From finite-memory winning strategies to finite-memory Nash equilibria

Stéphane Le Roux, joint work with Arno Pauly

Université libre de Bruxelles, inVest project

GT ALGA Marseille 12 April 2016

Questions on system verification as "simple" games (finite graphs).

Questions on system verification as "simple" games (finite graphs).

Theorem (Gurevich and Harrington 1982)

Simple Muller games have finite-memory solutions.

Questions on system verification as "simple" games (finite graphs).

Theorem (Gurevich and Harrington 1982)

Simple Muller games have finite-memory solutions.

Questions on distributed-system verification as "complex" games.

Questions on system verification as "simple" games (finite graphs).

Theorem (Gurevich and Harrington 1982)

Simple Muller games have finite-memory solutions.

Questions on distributed-system verification as "complex" games.

Theorem (Paul and Simon 2009)

Complex Muller games have finite-memory solutions.

Questions on system verification as "simple" games (finite graphs).

Theorem (Gurevich and Harrington 1982)

Simple Muller games have finite-memory solutions.

Questions on distributed-system verification as "complex" games.

Theorem (Paul and Simon 2009)

Complex Muller games have finite-memory solutions.

How about a transfer theorem?

Questions on system verification as "simple" games (finite graphs).

Theorem (Gurevich and Harrington 1982)

Simple Muller games have finite-memory solutions.

Questions on distributed-system verification as "complex" games.

Theorem (Paul and Simon 2009)

Complex Muller games have finite-memory solutions.

How about a transfer theorem?

Result by De Pril 2013, Brihaye, De Pril, and Schewe 2013.

Questions on system verification as "simple" games (finite graphs).

Theorem (Gurevich and Harrington 1982)

Simple Muller games have finite-memory solutions.

Questions on distributed-system verification as "complex" games.

Theorem (Paul and Simon 2009)

Complex Muller games have finite-memory solutions.

How about a transfer theorem?

Result by De Pril 2013, Brihaye, De Pril, and Schewe 2013.

Our transfer theorem:

- applicable to, *e.g.* , energy-parity games.

Questions on system verification as "simple" games (finite graphs).

Theorem (Gurevich and Harrington 1982)

Simple Muller games have finite-memory solutions.

Questions on distributed-system verification as "complex" games.

Theorem (Paul and Simon 2009)

Complex Muller games have finite-memory solutions.

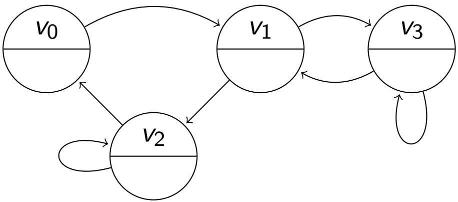
How about a transfer theorem?

Result by De Pril 2013, Brihaye, De Pril, and Schewe 2013.

Our transfer theorem:

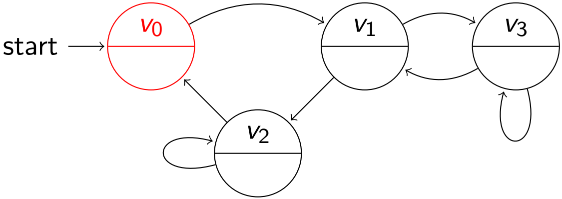
- applicable to, *e.g.* , energy-parity games.
- sufficient condition approaching necessity,

Turn-based games played on finite graphs



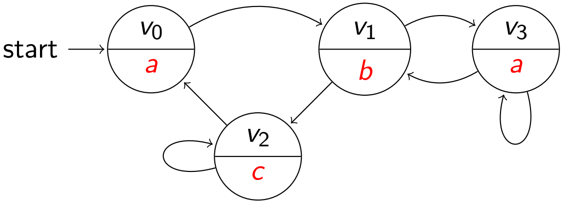
- (V, E) is a finite directed graph s.t. $vE \neq \emptyset$ for all $v \in V$.

Turn-based games played on finite graphs



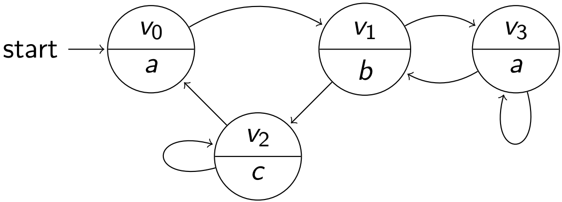
- (V, E) is a finite directed graph s.t. $vE \neq \emptyset$ for all $v \in V$.
- $v_0 \in V$ is the initial vertex.

Turn-based games played on finite graphs



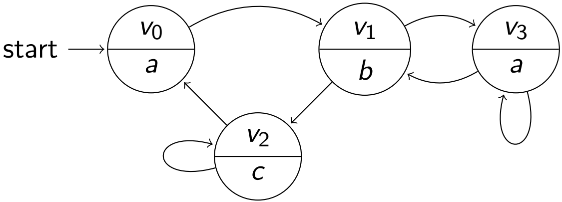
- (V, E) is a finite directed graph s.t. $vE \neq \emptyset$ for all $v \in V$.
- $v_0 \in V$ is the initial vertex.
- A is a set (of players) and $\{V_a\}_{a \in A}$ is a partition of V.

