(X)$ and $\nu X.\alpha(X)$?

What to do with $\mu X.\alpha(X)$ and $\nu X.\alpha(X)$?



These two rules may be the source of infinite plays.

Game rules

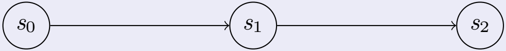


In $s \models P$ Eve wins iff $s \in P^{\mathcal{M}}$

In $s \models \neg P$ Eve wins iff $s \in P^{\mathcal{M}}$

Example: Reachability

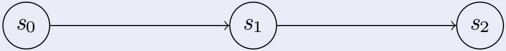
Reachability: $\langle\cdot\rangle^* P \equiv \mu X.P \vee \langle\cdot\rangle X$



$\alpha \equiv \mu X.P \vee \langle\cdot\rangle X$

Example: Reachability

Reachability: $\langle\cdot\rangle^* P \equiv \mu X.P \vee \langle\cdot\rangle X$



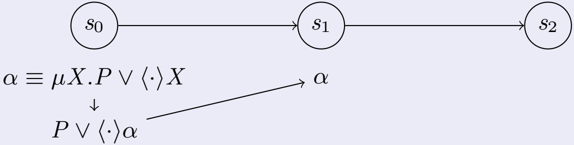
$$\alpha \equiv \mu X.P \vee \langle\cdot\rangle X$$

$$\downarrow$$

$$P \vee \langle\cdot\rangle\alpha$$

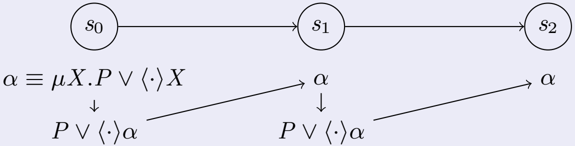
Example: Reachability

Reachability: $\langle\cdot\rangle^* P \equiv \mu X.P \vee \langle\cdot\rangle X$



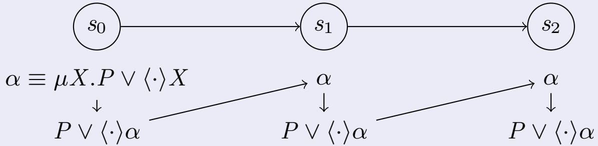
Example: Reachability

Reachability: $\langle\cdot\rangle^* P \equiv \mu X.P \vee \langle\cdot\rangle X$



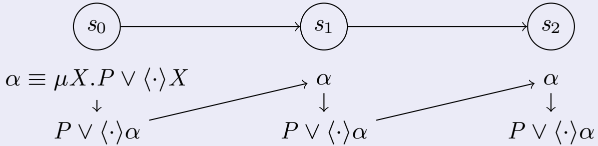
Example: Reachability

Reachability: $\langle\cdot\rangle^* P \equiv \mu X.P \vee \langle\cdot\rangle X$



Example: Reachability

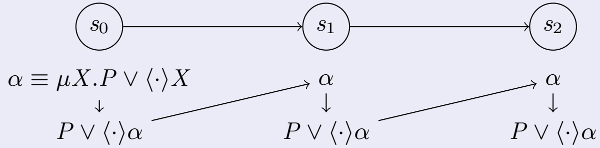
Reachability: $\langle\cdot\rangle^* P \equiv \mu X.P \vee \langle\cdot\rangle X$



Eve wins if the game ends in P. $\mu X.\alpha(X) = \bigcup_{\tau\in Ord} \mu^\tau X.\alpha(X)$

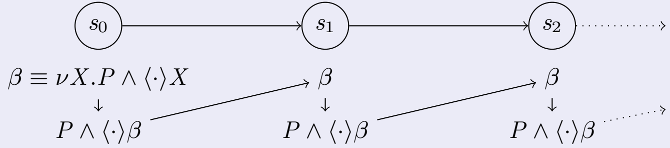
Example: Reachability

Reachability: $\langle\cdot\rangle^* P \equiv \mu X.P \vee \langle\cdot\rangle X$

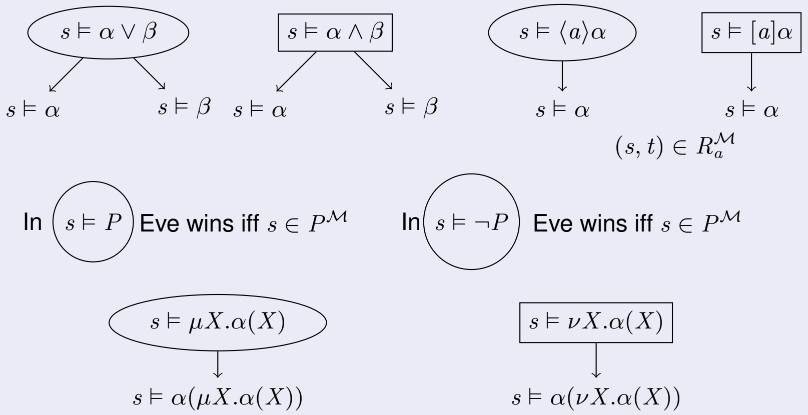


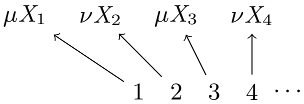
Eve wins if the game ends in P. $\mu X.\alpha(X) = \bigcup_{\tau\in Ord} \mu^\tau X.\alpha(X)$

