Automata theory for strategies in games

Sujata Ghosh and R. Ramanujam

University of Groningen, The Netherlands

The Institute of Mathematical Sciences, Chennai, India

August 2, 2011

Infinite games

We spoke of temporally large games, as games of unbounded duration.

Infinite games

We spoke of temporally large games, as games of unbounded duration.

- Why not consider games of infinite duration ?

Infinite games

We spoke of temporally large games, as games of unbounded duration.

- Why not consider games of infinite duration ?
- How do we define outcomes in such games ?

Infinite games

We spoke of temporally large games, as games of unbounded duration.

- Why not consider games of infinite duration ?
- How do we define outcomes in such games ?
- This is not only possible, but this is how game theory began.

Games set theorists play

The classic two-person zero-sum infinite game of perfect information.

- The game is given by $A \subseteq \mathbb{N}^\omega$.
- Player I picks a natural number x_1, II responds with $x_2 > x_1$, I now picks $x_3 > x_2$ and thus goes the game.
- If the resulting play is in A, player I wins, else she loses.

Games set theorists play

The classic two-person zero-sum infinite game of perfect information.

- The game is given by $A \subseteq \mathbb{N}^\omega$.
- Player I picks a natural number x_1, II responds with $x_2 > x_1$, I now picks $x_3 > x_2$ and thus goes the game.
- If the resulting play is in A, player I wins, else she loses.
- Does it make sense (unless you are a set theorist) to wait until ω to wait to see if you win ? Would you play such a game ?

Strategies for ever

Strategizing in such games depends on the structure of the outcome set A.

- Case: A is finite.

Strategies for ever

Strategizing in such games depends on the structure of the outcome set A.

- Case: A is finite.
- Case: A is of the form X^ω where $X \subseteq \mathbb{N}^*$.

Strategies for ever

Strategizing in such games depends on the structure of the outcome set A.

- Case: A is finite.
- Case: A is of the form X^ω where $X \subseteq \mathbb{N}^*$.
- Is every game determined ? (That is, one of the two players has a winning strategy ?)

Undetermined game

Theorem: There exists a game that is not determined.

Undetermined game

Theorem: There exists a game that is not determined.

- We need to construct a set $A \subseteq \mathbb{N}^\omega$ such that neither of the players has a winning strategy.
- Fix a set A with properties to be defined soon.

Undetermined game

Theorem: There exists a game that is not determined.

- We need to construct a set $A \subseteq \mathbb{N}^\omega$ such that neither of the players has a winning strategy.
- Fix a set A with properties to be defined soon.
- Assume player I has a winning strategy.
- We will show that player II can use copycat and have a winning strategy as well.
- But that's a contradiction.

The set A

A famous set $A \subseteq \mathbb{N}^\omega$ exists, with the following properties.

- It contains every co-finite subset of $\mathbb{N}$. '
- It is closed under intersections and supersets.
- $\emptyset \notin A$.
- For every $X \subseteq \mathbb{N}$, either $X \in A$ or $(\mathbb{N} - X) \in A$.

A rejoices under the name: the principal ultrafilter extending the Frechèt filter.

The set A

A famous set $A \subseteq \mathbb{N}^\omega$ exists, with the following properties.

- It contains every co-finite subset of $\mathbb{N}$. '
- It is closed under intersections and supersets.
- $\emptyset \notin A$.
- For every $X \subseteq \mathbb{N}$, either $X \in A$ or $(\mathbb{N} - X) \in A$.

A rejoices under the name: the principal ultrafilter extending the Frechèt filter.
Crucially, no finite set is in A.

Plays and outcomes

Let $\rho = x_1 x_2 x_3 \ldots$ be a play.

- Note that player I's choices are $x_1 x_3 x_5 \ldots$.
- Define $G_\rho = (0, x_1] \cup (x_2, x_3] \cup (x_4, x_5] \cup \ldots$.
- I wins iff $G_\rho \in A$.

Copycat

Player II plays a parallel game with Player I, the latter plays the winning strategy in both.

- In both games, I plays x_1 first. In the first game, II responds with arbitrary x_2, I responds with x_3.

Copycat

Player II plays a parallel game with Player I, the latter plays the winning strategy in both.

- In both games, I plays x_1 first. In the first game, II responds with arbitrary x_2, *I* responds with x_3.
- *II* copies x_3 as his response to x_1 in the second game, gets x_4 as I's response, copies it in the first game, and so on.

Copycat

Player II plays a parallel game with Player I, the latter plays the winning strategy in both.

- In both games, I plays x_1 first. In the first game, II responds with arbitrary x_2, *I* responds with x_3.
- *II* copies x_3 as his response to x_1 in the second game, gets x_4 as I's response, copies it in the first game, and so on.
- I wins both games, so we get two winning sets in A:
  $G_1 = (0, x_1] \cup (x_2, x_3] \cup (x_4, x_5] \cup \ldots$.
  $G_2 = (0, x_2] \cup (x_3, x_4] \cup (x_5, x_6] \cup \ldots$.

Copycat

Player II plays a parallel game with Player I, the latter plays the winning strategy in both.

- In both games, I plays x_1 first. In the first game, II responds with arbitrary x_2, I responds with x_3.
- II copies x_3 as his response to x_1 in the second game, gets x_4 as I's response, copies it in the first game, and so on.
- I wins both games, so we get two winning sets in A:
  $G_1 = (0, x_1] \cup (x_2, x_3] \cup (x_4, x_5] \cup \ldots.$
  $G_2 = (0, x_2] \cup (x_3, x_4] \cup (x_5, x_6] \cup \ldots.$
- But then $G_1 \cap G_2 = (0, x_1]$, a finite set is in A, a contradiction.

Regular infinite games

We will be presented in finitely presented infinite games.

Regular infinite games

We will be presented in finitely presented infinite games.

- How do we present such games ?

Regular infinite games

We will be presented in finitely presented infinite games.

- How do we present such games ?
- The game arena is a finite graph.

Regular infinite games

We will be presented in finitely presented infinite games.

- How do we present such games ?
- The game arena is a finite graph.
- It can have cycles since finite state players can forget history of play.

Regular infinite games

We will be presented in finitely presented infinite games.

- How do we present such games ?
- The game arena is a finite graph.
- It can have cycles since finite state players can forget history of play.
- Outcomes can be given on the graph themselves.

Why bother ?

Are such infinite games only of mathematical interest ?

- Reactive systems: browsers, operating systems, routers, . . .

Why bother ?

Are such infinite games only of mathematical interest ?

- Reactive systems: browsers, operating systems, routers, ...
- Objectives are long range, eventually stable behaviour.

Why bother ?

Are such infinite games only of mathematical interest ?

- Reactive systems: browsers, operating systems, routers, ...
- Objectives are long range, eventually stable behaviour.
- Once again game theoretic considerations are relevant.

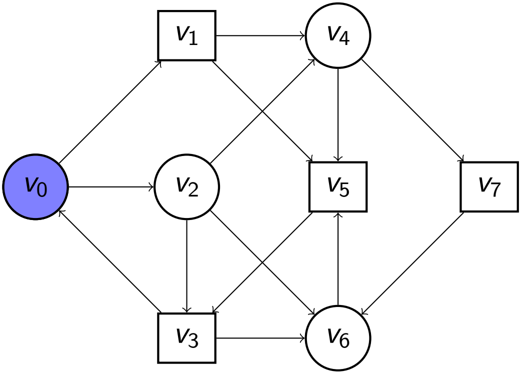
Infinite plays

Arenas and winning conditions

- Game $\mathcal{G} = (\mathcal{G}, Win)$, where
  - Arena $\mathcal{G} = (V, E)$ such that $V = V_0 \cup V_1$ where
    - V_0 are player 0 vertices, denoted by $\bigcirc$ and
    - V_1 are player 1 vertices, denoted by $\square$
    - $E \subseteq V \times V$ is the edge relation
  - $Win \subseteq V^\omega$ is the winning condition
  - $vE = \{u \mid (v, u) \in E\}$ are the neighbours of v
- $(\mathcal{G}, v_0)$ is an initialised game where $v_0 \in V$ is a designated vertex
- We assume $vE \neq \emptyset$ for every $v \in V$

Infinite plays[2]

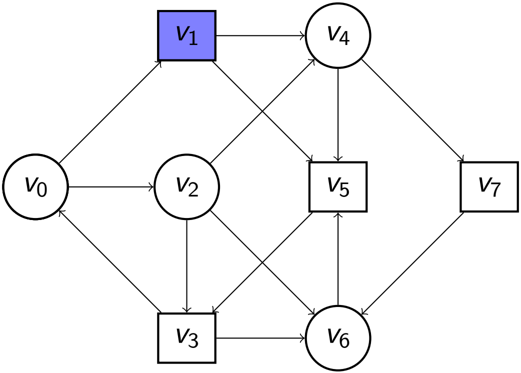
Plays



$\rho = v_0$

Infinite plays[2]

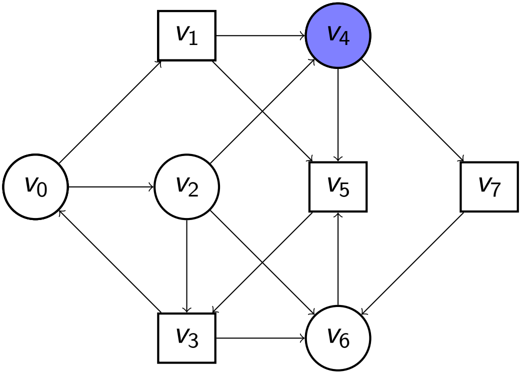
Plays



$\rho \; = \; v_0 \;\; v_1$

Infinite plays[2]

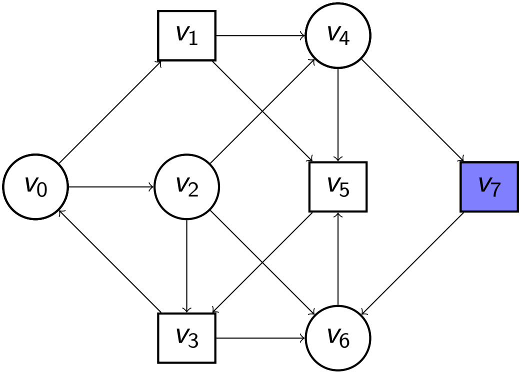
Plays



$$\rho = v_0 \; v_1 \; v_4$$

Infinite plays[2]

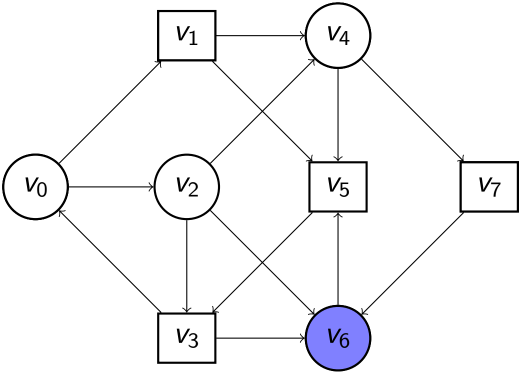
Plays



$$\rho = v_0 \ v_1 \ v_4 \ v_7$$

Infinite plays[2]

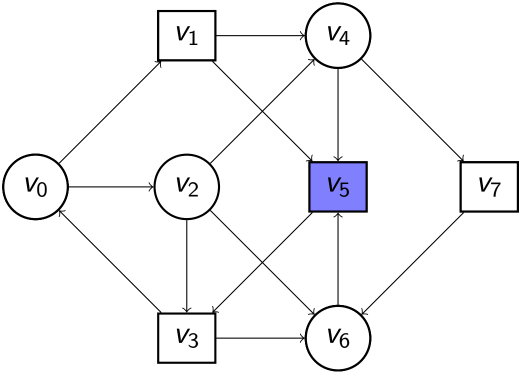
Plays



$$\rho = v_0 \; v_1 \; v_4 \; v_7 \; v_6$$

Infinite plays[2]

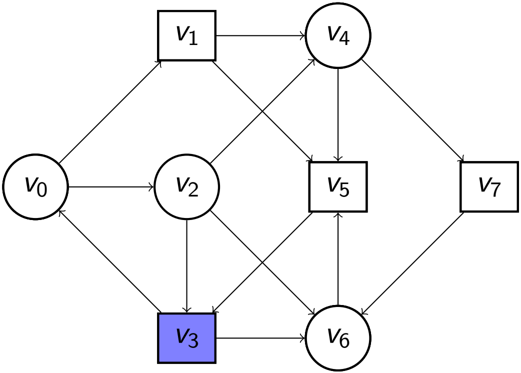
Plays



$\rho \ = \ v_0 \ v_1 \ v_4 \ v_7 \ v_6 \ v_5$

Infinite plays[2]