Safety: $\langle\cdot\rangle^\omega P \equiv \nu X.P \wedge \langle\cdot\rangle X$



Eve wins if the game continues for ever.


$s \vDash \alpha \vee \beta$
$s \vDash \alpha$
$s \vDash \beta$
$s \vDash \alpha \wedge \beta$
$s \vDash \alpha$
$s \vDash \beta$
$s \vDash \langle a \rangle \alpha$
$s \vDash \alpha$
$s \vDash [a]\alpha$
$s \vDash \alpha$
$(s, t) \in R_a^{\mathcal{M}}$
In $s \vDash P$ Eve wins iff $s \in P^{\mathcal{M}}$
In $s \vDash \neg P$ Eve wins iff $s \in P^{\mathcal{M}}$
$s \vDash \mu X.\alpha(X)$
$s \vDash \alpha(\mu X.\alpha(X))$
$s \vDash \nu X.\alpha(X)$
$s \vDash \alpha(\nu X.\alpha(X))$



$\alpha(X_1, X_2, \dots)$

- μ's have odd ranks,
- ν's have even ranks,
- if β is a subformula of α then β has bigger rank than α.

THE WINNING CONDITION IS THE PARITY CONDITION

Eve wins if the smallest priority appearing infinitely often is even.

EXAMPLE

$\mu_1 Y.\nu_2 X.(P \wedge \langle\cdot\rangle X) \vee \langle\cdot\rangle Y \qquad \nu_2 X.\mu_3 Y(P \wedge \langle\cdot\rangle X) \vee \langle\cdot\rangle Y$

MC ⇒ GAME SOLVING

The problem $\mathcal{M}, s_0 \overset{?}{\vDash} \alpha_0$ is reduced to deciding if Eve wins in the game $\mathcal{G}(\mathcal{M}, s_0, \alpha_0)$.

GAME SOLVING ⇒ MC

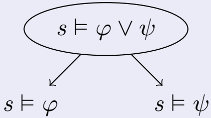
- Game can be represented as a transition system.
- There is a μ-calculus formula that is true exactly in the positions where Eve wins.

REMARKS

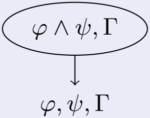
- Other logics can be handled in the same way.
- This also explains algorithmics of verification nicely, which is especially useful for verification of infinite structures.
- Satisfiability can be also reduced to parity games.

Games and satisfiability.

We want to design a game for satisfiability checking



Eve chooses

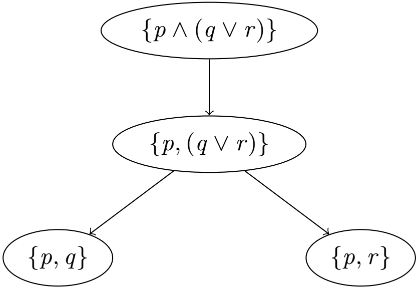


Adam chooses

If Γ is irreducible then Adam wins iff $P, \neg P \in \Gamma$.

Properties

- Eve has a winning strategy from φ iff φ is satisfiable.
- Every model of φ can be obtained from a winning strategy in the game.



The two leaves represent two valuations that satisfy the root formula.

Remarks

- This kind of game can be extended to the mu-calculus
- Interestingly, we still obtain parity games at the end.
- Moreover every winning strategy corresponds to a model, and “all” models can be obtained in such a way.

Properties of games.

Remark

From Martin's theorem it follows that parity games are determined, i.e., form every position one of the players has a winning strategy.

Theorem (Mostowski, Emerson & Jutla)

In a parity game a player has a memoryless winning strategy from each of his winning positions.

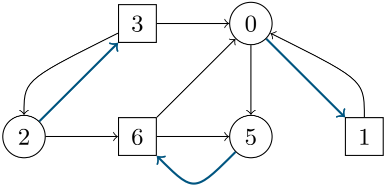
Memoryless strategy

- In general a strategy for Eve is $\rho : V^* V_E \to V$.
- Memoryless strategy is $\sigma : V^E \to V$ (depends only on the current position).
- **Rem:** One can also often see the term positional determinacy.
- **Rem:** If games are presented as trees, memoryless means that it behaves identically in isomorphic subtrees.