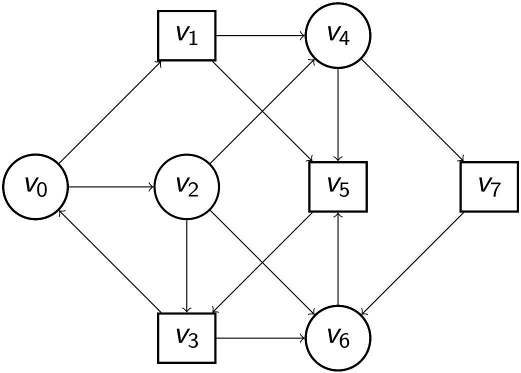
Plays



$$\rho = v_0 \ v_1 \ v_4 \ v_7 \ v_6 \ v_5 \ v_3$$

Infinite plays[2]

Plays



$$\rho \;=\; v_0 \;\; v_1 \;\; v_4 \;\; v_7 \;\; v_6 \;\; v_5 \;\; v_3 \cdots \;\; \in V^\omega$$

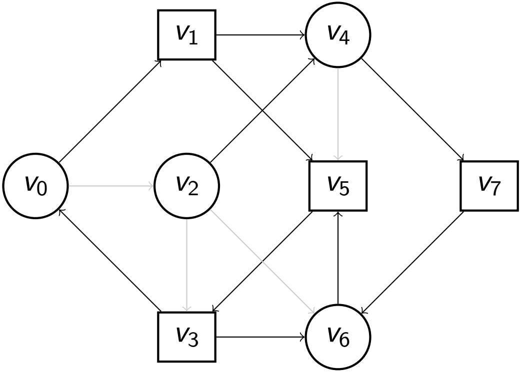
Strategies

Strategies

- A history ρ is any finite prefix of a play ρ
- A strategy *Strat* for player p, $p \in \{0, 1\}$ is a function $Strat : V^* V_p \to V$ from the set of histories to vertices such that $Strat(\rho v) \in vE$ for all ρ
- A strategy *Strat* is called finite memory or bounded memory or forgetful if it can be represented by a finite state machine
- A strategy *Strat* for player p is memoryless or positional if it does not depend on the history. That is $Strat : V_p \to V$

Memoryless strategies

Memoryless strategy as a subgraph



Determinacy

Zero-sum or win-lose games and determinacy

- A play ρ is winning for player 0 and losing for player 1 if $\rho \in \mathit{Win}$. Otherwise it is winning for player 1 and losing for player 0
- A strategy *Strat* of player 0 is winning if and only if all plays played according to *Strat* are winning for her
- The winning region of player p, W_p is a subset of V such that for every vertex $v \in W_p$, player p has a winning strategy for the game $(\mathcal{G}, v)$
- A game $\mathcal{G}$ is determined if $W_0 \cup W_1 = V$ and $W_0 \cap W_1 = \emptyset$

Theorem (Martin '75)

Every game where Win is a Borel set is determined.

Specifying outcomes and preferences

Even the set of finite plays can be infinite.

Specifying outcomes and preferences

Even the set of finite plays can be infinite.

- For each play, we need to specify outcomes.
- For each player, we need to give an ordering over plays.

Specifying outcomes and preferences

Even the set of finite plays can be infinite.

- For each play, we need to specify outcomes.
- For each player, we need to give an ordering over plays.
- For finite plays, simple reachability objectives suffice.

Specifying outcomes and preferences

Even the set of finite plays can be infinite.

- For each play, we need to specify outcomes.
- For each player, we need to give an ordering over plays.
- For finite plays, simple reachability objectives suffice.
- Simply given by a subset of positions.

A logical syntax

An elegant solution to this is to use propositions to label game positions.

A logical syntax

An elegant solution to this is to use propositions to label game positions.

- Then “winning” a reachability condition is simply given by a boolean formula.

A logical syntax

An elegant solution to this is to use propositions to label game positions.

- Then "winning" a reachability condition is simply given by a boolean formula.
- Ordering over outcomes can be given by implication.
- This also paves the way for strategy specifications.

Local strategies

How do resource limited players strategize ?

Local strategies

How do resource limited players strategize ?

- Test a (local) condition; if it holds, play a specific move.

Local strategies

How do resource limited players strategize ?

- Test a (local) condition; if it holds, play a specific move.
- Record some observables during course of play; depending on what the record shows, play some move.

Local strategies

How do resource limited players strategize ?

- Test a (local) condition; if it holds, play a specific move.
- Record some observables during course of play; depending on what the record shows, play some move.
- Watch another player's moves: when he plays *a*, respond with *b*.

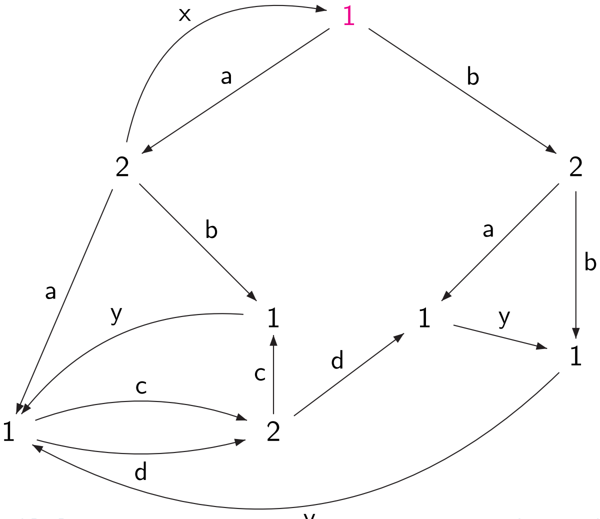
Local strategies

How do resource limited players strategize ?

- Test a (local) condition; if it holds, play a specific move.
- Record some observables during course of play; depending on what the record shows, play some move.
- Watch another player's moves: when he plays *a*, respond with *b*.
- Tit for tat; copycat; co-operate until defection, then punish, etc.

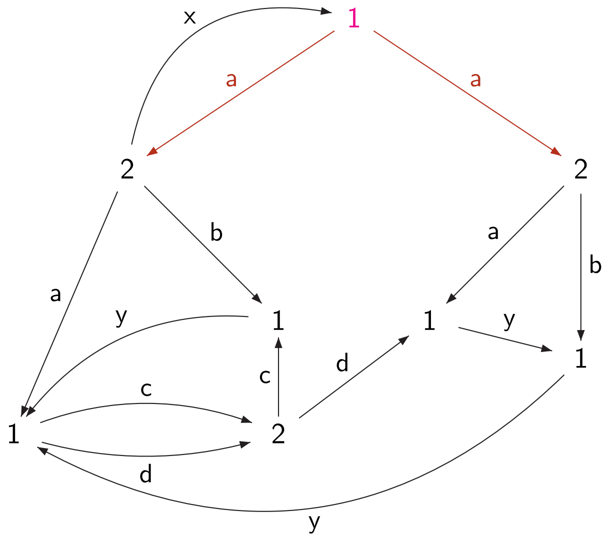
Game Arena

Nodes represents player positions. Edges represent player moves.



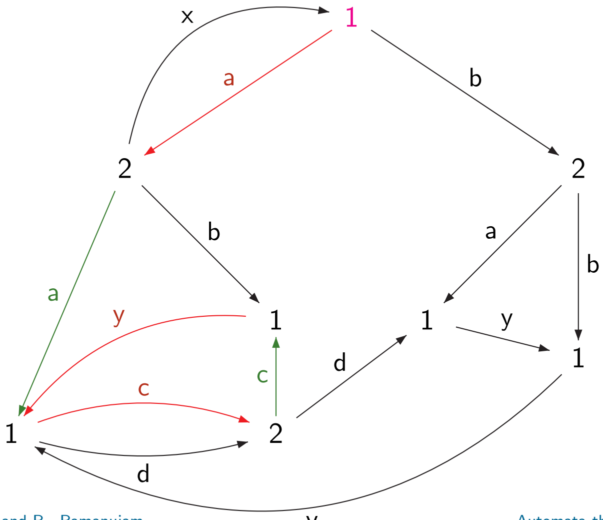
Game Arena

$\rightarrow: (Positions \times Actions) \rightarrow Positions.$



Play

A play is simply a path in the graph where at each node the players choose an action.



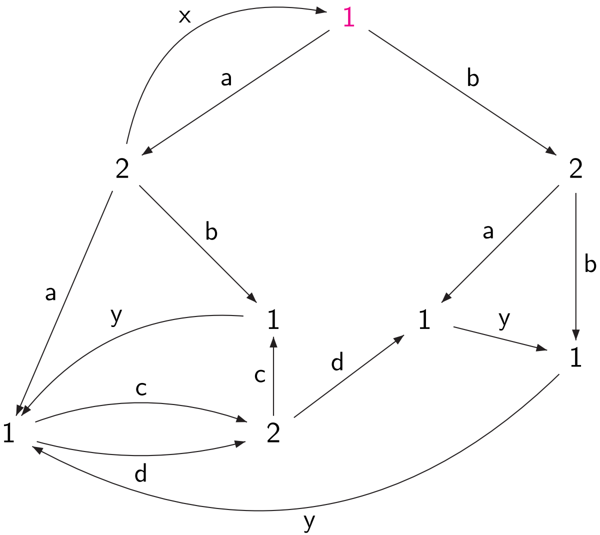
Strategy

Strategy for player 1 is a subgraph of the arena where:

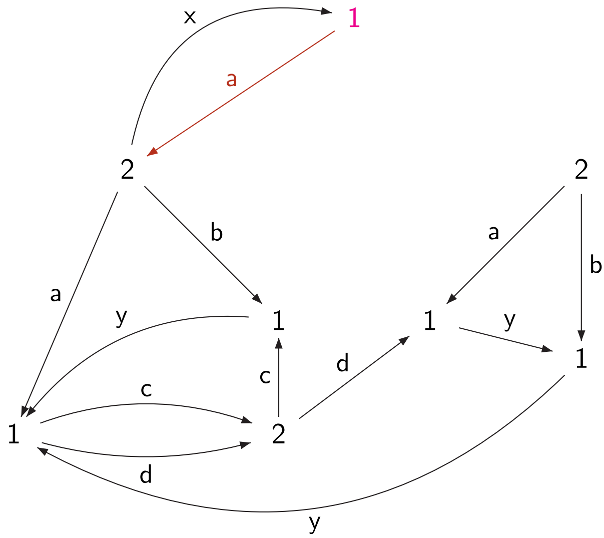
- For each player 1 node, there is a unique outgoing edge.
- For each player 2 node, every move is included.

Strategy

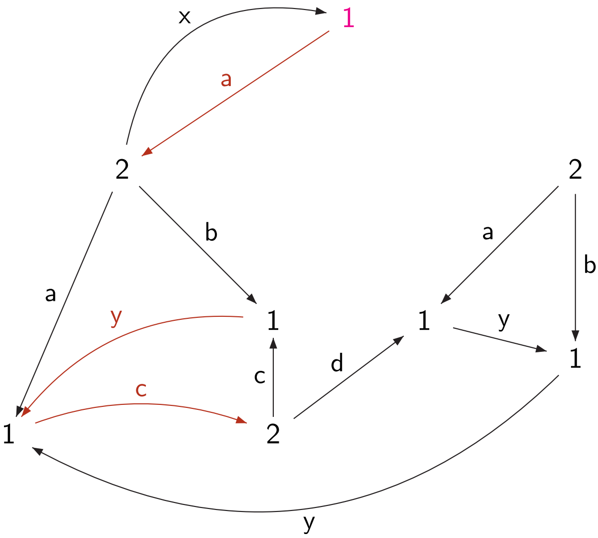
Player 1 strategy



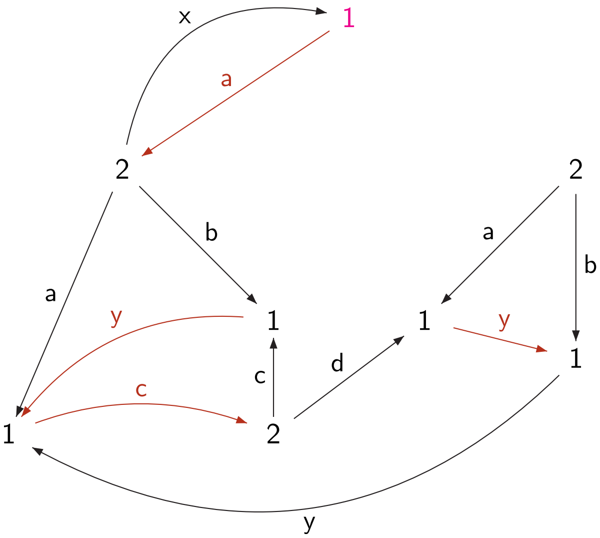
Strategy



Strategy

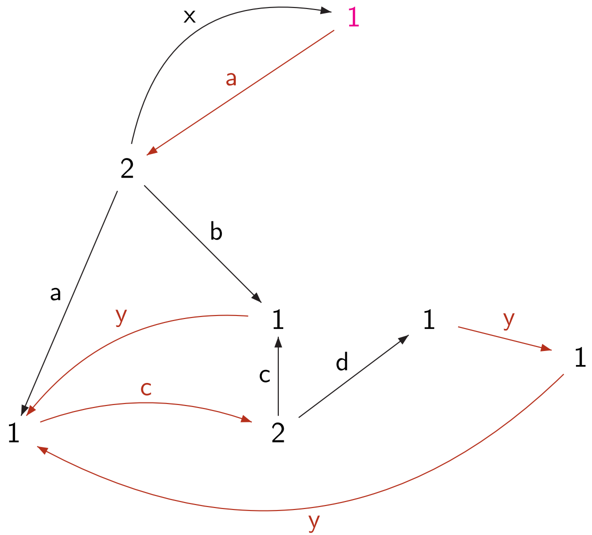


Strategy



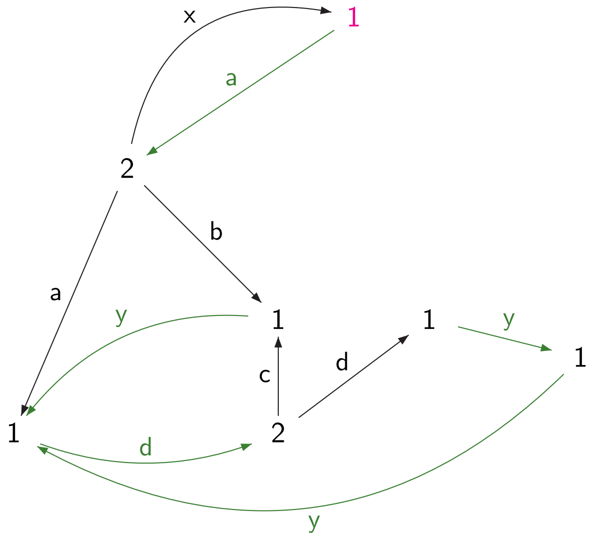
Strategy

Red strategy for player 1.