Memoryless strategies: an example

Memoryless strategy

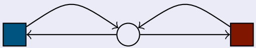
- Memoryless strategy is $\sigma : V^E \to V$ (depends only on the current position).



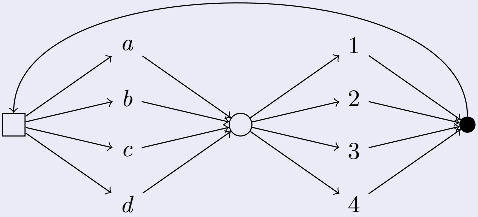
Muller conditions

Coloring vertices with a finite number of colors. The winner is decided by looking at the colors that appear infinitely often.

Example of a Muller cond.: see both colors infinitely often



A more complicated example



Some winning sets:

$\{a, 1\}$
$\{b, 1\}$
$\{c, d, 2\}$
$\{c, d, 1, 2\}$

The biggest number seen infinitely often = the number of letters seen infinitely often

Memoryless determinacy

A winning condition admits memoryless determinacy iff all the games with this condition are memoryless determined. (from every position one of the players has a memoryless winning strategy).

Theorem (McNaughton, Gurevich & Harrington)

Parity conditions are the only Muller conditions admitting memoryless determinacy. In general Muller conditions need finite memory.

Colors in ω.

- We can still talk about minimal color appearing infinitely often, even though it may not always exist.
- **Theorem [Graedel & W.]** Games with infinite parity conditions admit memoryless determinacy. All other conditions need infinite memory.

Definition

To solve a game is to determine for each position who has a winning strategy.

Fact

There is an algorithm for solving finite parity games.

Open problem

Is there a polynomial time algorithm?

Decidability of MSOL on trees

Monadic second-order logic

- Quantification over sets instead of quantification over elements.
$$\exists X.\varphi(X), \qquad \forall X.\varphi(X)$$
- The inclusion predicate: $X \subseteq Y$.
- Standard predicates "lifted" to sets: $succ(X, Y)$, $X \subseteq P$

Model: infinite binary tree

Theorem (Rabin)

Monadic second-order theory of the binary tree is decidable

Remark

This is a very strong decidability result. Many other problems (Presburger arithmetic, theory of order, ...) reduce to it.

Remark

Memoryless determinacy of parity games is the combinatorial content of the proof of Rabin's theorem.

Mean pay-off game: $G = \langle V_E, V_A, R, w : (V_E \cup V_A) \to \mathbb{N} \rangle$

Outcome for Eve of a play $v_0, v_1, \ldots$ is:

$$\liminf_{n \to \inf} \frac{1}{n} \sum_{i=1}^{n} w(v_i).$$

For Adam it is $\lim\sup$.

Discounted payoff game $G = \langle V_E, V_A, R, w : (V_E \cup V_A) \to \mathbb{R} \rangle$

Outcome of $v_0, v_1, \ldots$ is

$$(1-\delta) \sum_{i=0}^{\infty} \delta^i w(v_i)$$

here $0 < \delta < 1$ is a discount factor.

Relation to parity games

Solving parity games can be reduced to solving games with one of these conditions.

Extensions

- Games on infinite graphs.
- Games with partial information.

Definition (Pushdown graph $G(P)$)

- Vertices: $Q \times \Gamma^*$
- Edges: $qw \to q'w'$ according to the rules applied to prefixes.

$$
\begin{array}{ccccccccccccc}
q_0\bot & \longrightarrow & q_0\,a\bot & \longrightarrow & q_0\,aa\bot & \to & q_0\,aaa\bot & \longrightarrow & \cdots & \longrightarrow & q_0\,a^k\bot & \longrightarrow & \cdots \\
 & & \downarrow\uparrow & & \downarrow & & \downarrow & & & & \downarrow & & \\
 & & q_1\bot & \longleftarrow & q_1\,a\bot & \longleftarrow & q_1\,aa\bot & \longleftarrow & \cdots & \longleftarrow & q_1\,a^{k-1}\bot & \longleftarrow & \cdots
\end{array}
$$

This is (a part of) the graph of the system:

$$
\begin{array}{ll}
q_0\bot \rightarrowtail q_0\,a\bot & \qquad q_1\bot \rightarrowtail q_0\,a\bot \\
q_0\,a \rightarrowtail q_0\,aa & \qquad q_1\,a \rightarrowtail q_1 \\
q_0\,a \rightarrowtail q_1 &
\end{array}
$$

Some questions

Solving games

Algorithmic feasibility of solving infinite games given in a finite way.