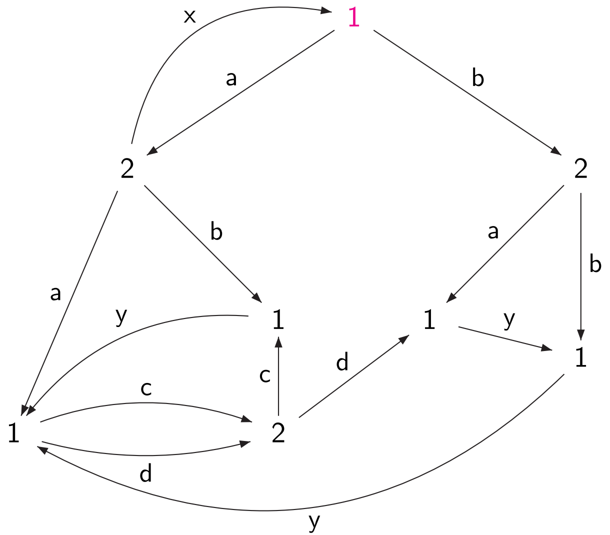
Strategy

Green strategy for player 1.



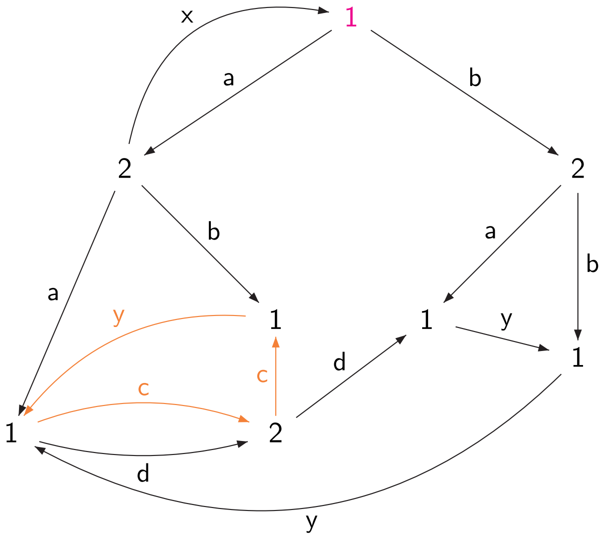
Objectives

Players have preferences over paths in the arena.



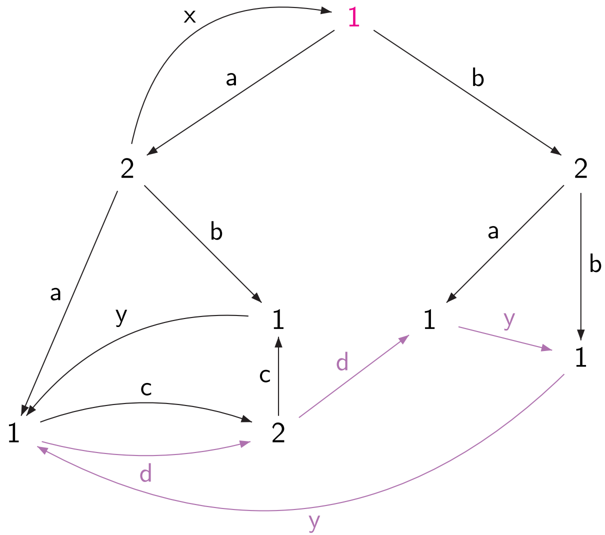
Objectives (player 2)

Path 1: Match a "c" move of player 1 with "c".



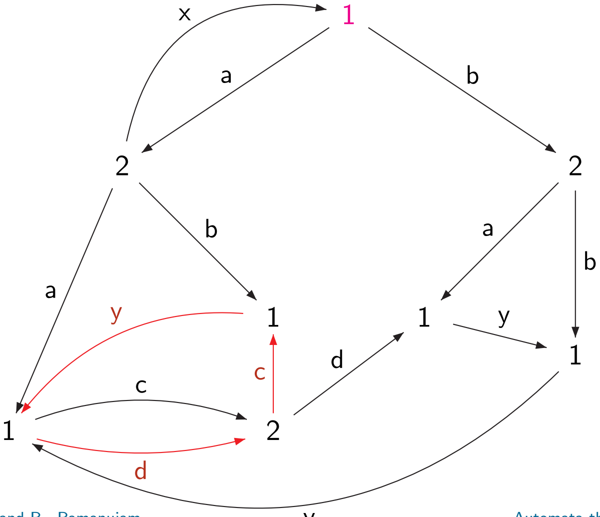
Objectives (player 2)

Path 2: Match a “d” move of player 1 with “d”.



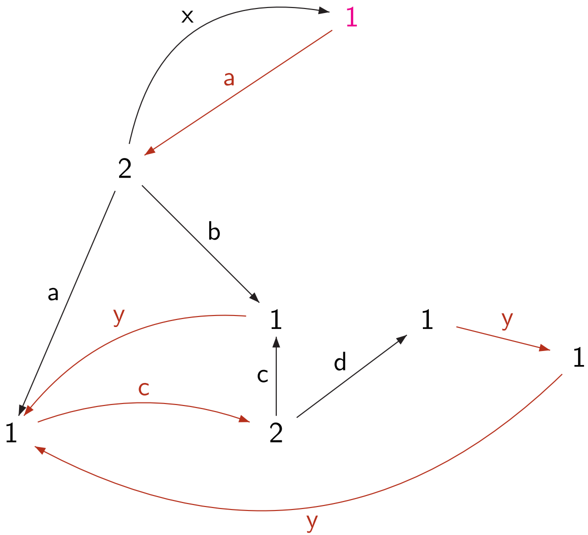
Objectives

Matching a "d" move of player 1 with "c" - least prefered by player 2.



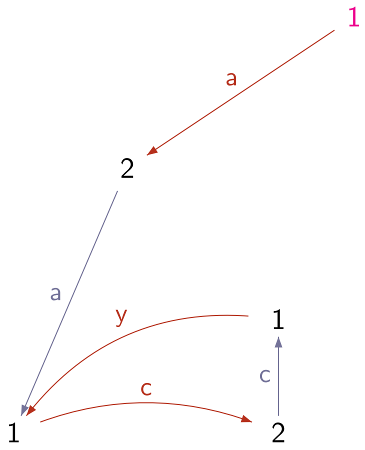
Response

Red strategy of player 1.



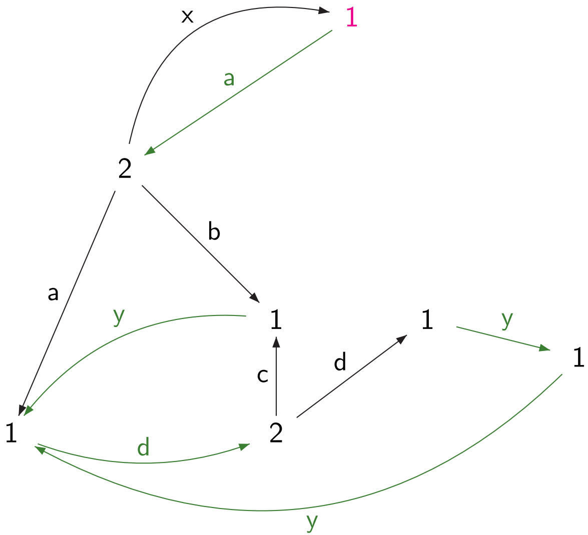
Response

Response of player 2 to the Red strategy achieves “path 1”.



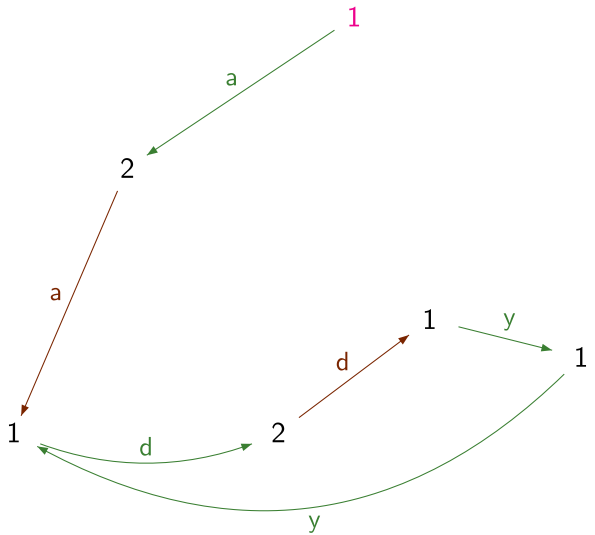
Response

Green strategy of player 1.

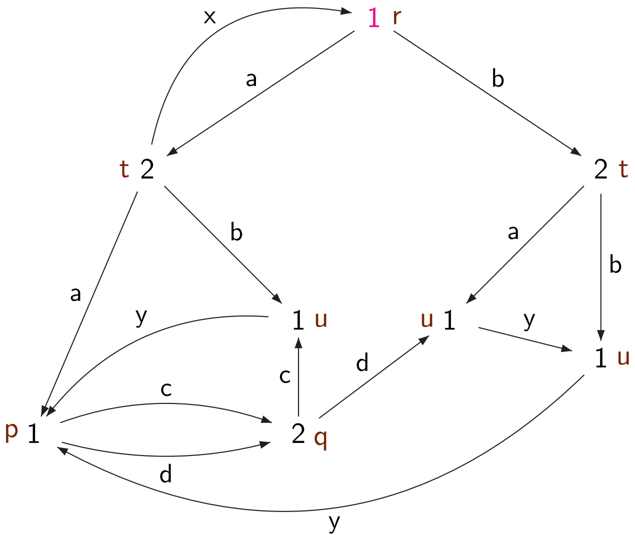


Response

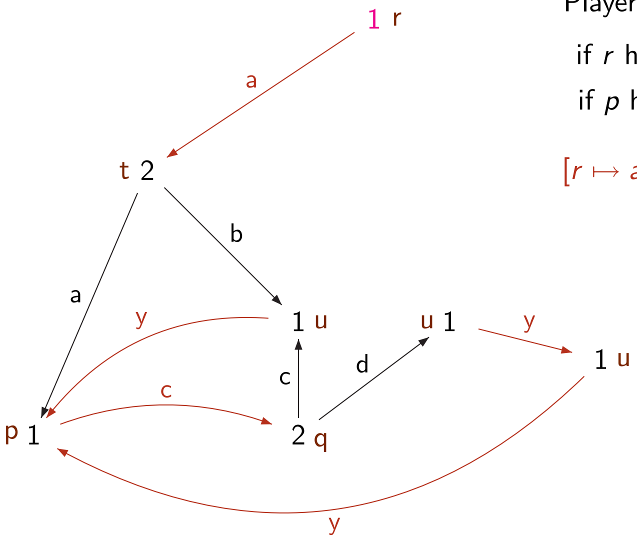
Response of player 2 to the Green strategy achieves “path 2”.



Game Arena with Valuation



Properties of strategies



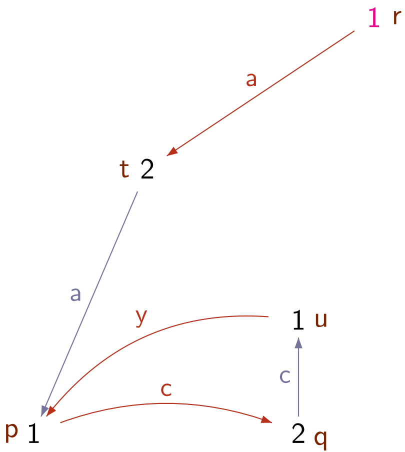
Player 1 strategy:

if r holds play a,

if p holds play c

$[r \longmapsto a]^1 \cdot [p \longmapsto c]^1$

Properties of strategies



Player 1 strategy:

if r holds play a,

if p holds play c

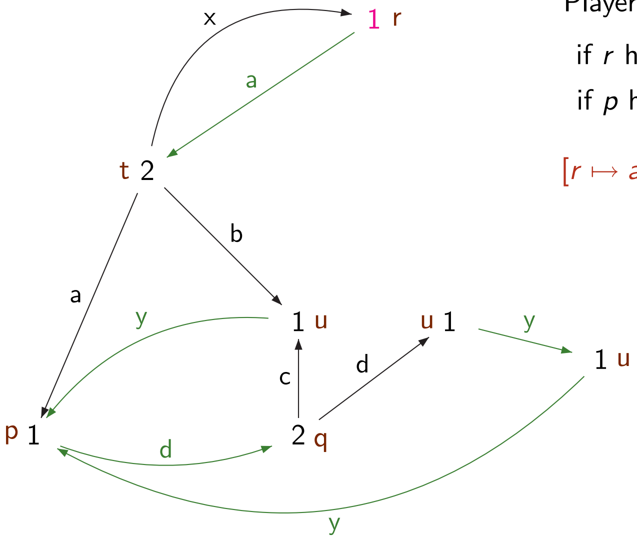
$[r \longmapsto a]^1 \cdot [p \longmapsto c]^1$

Response of player 2:

if q holds play c.

$[q \longmapsto c]^2$

Properties of strategies



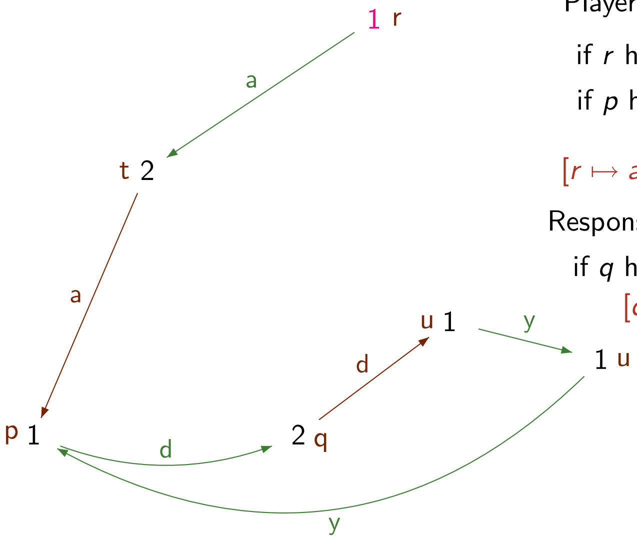
Player 1 strategy:

if r holds play a,

if p holds play d

$[r \longmapsto a]^1 \cdot [p \longmapsto d]^1$

Properties of strategies



Player 1 strategy:

if r holds play a,

if p holds play d

$[r \mapsto a]^1 \cdot [p \mapsto d]^1$

Response of player 2:

if q holds play d.

$[q \mapsto d]^2$

Properties of Strategies

- $([p \mapsto c]^1 \Rightarrow [q \mapsto c]^2) \cdot ([p \mapsto d]^1 \Rightarrow [q \mapsto d]^2)$
- If 1 plays $[p \mapsto c]$ then play $[q \mapsto c]$ and if 1 plays $[p \mapsto d]$ then play $[q \mapsto d]$.

Structured strategy specification

- Syntax: $[p \mapsto a]^1 \mid \ \sigma_1 + \sigma_2 \mid \ \sigma_1 \cdot \sigma_2 \mid \ \pi \Rightarrow \sigma$.
- Strategy conforming to a specification σ.

Game

We will consider finite and infinite plays.

- Players have an *exit* action.

Game

We will consider finite and infinite plays.

- Players have an *exit* action.
- Game arena consists of a unique "terminal" node.

Game

We will consider finite and infinite plays.

- Players have an *exit* action.
- Game arena consists of a unique "terminal" node.

A game $G = (\mathcal{G}, \{\preceq^i\}_{i \in \{1,2\}})$

- $\mathcal{G}$ is the game arena.

Game

We will consider finite and infinite plays.

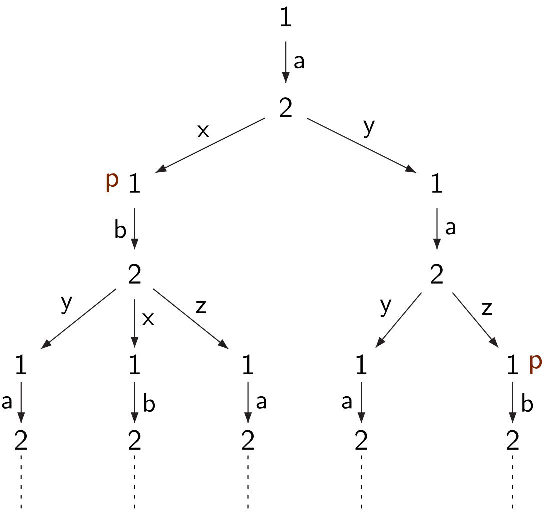
- Players have an *exit* action.
- Game arena consists of a unique “terminal” node.

A game $G = (\mathcal{G}, \{\preceq^i\}_{i\in\{1,2\}})$

- $\mathcal{G}$ is the game arena.
- $\preceq^i$ is the preference relation of player i over finite plays.

Strategy conforming to a specification

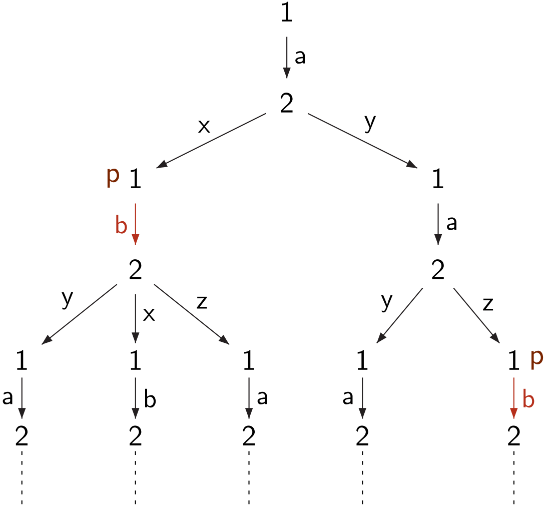
Player 1 strategy.



Strategy conforming to a specification

Player 1 strategy.

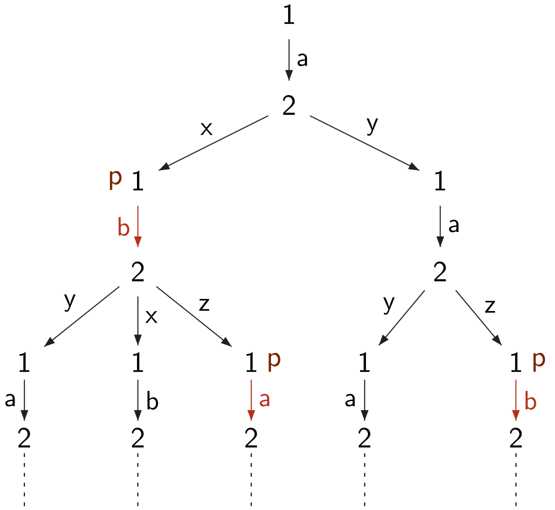
$[p \mapsto b]^1$



Strategy conforming to a specification

Player 1 strategy.

$[p \mapsto b]^1$

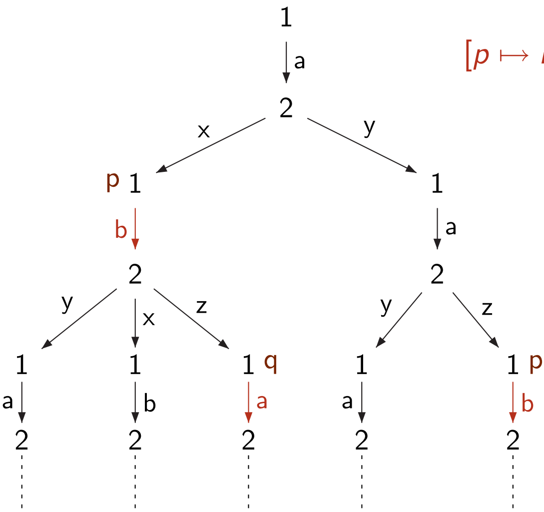


Strategy conforming to a specification

Player 1 strategy.

$\sigma_1 \cdot \sigma_2$

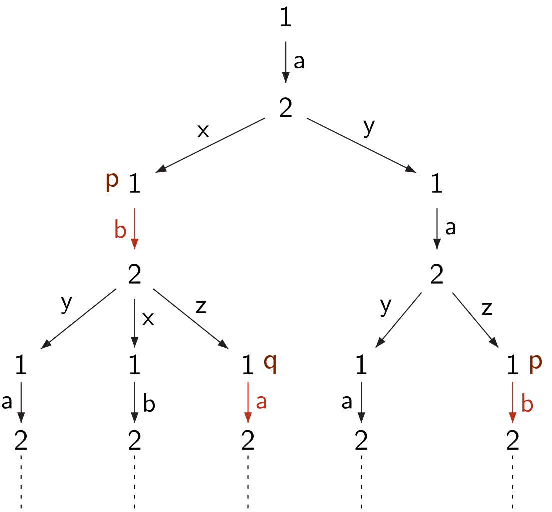
$[p \mapsto b]^1 \cdot [q \mapsto a]^1$



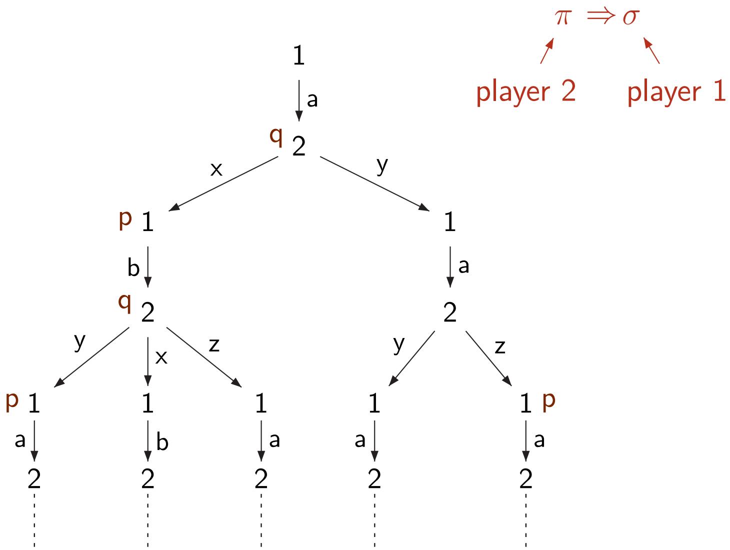
Strategy conforming to a specification

Player 1 strategy.

$\sigma_1 + \sigma_2$

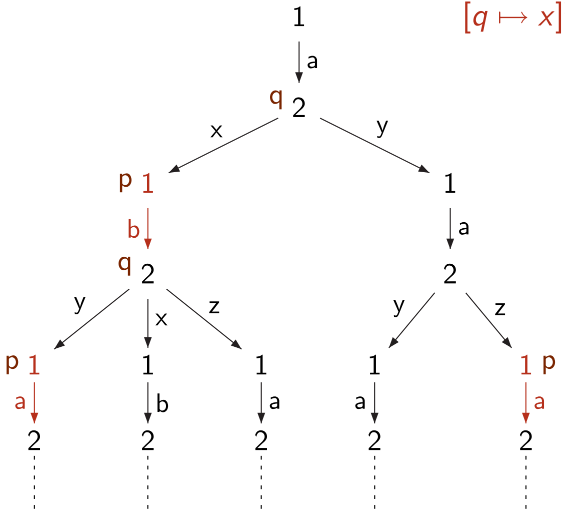


Strategy conforming to a specification



Strategy conforming to a specification

$$\pi \Rightarrow \sigma$$
$$[q \mapsto x]^2 \Rightarrow [p \mapsto b]^1$$



Strategy specification (revisited)

- Syntax: $[\psi \mapsto a]^i | \sigma_1 + \sigma_2 | \sigma_1 \cdot \sigma_2 | \pi \Rightarrow \sigma$.
- ψ- Past time formula of a simple tense logic.