Some other kinds of winning conditions

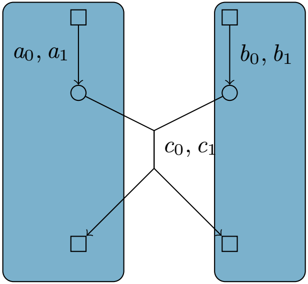
- In pushdown games we can ask that the size of the stack stays bounded.

Quality of strategies

- Do there exist memoryless strategies? Finite memory strategies?
- If so, are they “implementable” by a finite automaton, pushdown automaton?

SITUATION

A team of players put against one opponent. Each of the players in the team sees only part of the play (but has total knowledge of the arena).



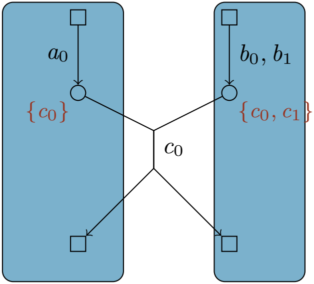
WINNING CONDITIONS

STRATEGY

In each round vertex the player declares which action he is ready to do.

Situation

A team of players put against one opponent. Each of the players in the team sees only part of the play (but has total knowledge of the arena).



Winning conditions

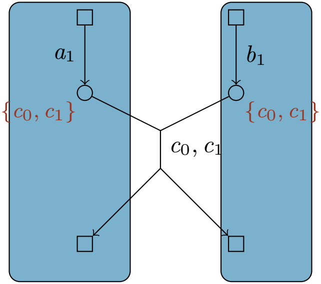
1. $a_i b_j c_k$ with $k = i$.

Strategy

In each round vertex the player declares which action he is ready to do.

SITUATION

A team of players put against one opponent. Each of the players in the team sees only part of the play (but has total knowledge of the arena).



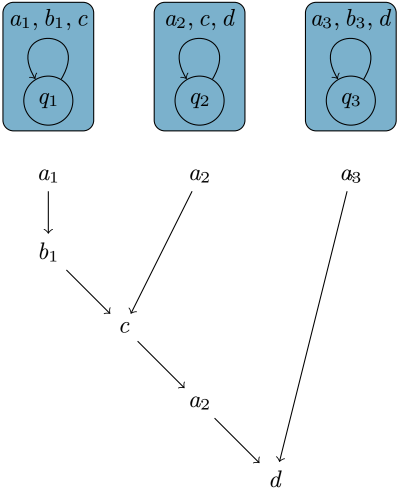
WINNING CONDITIONS

2. $a_i b_j c_k$ with $k = min\{i, j\}$.

STRATEGY

In each round vertex the player declares which action he is ready to do.

Partial information



What makes this situation special

- The game is repeating of infinite duration.
- The rounds that are played depend on the states of others.
- There is an implicit flow of information.

Conclusions

- Parity conditions have been “invented” in a study of tree automata [Mostowski].
- Relation with fixpoints or monadic second-order logic took some time to be discovered. [Niwiński, Emerson & Jutla]
- The memoryless determinacy [Gurevich & Harrington] is an important concept, and a very useful result.
- Open questions (directions):
  - Is it possible to solve parity games in PTIME?
  - Can partial information games be solved algorithmically?

[5, 8, 2, 4, 7, 1, 6, 3]

E. Grädel and I. Walukiewicz.
Positional determinacy of games with infinitely many priorities.
Logical Methods in Computer Science, 2(4), 2006.

M. Jurdziński.
Deciding the winner in parity games is in UP$\cap$co-UP.
Information Processing Letters, 68(3):119–124, 1998.

A. Muscholl, I. Walukiewicz, and M. Zeitoun.
A look at the control of asynchronous automata.
In *Perspectives in Concurrency Theory – Festschrift for P.S. Thiagarajan.*, pages 356–371. Universities Press, 2008.

O. Serre.
Contribution à l'étude des jeux sur des graphes de processus à pile.
PhD thesis, Université Paris VII, 2004.

W. Thomas.
Languages, automata, and logic.
In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume 3. Springer-Verlag, 1997.

M. Y. Vardi and T. Wilke.
Automata: From logics to algorithms.
In J. Flum, E. Grädel, and T. Wilke, editors, *Logic and Automata*, volume 2 of *Texts in Logic and Games*. Amsterdam University Press, 2007.

I. Walukiewicz.
A landscape with games in the background.
In *IEEE LICS*, pages 356–366, 2004.
Invited lecture.

W. Zielonka.
Infinite games on finitely coloured graphs with applications to automata on infinite trees.
Theoretical Computer Science, 200:135–183, 1998.