Strategy comparison

Given σ, π and a condition C,

- Does 1 have a strategy conforming to σ which ensures C, as long as 2 plays a strategy conforming to π?
  $\exists \sigma, \forall \pi : C$
- If 2's strategy conforms to π is it the case that every strategy of 1 which conforms to σ ensures C.
  $\forall \sigma, \forall \pi : C$

Strategy comparison

Given σ, π and a condition C,

- Does 1 have a strategy conforming to σ which ensures C, as long as 2 plays a strategy conforming to π?
  $\exists\sigma, \forall\pi : C$
- If 2's strategy conforms to π is it the case that every strategy of 1 which conforms to σ ensures C.
  $\forall\sigma, \forall\pi : C$

σ is *better* than σ' against π:

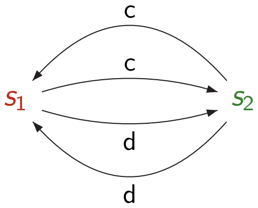
1. $\exists\sigma', \forall\pi : C \Rightarrow \exists\sigma, \forall\pi : C$
2. $\forall\sigma', \forall\pi : C \Rightarrow \forall\sigma, \forall\pi : C$
3. $\exists\sigma', \forall\pi : C \Rightarrow \forall\sigma, \forall\pi : C$

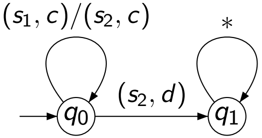
Advice Automaton

Advice Automaton $\mathcal{A} = (Q, \delta, o.I)$

- Q - set of states.
- $\delta : Q \times positions \times actions \rightarrow 2^Q$.
- I - set of initial states.
- $o : Q \times positions \rightarrow actions$.

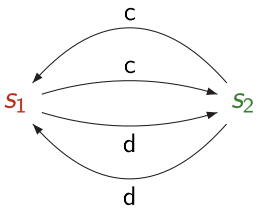
Example



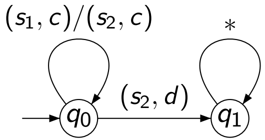


$$o(q_0, s_1) = c$$
$$o(q_1, s_1) = d$$

Example

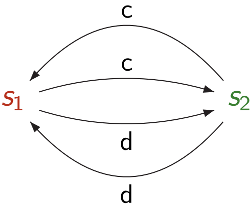


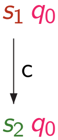
$s_1 \; q_0$

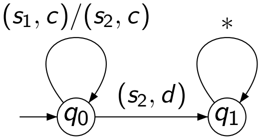


$$o(q_0, s_1) = c$$
$$o(q_1, s_1) = d$$

Example

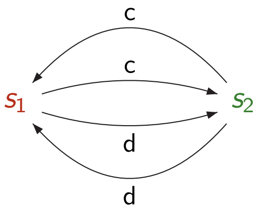


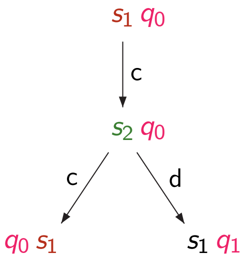


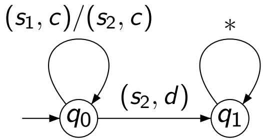


$$o(q_0, s_1) = c$$
$$o(q_1, s_1) = d$$

Example

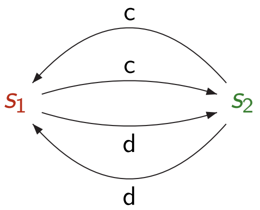


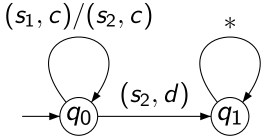




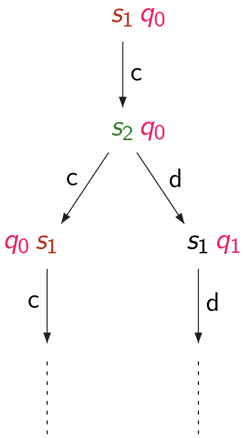
$$o(q_0, s_1) = c$$
$$o(q_1, s_1) = d$$

Example





$$o(q_0, s_1) = c$$
$$o(q_1, s_1) = d$$



Evaluation Automaton

The preference relation on finite plays for each player is presented as an *evaluation* automaton:

$\mathcal{E} = (U, \Delta, I, F, \{\preceq^i\}_{i \in \{1,2\}})$.

- U - set of states.
- $\Delta : Q \times positions \times actions \rightarrow 2^Q$.
  - $\forall q, s, a, \Delta(q, s, a) \in F$ iff $a = exit$.
- I - set of initial states.
- F - set of final states.
- $\preceq^i \subseteq (F \times F)$ is the preference relation for player i over F.

Evaluation Automaton

Win-loss objectives for player i can be modelled easily.

$\mathcal{E} = (U, \Delta, I, F, \{\preceq^i\}_{i \in \{1,2\}})$

- $F = \{f_0, f_1\}$.
- $f_0 \preceq^i f_1$.
- All plays ending in state f_1, are taken to be winning for i.

Verification question

Given an arena $\mathcal{G}$ and a deterministic "win-loss" evaluation automaton $\mathcal{E}$,

- Check if $\exists\sigma, \forall\pi : \mathcal{E}$ holds.
- Check if $\forall\sigma, \forall\pi : \mathcal{E}$ holds.

Advice automaton lemma: Given a strategy specification σ, we can construct an advice automaton $\mathcal{A}_\sigma$ such that $\mathit{Lang}(\mathcal{A}_\sigma)$ is the set of all strategies that conform to σ.

Verification question

Given an arena $\mathcal{G}$ and a deterministic "win-loss" evaluation automaton $\mathcal{E}$, check if $\exists\sigma, \forall\pi : \mathcal{E}$ holds.

- $\sigma \rightarrow \mathcal{A}_\sigma$, $\pi \rightarrow \mathcal{A}_\pi$.
- Construct $\mathcal{G} \upharpoonright \mathcal{A}_\pi$.
- Construct a nondeterministic tree automaton $\mathcal{T}$
  - States - states of $\mathcal{A}_\sigma$.
  - Guesses a state for each player 1 game position.
  - Branches out on all player 2 game positions.
  - $\mathcal{T}$ runs $\mathcal{E}$ in parallel and checks if all paths are "winning" for player 1.

Verification question

Given an arena $\mathcal{G}$ and a deterministic "win-loss" evaluation automaton $\mathcal{E}$, check if $\forall\sigma, \forall\pi : \mathcal{E}$ holds.

- $\sigma \rightarrow \mathcal{A}_\sigma$, $\pi \rightarrow \mathcal{A}_\pi$.
- Construct $(\mathcal{G} \upharpoonright \mathcal{A}_\pi) \upharpoonright \mathcal{A}_\sigma$.
- Check if all paths are "winning" for player 1 according to $\mathcal{E}$.

Best response verification

Given $\mathcal{G}$, a deterministic evaluation automaton $\mathcal{E}$, σ and π, check if σ is the best response to π.

- Enumerate final states of $\mathcal{E}$ according to 1's preference ordering.
- Use $\mathcal{T}$ to find the "best" final state f_j which 1 can ensure.
- Construct $\mathcal{E}_j$ with f_j being the most preferred final state for player 1.
- Check if $\exists\sigma, \forall\pi : \mathcal{E}_j$ holds.

Synthesis question

Synthesis question: Given $\mathcal{G}$, π and $\mathcal{E}$, synthesize a deterministic advice automaton $\mathcal{A}$ such that $\mathcal{A}, \forall\pi : \mathcal{E}$ holds.

Synthesis

Proposition

Given a deterministic advice automaton $\mathcal{A}_2$ for player 2, we can synthesise a deterministic advice automaton $\mathcal{A}$ such that $\mathcal{A}, \forall\pi : \mathcal{E}$ holds.

Proof Idea

- Consult $\mathcal{A}_2$ to pick player 2 moves.
- Guess player 1 moves.
- Check if the resulting path is winning for player 1 according to $\mathcal{E}$.

Synthesis

Synthesis question: Given $\mathcal{G}$, π and $\mathcal{E}$, synthesize a deterministic advice automaton $\mathcal{A}$ such that $\mathcal{A}, \forall\pi : \mathcal{E}$ holds.

- $\mathcal{A}_\pi$ is nondeterministic.
- States of $\mathcal{A}$ will be subsets of the states of $\mathcal{A}_\pi$.

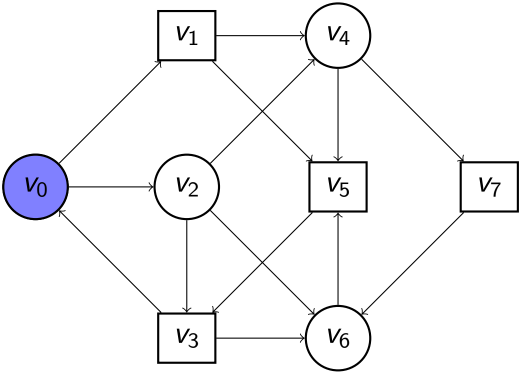
Infinite plays

Arenas and winning conditions

- Game $\mathcal{G} = (\mathcal{G}, Win)$, where
  - Arena $\mathcal{G} = (V, E)$ such that $V = V_0 \cup V_1$ where
    - V_0 are player 0 vertices, denoted by $\bigcirc$ and
    - V_1 are player 1 vertices, denoted by $\Box$
    - $E \subseteq V \times V$ is the edge relation
  - $Win \subseteq V^\omega$ is the winning condition
  - $vE = \{u \mid (v, u) \in E\}$ are the neighbours of v
- $(\mathcal{G}, v_0)$ is an initialised game where $v_0 \in V$ is a designated vertex
- We assume $vE \neq \emptyset$ for every $v \in V$

Infinite plays[2]

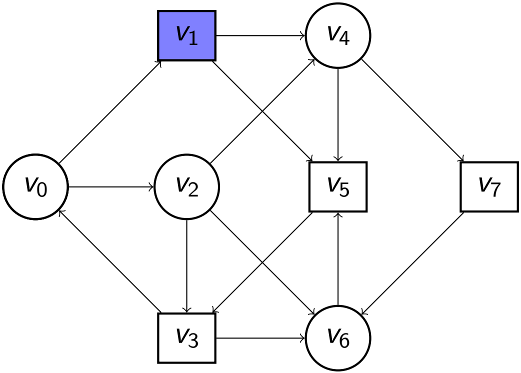
Plays



$\rho = v_0$

Infinite plays[2]

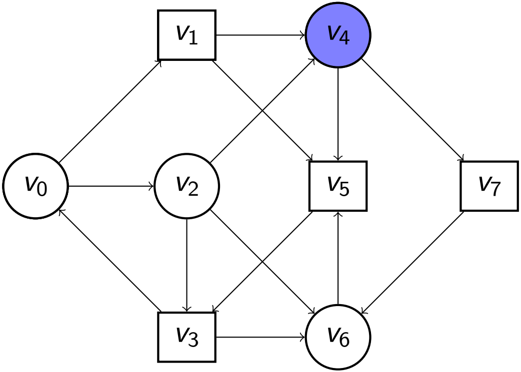
Plays



$\rho = v_0 \; v_1$

Infinite plays[2]

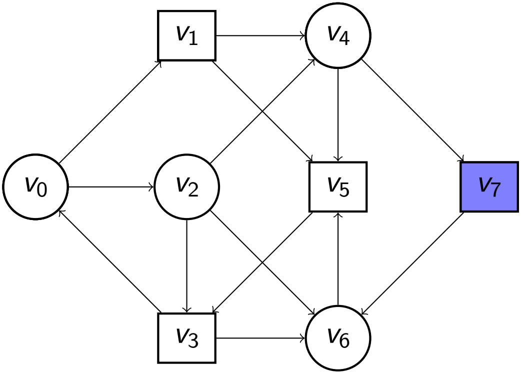
Plays



$$\rho = v_0 \ v_1 \ v_4$$

Infinite plays[2]

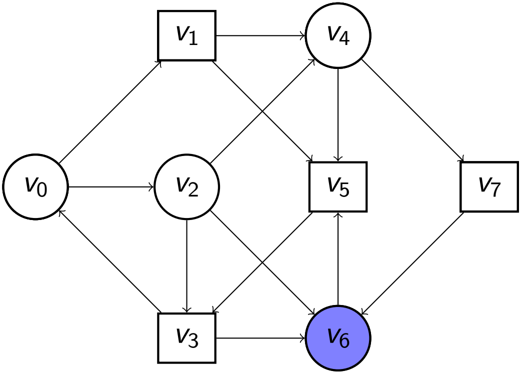
Plays



$\rho = v_0 \; v_1 \; v_4 \; v_7$

Infinite plays[2]

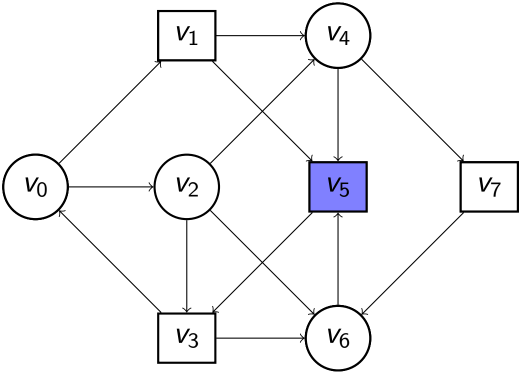
Plays



$$\rho = v_0 \; v_1 \; v_4 \; v_7 \; v_6$$

Infinite plays[2]

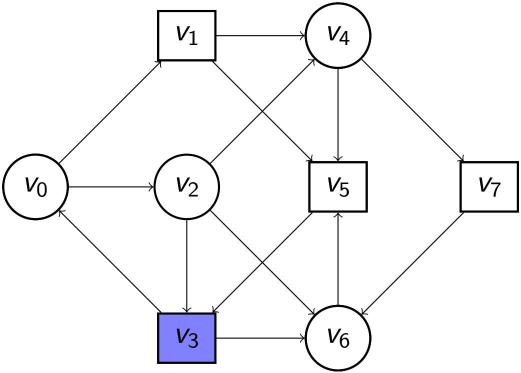
Plays



$$\rho = v_0 \; v_1 \; v_4 \; v_7 \; v_6 \; v_5$$

Infinite plays[2]

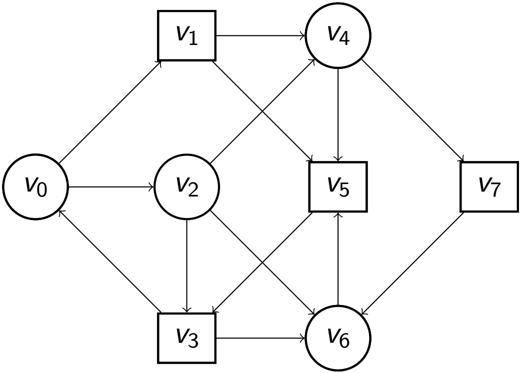
Plays



$$\rho \; = \; v_0 \;\; v_1 \;\; v_4 \;\; v_7 \;\; v_6 \;\; v_5 \;\; v_3$$

Infinite plays[2]

Plays



$$\rho = v_0 \; v_1 \; v_4 \; v_7 \; v_6 \; v_5 \; v_3 \cdots \in V^\omega$$

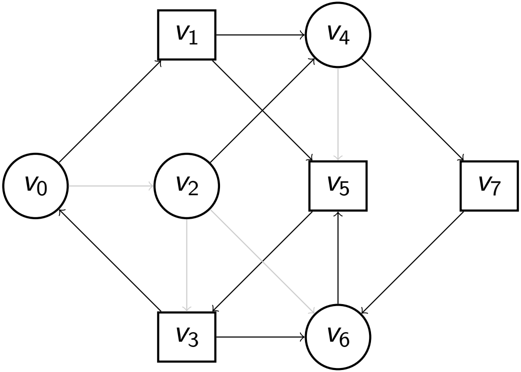
Strategies

Strategies

- A history ρ is any finite prefix of a play ρ
- A strategy *Strat* for player p, $p \in \{0, 1\}$ is a function $Strat : V^* V_p \rightarrow V$ from the set of histories to vertices such that $Strat(\rho v) \in vE$ for all ρ
- A strategy *Strat* is called finite memory or bounded memory or forgetful if it can be represented by a finite state machine
- A strategy *Strat* for player p is memoryless or positional if it does not depend on the history. That is $Strat : V_p \rightarrow V$

Memoryless strategies

Memoryless strategy as a subgraph



Determinacy

Zero-sum or win-lose games and determinacy

- A play ρ is winning for player 0 and losing for player 1 if $\rho \in Win$. Otherwise it is winning for player 1 and losing for player 0
- A strategy *Strat* of player 0 is winning if and only if all plays played according to *Strat* are winning for her
- The winning region of player p, W_p is a subset of V such that for every vertex $v \in W_p$, player p has a winning strategy for the game $(\mathcal{G}, v)$
- A game $\mathcal{G}$ is determined if $W_0 \cup W_1 = V$ and $W_0 \cap W_1 = \emptyset$

Theorem (Martin '75)

Every game where Win is a Borel set is determined

Muller condition

Muller games

- For a play ρ, $\inf(\rho)$ denotes the set of vertices occuring infinitely often in ρ
- The winning condition of a Muller game is specified by a family of subsets of vertices $\mathcal{F} \subseteq 2^V$
- A play $\rho \in \mathit{Win}$ if and only if $\inf(\rho) \in \mathcal{F}$

Theorem (Büchi and Landweber '69)

Muller games are bounded memory determined

Multiplayer Games

Binary or win-lose objectives

- $V = V_0 \cup V_1 \ldots V_n$ and $Win = (Win_1, Win_2, \ldots, Win_n)$
- Player i wins a play ρ if and only if $\rho \in Win_i$. Otherwise she loses
- $Win_i \cap Win_j$ may be non-empty for $i \neq j$

Nash equilibrium

Best-response and Nash equilibrium

- For a strategy tuple $\bar{\sigma} = (Strat_1, Strat_2, \ldots, Strat_n)$, $\rho_{\bar{\sigma}}$ denotes a play consistent with $\bar{\sigma}$. $\bar{\sigma}_{-i}$ denotes the tuple $(Strat_1, \ldots, Strat_{i-1}, Strat_{i+1}, \ldots, Strat_n)$ and for a player i strategy $Strat'_i$, $(\bar{\sigma}_{-i}, Strat'_i)$ denotes the tuple $(Strat_1, \ldots, Strat_{i-1}, Strat'_i, Strat_{i+1}, \ldots, Strat_n)$
- A strategy $Strat_i$ of player i is said to be a best response to $\bar{\sigma}_{-i}$ if for all player i strategies $Strat'_i$, $\rho_{(\bar{\sigma}_{-i}, Strat'_i)} \in Win_i$ implies $\rho_{\bar{\sigma}} \in Win_i$
- A strategy tuple $\bar{\sigma}$ is a Nash equilibrium if for all i, $Strat_i$ is the best response to $\bar{\sigma}_{-i}$.

Sequentiality

Subgame perfection

- Let ρ be a finite path in the arena. Given a strategy $Strat_i$ of player i, the strategy $Strat_i[\rho]$ is defined to be a function: $Strat_i[\rho] : \rho V^* V_i \to V$ such that $Strat_i[\rho](\rho') = Strat_i(\rho\rho')$. Let $\bar{\sigma}[\rho]$ denote the tuple $(Strat_1[\rho], \ldots, Strat_n[\rho])$. A strategy tuple $\bar{\sigma}$ in the game $(\mathcal{G}, v_0)$ is said to be a subgame perfect equilibrium if for every vertex v in $\mathcal{G}$ and for every path ρ from v_0 to v in $\mathcal{G}$, $\bar{\sigma}[\rho]$ is a Nash equilibrium for the game $(\mathcal{G}, v)$.

Results

Existence of Nash and subgame perfect equilibria

Theorem (Chatterjee, Jurdzinski, Mazumdar '04)

Every multiplayer win-lose game with Borel winning conditions has a Nash equilibrium

Theorem (Ummels '05)

Every multiplayer win-lose Muller game has a subgame perfect equilibrium

Proof.

Use threat strategies $\square$

Threat and punishment

Threat strategies

- Let $(\mathcal{G}, v_0)$ be a multiplayer game where the winning sets Win_i are Borel
- For each player i define the game $(\mathcal{G}_i, v_0)$ in which player i plays against the coalition of all the other players and her winning set is Win_i whereas that of the coalition is $V^\omega \setminus Win_i$
- By Martin's theorem $\mathcal{G}_i$ is determined and player i and the coalition have optimal strategies $Strat_i$ and $Strat_{-i}$ respectively
- At Nash equilibrium:
  - Every player i plays strategy $Strat_i$ till no one deviates
  - As soon as player j deviates all the other players switch to their respective strategies induced by $Strat_{-j}$

Overlapping objectives

Generalised objectives

- Not win-lose
- Each player has a preference over the various plays
- Generalised Muller games: each player i has a (total) preference over the various Muller sets which can be described by a utility function $u_i : \mathcal{F} \rightarrow \mathbb{N}$. For a play ρ player i's utility is $u_i(\inf(\rho))$

Does a Nash equilibrium always exist in a generalised Muller game? If so, is it subgame perfect?

Generalised Muller Games

Existence of Nash equilibrium

- Threat strategies can be used to show that a Nash equilibrium always exists in a generalised Muller game
- Is threat a viable strategy?
- Problems arise in the finite case itself
- In the infinite case, players have to play punishing moves forever

	L	R
U	(2,2)	(0,3)
D	(2,0)	(0,-1000)

Generalised Muller Games[2]

We show

- Existence of Nash equilibrium in generalised Muller games by using a backward induction procedure
- Subgame perfect equilibrim may not exist for such games in general
- Our procedure finds a subgame perfect equilibrium when it exists

Generalised Muller Games[3]

The Latest Appearance Record (LAR) data structure (Gurevich, Harrington'82)

- The set of LARs
  $L = \{x \in (V \cup \{\sharp\})^{|V|+1} \mid |x|_\sharp = 1 \text{ and } |x|_v = 1,\ \forall v \in V\}$
- $\langle N \rangle : L \times V \to L$ as

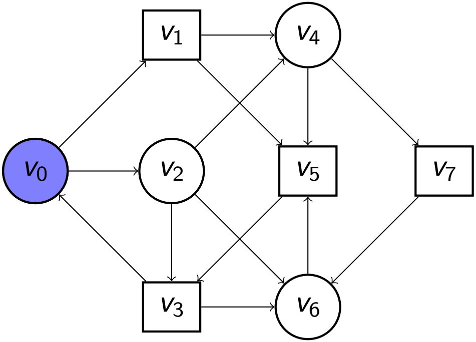
$$\langle N \rangle(x\sharp y, v) = \begin{cases} x'\sharp x'' y v & \text{iff } x\sharp y = x' v x'' \sharp y \\ x y' \sharp y'' v & \text{iff } x\sharp y = x \sharp y' v y'' \\ x\sharp y & \text{iff } x\sharp y = x\sharp y' v \end{cases}$$

- Let $\prec$ be a total order on V. For a finite play $\rho = v_0 v_1 \dots v_k$ define LAR(ρ) inductively as:
  - LAR(v_0)=$x\sharp v_0$ where x is ordered according to $\prec$
  - LAR($v_0 \dots v_i$)=$\langle N \rangle$(LAR($v_0 \dots v_{i-1}$),v_i), $i \geq 1$

Generalised Muller Games[4]

Example

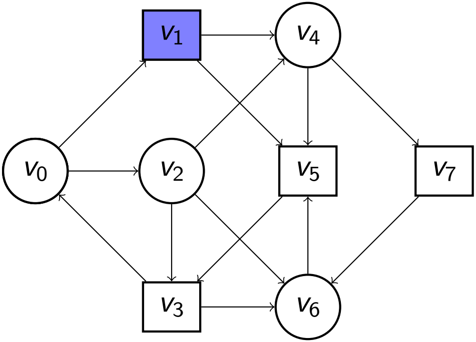
$v_1 v_2 v_3 v_4 v_5 v_6 v_7 \sharp v_0$



$\rho = v_0$

Generalised Muller Games[4]

Example

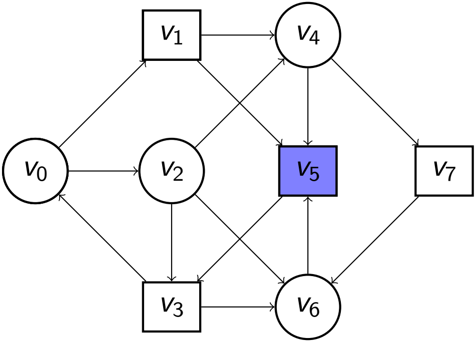


$v_1\, v_2\, v_3\, v_4\, v_5\, v_6\, v_7 \sharp v_0$
$\sharp v_2\, v_3\, v_4\, v_5\, v_6\, v_7\, v_0\, v_1$

$\rho = v_0\ v_1$

Generalised Muller Games[4]

Example

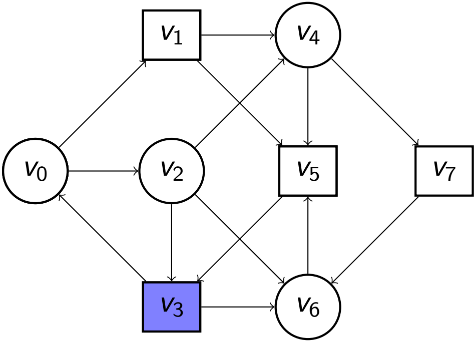


$v_1\, v_2\, v_3\, v_4\, v_5\, v_6\, v_7 \sharp v_0$
$\sharp v_2\, v_3\, v_4\, v_5\, v_6\, v_7\, v_0\, v_1$
$v_2\, v_3\, v_4 \sharp v_6\, v_7\, v_0\, v_1\, v_5$

$\rho = v_0\ v_1\ v_5$

Generalised Muller Games[4]

Example

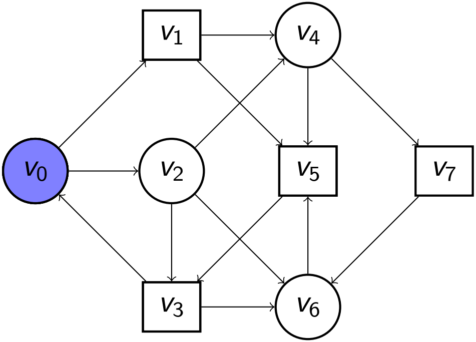


$\rho = v_0 \; v_1 \; v_5 \; v_3$

$v_1 \, v_2 \, v_3 \, v_4 \, v_5 \, v_6 \, v_7 \sharp v_0$
$\sharp v_2 \, v_3 \, v_4 \, v_5 \, v_6 \, v_7 \, v_0 \, v_1$
$v_2 \, v_3 \, v_4 \sharp v_6 \, v_7 \, v_0 \, v_1 \, v_5$
$v_2 \sharp v_4 \, v_6 \, v_7 \, v_0 \, v_1 \, v_5 \, v_3$

Generalised Muller Games[4]

Example

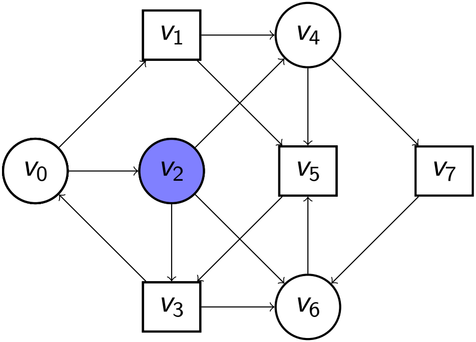


$v_1\, v_2\, v_3\, v_4\, v_5\, v_6\, v_7 \sharp v_0$
$\sharp v_2\, v_3\, v_4\, v_5\, v_6\, v_7\, v_0\, v_1$
$v_2\, v_3\, v_4 \sharp v_6\, v_7\, v_0\, v_1\, v_5$
$v_2 \sharp v_4\, v_6\, v_7\, v_0\, v_1\, v_5\, v_3$
$v_2\, v_4\, v_6\, v_7 \sharp v_1\, v_5\, v_3\, v_0$

$\rho = v_0\ v_1\ v_5\ v_3\ v_0$

Generalised Muller Games[4]

Example

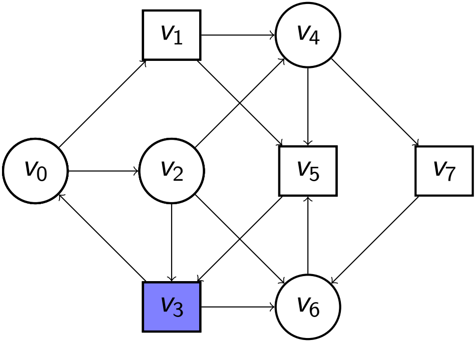


$\rho = v_0\ v_1\ v_5\ v_3\ v_0\ v_2$

$v_1 v_2 v_3 v_4 v_5 v_6 v_7 \sharp v_0$
$\sharp v_2 v_3 v_4 v_5 v_6 v_7 v_0 v_1$
$v_2 v_3 v_4 \sharp v_6 v_7 v_0 v_1 v_5$
$v_2 \sharp v_4 v_6 v_7 v_0 v_1 v_5 v_3$
$v_2 v_4 v_6 v_7 \sharp v_1 v_5 v_3 v_0$
$\sharp v_4 v_6 v_7 v_1 v_5 v_3 v_0 v_2$

Generalised Muller Games[4]

Example

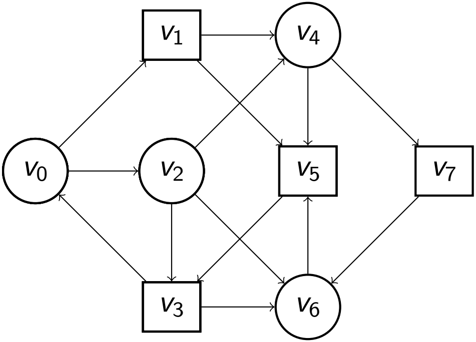


$v_1\, v_2\, v_3\, v_4\, v_5\, v_6\, v_7 \sharp v_0$
$\sharp v_2\, v_3\, v_4\, v_5\, v_6\, v_7\, v_0\, v_1$
$v_2\, v_3\, v_4 \sharp v_6\, v_7\, v_0\, v_1\, v_5$
$v_2 \sharp v_4\, v_6\, v_7\, v_0\, v_1\, v_5\, v_3$
$v_2\, v_4\, v_6\, v_7 \sharp v_1\, v_5\, v_3\, v_0$
$\sharp v_4\, v_6\, v_7\, v_1\, v_5\, v_3\, v_0\, v_2$
$v_4\, v_6\, v_7\, v_1\, v_5 \sharp v_0\, v_2\, v_3$

$\rho = v_0\ v_1\ v_5\ v_3\ v_0\ v_2\ v_3$

Generalised Muller Games[4]

Example

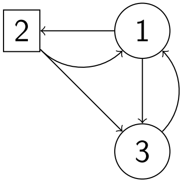


$v_1\, v_2\, v_3\, v_4\, v_5\, v_6\, v_7 \sharp v_0$
$\sharp v_2\, v_3\, v_4\, v_5\, v_6\, v_7\, v_0\, v_1$
$v_2\, v_3\, v_4 \sharp v_6\, v_7\, v_0\, v_1\, v_5$
$v_2 \sharp v_4\, v_6\, v_7\, v_0\, v_1\, v_5\, v_3$
$v_2\, v_4\, v_6\, v_7 \sharp v_1\, v_5\, v_3\, v_0$
$\sharp v_4\, v_6\, v_7\, v_1\, v_5\, v_3\, v_0\, v_2$
$v_4\, v_6\, v_7\, v_1\, v_5 \sharp v_0\, v_2\, v_3$
$\vdots$

$$\rho = v_0\ v_1\ v_5\ v_3\ v_0\ v_2\ v_3 \cdots \in V^\omega$$

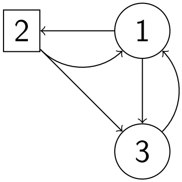
Generalised Muller Games[5]

The LAR tree



Generalised Muller Games[5]

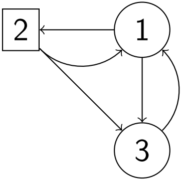
The LAR tree



$$u(\{1,2,3\}) = (1,3)$$
$$u(\{2,1\}) = (2,1)$$
$$u(\{1,3\}) = (2,2)$$

Generalised Muller Games[5]

The LAR tree



$$u(\{1,2,3\}) = (1,3)$$
$$u(\{2,1\}) = (2,1)$$
$$u(\{1,3\}) = (2,2)$$

$23\sharp1$

$\sharp312$ $2\sharp13$

$3\sharp21$ $\sharp123$ $2\sharp31$

$3\sharp12$ $\sharp213$ $\sharp231$ $2\sharp13$ $\sharp312$

$3\sharp21$ $\sharp123$ $2\sharp31$ $\sharp312$ $2\sharp13$ $3\sharp21$ $\sharp123$

$\sharp231$ $2\sharp13$ $\sharp312$ $2\sharp31$ $3\sharp12$ $\sharp213$ $\sharp231$

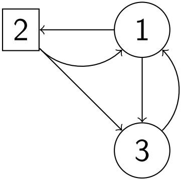
$\sharp312$ $2\sharp13$ $2\sharp31$ $\sharp312$ $2\sharp13$ $3\sharp21$ $\sharp123$ $2\sharp31$ $\sharp312$ $2\sharp13$

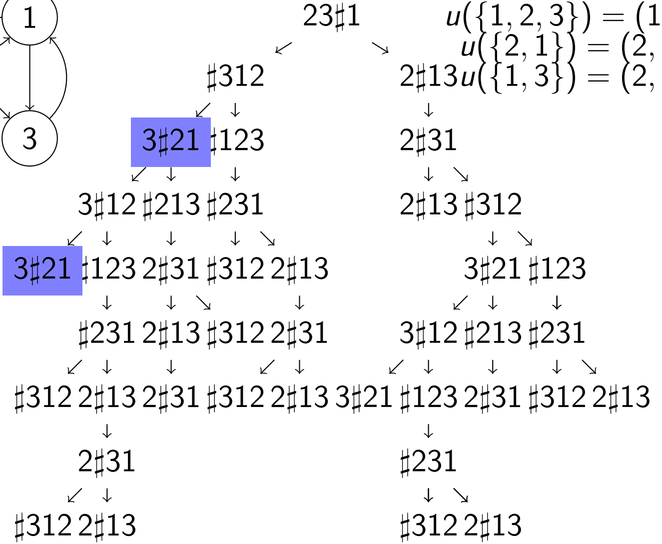
$2\sharp31$ $\sharp231$

$\sharp312$ $2\sharp13$ $\sharp312$ $2\sharp13$

Generalised Muller Games[5]

The LAR tree





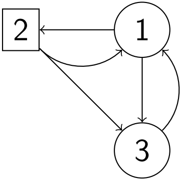
$u(\{1,2,3\}) = (1,3)$

$u(\{2,1\}) = (2,1)$

$u(\{1,3\}) = (2,2)$

Generalised Muller Games[5]

The LAR tree



$$u(\{1,2,3\}) = (1,3)$$
$$u(\{2,1\}) = (2,1)$$
$$u(\{1,3\}) = (2,2)$$

$23\sharp1$

$\sharp312$ $2\sharp13$

$3\sharp21$ $\sharp123$ $2\sharp31$

$3\sharp12$ $\sharp213$ $\sharp231$ $2\sharp13$ $\sharp312$

$3\sharp21$ $\sharp123$ $2\sharp31$ $\sharp312$ $2\sharp13$ $3\sharp21$ $\sharp123$

$\sharp231$ $2\sharp13$ $\sharp312$ $2\sharp31$ $3\sharp12$ $\sharp213$ $\sharp231$

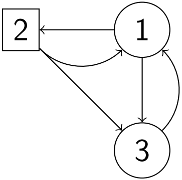
$\sharp312$ $2\sharp13$ $2\sharp31$ $\sharp312$ $2\sharp13$ $3\sharp21$ $\sharp123$ $2\sharp31$ $\sharp312$ $2\sharp13$

$2\sharp31$ $\sharp231$

$\sharp312$ $2\sharp13$ $\sharp312$ $2\sharp13$

Generalised Muller Games[5]

The LAR tree



$u(\{1,2,3\}) = (1,3)$
$u(\{2,1\}) = (2,1)$
$u(\{1,3\}) = (2,2)$

$23\sharp1$

$\sharp312 \quad 2\sharp13$

$3\sharp21 \quad \sharp123 \quad 2\sharp31$

$3\sharp12 \quad \sharp213 \quad \sharp231 \quad 2\sharp13 \quad \sharp312$

$3\sharp21 \quad \sharp123 \quad 2\sharp31 \quad \sharp312 \quad 2\sharp13 \quad 3\sharp21 \quad \sharp123$

$\sharp231 \quad 2\sharp13 \quad \sharp312 \quad 2\sharp31 \quad 3\sharp12 \quad \sharp213 \quad \sharp231$

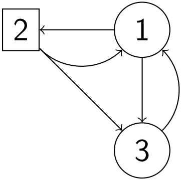
$\sharp312 \quad 2\sharp13 \quad 2\sharp31 \quad \sharp312 \quad 2\sharp13 \quad 3\sharp21 \quad \sharp123 \quad 2\sharp31 \quad \sharp312 \quad 2\sharp13$

$2\sharp31 \quad \sharp231$

$\sharp312 \quad 2\sharp13 \quad \sharp312 \quad 2\sharp13$

Generalised Muller Games[5]

The LAR tree



$$u(\{1,2,3\}) = (1,3)$$
$$u(\{2,1\}) = (2,1)$$
$$u(\{1,3\}) = (2,2)$$

$23\sharp1$

$\sharp312 \quad 2\sharp13$

$\boxed{3\sharp21} \quad \sharp123 \quad 2\sharp31$

$3\sharp12 \quad \sharp213 \quad \sharp231 \quad 2\sharp13 \quad \sharp312$

$\boxed{3\sharp21} \quad \sharp123 \quad 2\sharp31 \quad \sharp312 \quad 2\sharp13 \quad 3\sharp21 \quad \sharp123$

2,1

$\sharp231 \quad 2\sharp13 \quad \sharp312 \quad 2\sharp31 \quad 3\sharp12 \quad \sharp213 \quad \sharp231$

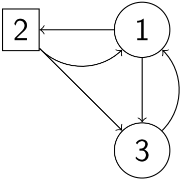
$\sharp312 \quad 2\sharp13 \quad 2\sharp31 \quad \sharp312 \quad 2\sharp13 \quad 3\sharp21 \quad \sharp123 \quad 2\sharp31 \quad \sharp312 \quad 2\sharp13$

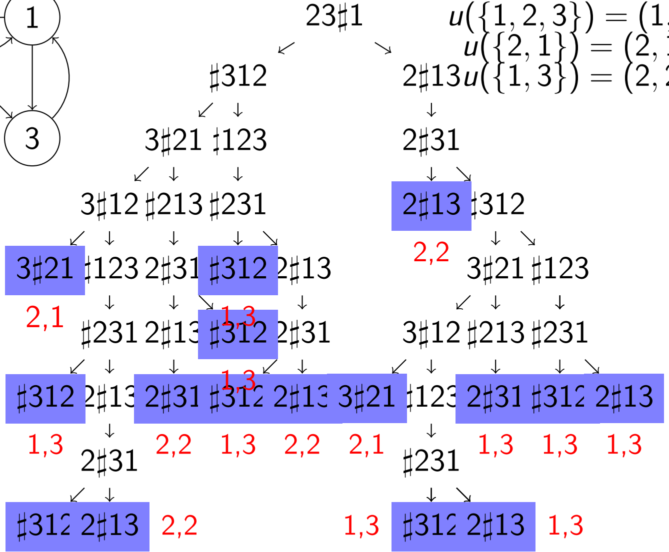
$2\sharp31 \quad \sharp231$

$\sharp312 \quad 2\sharp13 \quad \sharp312 \quad 2\sharp13$

Generalised Muller Games[5]

The LAR tree

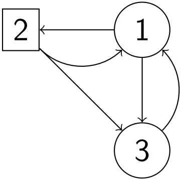


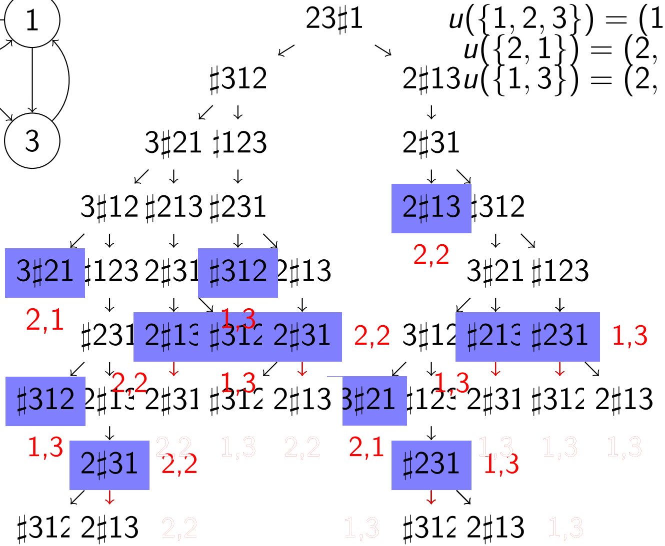


$$u(\{1,2,3\}) = (1,3)$$
$$u(\{2,1\}) = (2,1)$$
$$u(\{1,3\}) = (2,2)$$

Generalised Muller Games[5]

The LAR tree

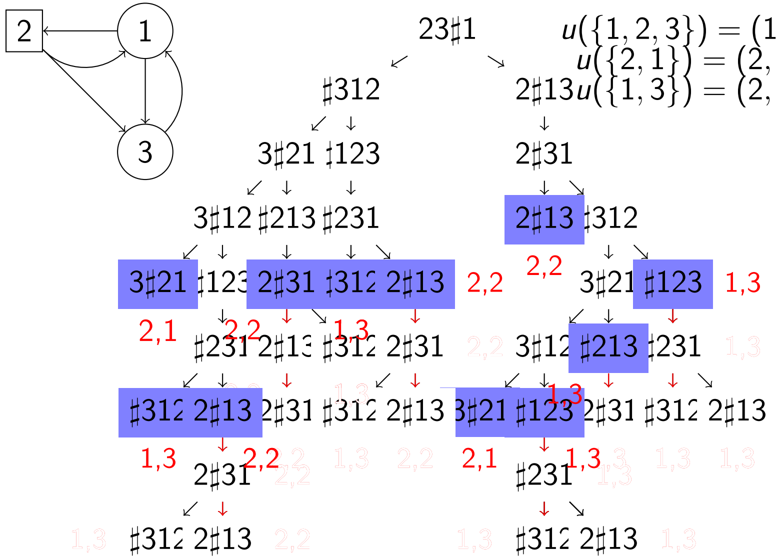




$u(\{1,2,3\}) = (1,3)$
$u(\{2,1\}) = (2,1)$
$u(\{1,3\}) = (2,2)$

Generalised Muller Games[5]

The LAR tree

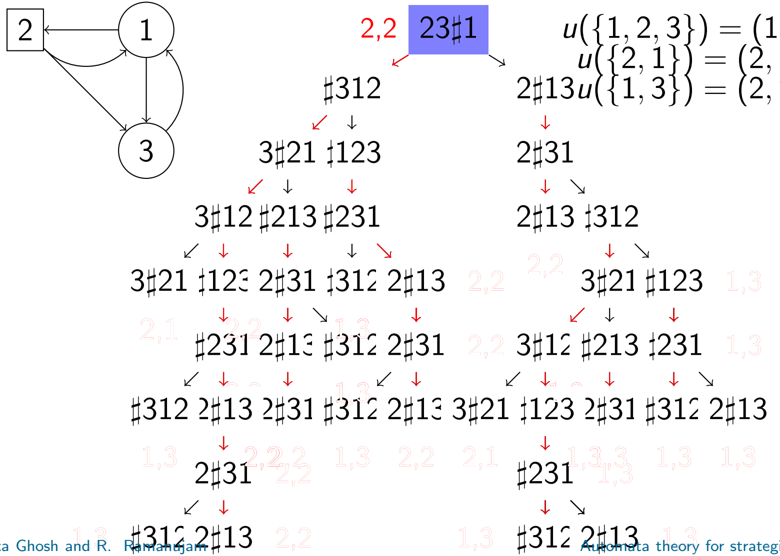


$u(\{1,2,3\}) = (1,3)$
$u(\{2,1\}) = (2,1)$
$u(\{1,3\}) = (2,2)$

Generalised Muller Games[5]

The LAR tree

$u(\{1,2,3\}) = (1,3)$
$u(\{2,1\}) = (2,1)$
$u(\{1,3\}) = (2,2)$



Generalised Muller Games[6]

Memoryless strategy μ of player to bounded memory strategy *Strat* of player i

- The memory M of *Strat* is the set L and the initial memory m_I is the root of the LAR tree
- The memory update function $g : V \times M \rightarrow M$ is $g(v, x\sharp y) = \langle N \rangle (x\sharp y, v)$
- The output function $h : V_i \times M \rightarrow V$ is $h(v, x\sharp y) = \mu(x\sharp y)$

Generalised Muller Games[7]

Lemma

The bounded memory strategy tuple $\bar{\sigma}$ corresponding to the memoryless strategy tuple $\bar{\mu}$ derived from backward induction on the LAR tree is a Nash equilibrium

Proof.

A player i has an incentive to deviate from μ_i on the LAR tree if and only if she has an incentive to deviate from $Strat_i$ in the arena □

Theorem

Every finite generalised Muller game has a Nash equilibrium in bounded memory strategies

Subgame Perfection

Theorem

In a finite multiplayer win-lose Muller game, the bounded memory strategy tuple $\bar{\sigma}$ corresponding to the memoryless strategy tuple $\bar{\mu}$ derived from backward induction on the LAR tree is subgame perfect

Proof.

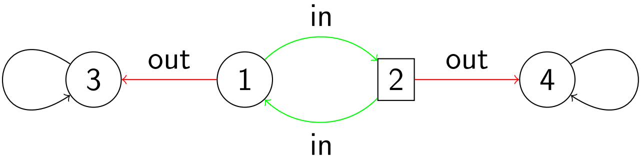
After every finite path ρ, a player i has an incentive to deviate from $Strat_i[\rho]$ in the arena if and only if she has an incentive to deviate from $\mu_i[\mathsf{LAR}(\rho)]$ on the LAR tree $\square$

Corollary (Ummels '05)

Every finite multiplayer win-lose Muller game has a subgame perfect equilibrium

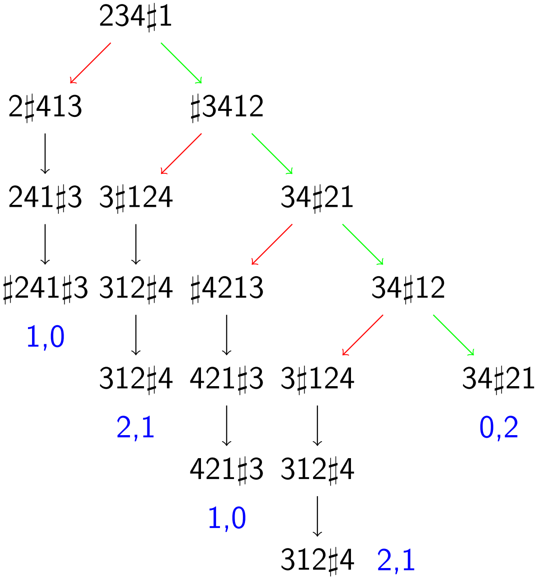
Subgame Perfection[2]

What about generalised Muller games?

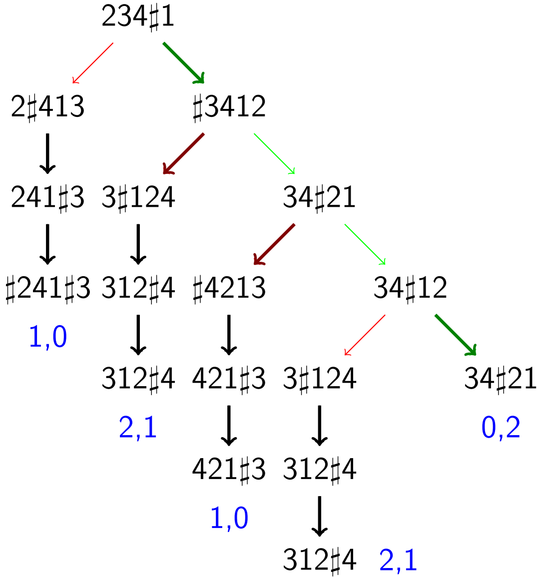


$$u(\{3\}) = (1, 0)$$
$$u(\{1, 2\}) = (0, 2)$$
$$u_1(\{4\}) = (2, 1)$$

Subgame Perfection[3]

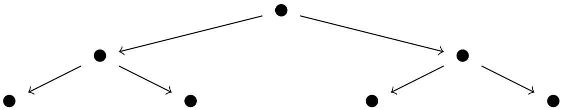


Subgame Perfection[3]



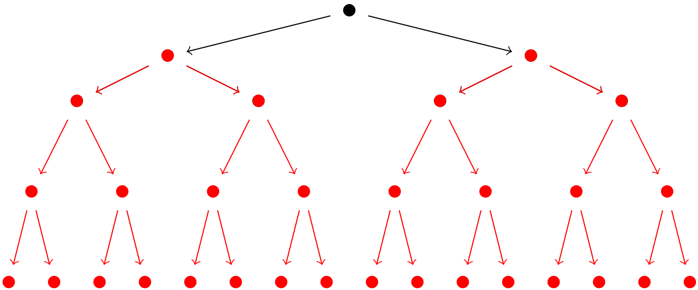
Subgame Perfection[4]

Explore the LAR tree further



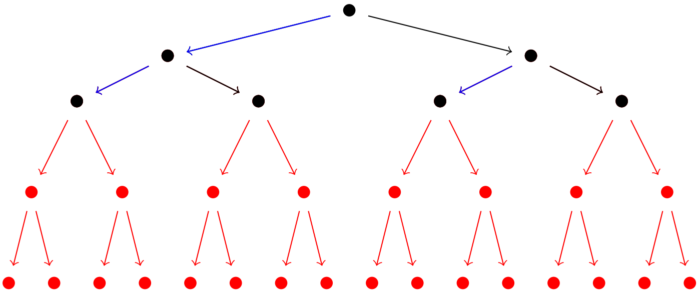
Subgame Perfection[4]

Explore the LAR tree further



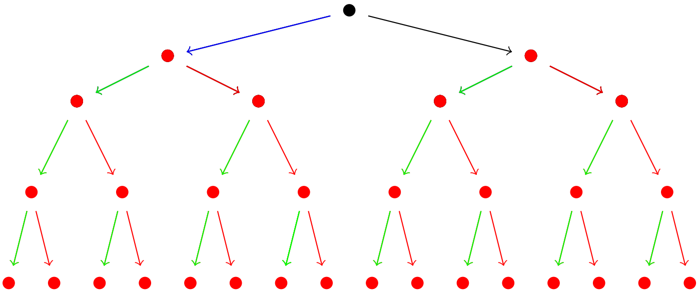
Subgame Perfection[4]

Explore the LAR tree further



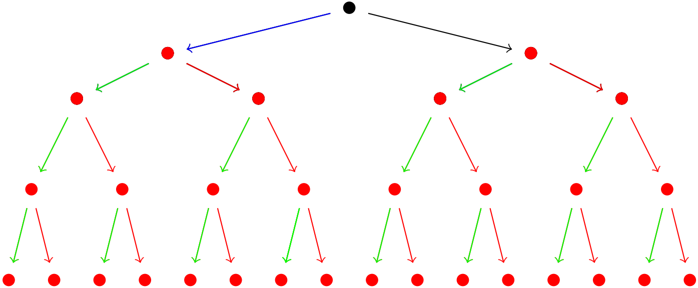
Subgame Perfection[4]

Explore the LAR tree further



Subgame Perfection[4]

Explore the LAR tree further



Theorem

It can be effectively decided whether a generalised Muller game has a subgame perfect equilibrium.