Turn-based games played on finite graphs



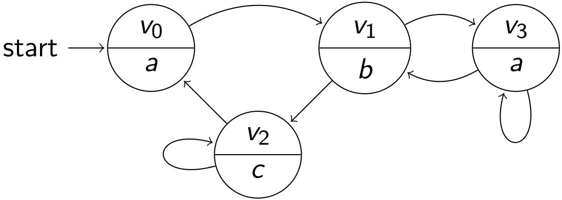
- (V, E) is a finite directed graph s.t. $vE \neq \emptyset$ for all $v \in V$.
- $v_0 \in V$ is the initial vertex.
- A is a set (of players) and $\{V_a\}_{a \in A}$ is a partition of V.
- $\mathcal{H}$ are the histories: finite paths in (V, E) starting at v_0.

Turn-based games played on finite graphs



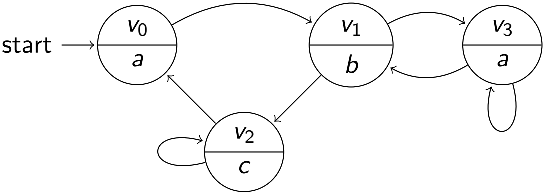
- (V, E) is a finite directed graph s.t. $vE \neq \emptyset$ for all $v \in V$.
- $v_0 \in V$ is the initial vertex.
- A is a set (of players) and $\{V_a\}_{a \in A}$ is a partition of V.
- $\mathcal{H}$ are the histories: finite paths in (V, E) starting at v_0.
- $[\mathcal{H}]$ are the runs: infinite paths in (V, E) starting at v_0.

Turn-based games played on finite graphs



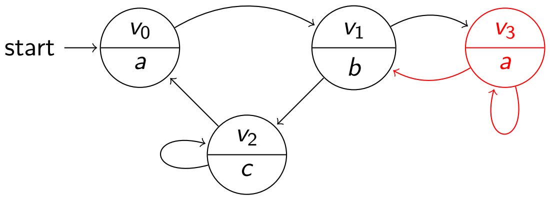
- (V, E) is a finite directed graph s.t. $vE \neq \emptyset$ for all $v \in V$.
- $v_0 \in V$ is the initial vertex.
- A is a set (of players) and $\{V_a\}_{a \in A}$ is a partition of V.
- $\mathcal{H}$ are the histories: finite paths in (V, E) starting at v_0.
- $[\mathcal{H}]$ are the runs: infinite paths in (V, E) starting at v_0.
- $\prec_a \subseteq [\mathcal{H}] \times [\mathcal{H}]$ (is the preference of player $a \in A$).

Nash equilibrium



Def $s : \mathcal{H} \to V$ is a strategy profile iff $h \cdot s(h) \in \mathcal{H}$ for all $h \in \mathcal{H}$.

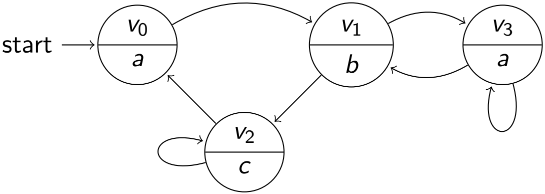
Nash equilibrium



Def $s : \mathcal{H} \to V$ is a strategy profile iff $h \cdot s(h) \in \mathcal{H}$ for all $h \in \mathcal{H}$.

E.g. $s(hv_3) \in \{v_1, v_3\}$

Nash equilibrium

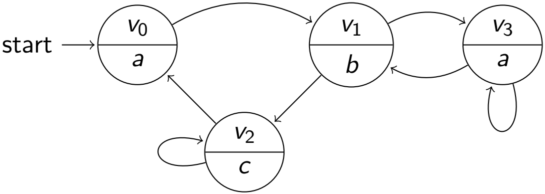


Def $s : \mathcal{H} \to V$ is a strategy profile iff $h \cdot s(h) \in \mathcal{H}$ for all $h \in \mathcal{H}$.

E.g. $s(hv_3) \in \{v_1, v_3\}$

Def A strategy profile s induces a unique run $\rho(s)$.

Nash equilibrium



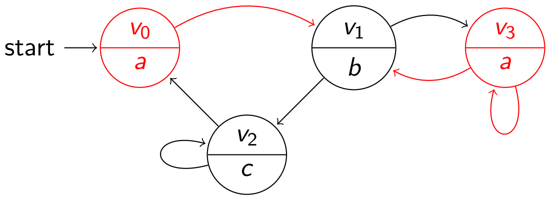
Def $s : \mathcal{H} \to V$ is a strategy profile iff $h \cdot s(h) \in \mathcal{H}$ for all $h \in \mathcal{H}$.

E.g. $s(hv_3) \in \{v_1, v_3\}$

Def A strategy profile s induces a unique run $\rho(s)$.

Notation Let $s \prec_a s'$ stand for $\rho(s) \prec_a \rho(s')$.

Nash equilibrium



Def $s : \mathcal{H} \to V$ is a strategy profile iff $h \cdot s(h) \in \mathcal{H}$ for all $h \in \mathcal{H}$.

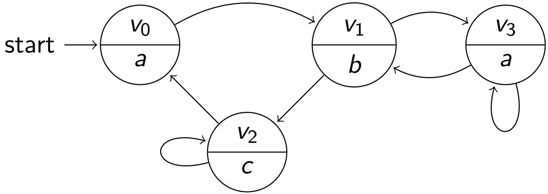
E.g. $s(hv_3) \in \{v_1, v_3\}$

Def A strategy profile s induces a unique run $\rho(s)$.

Notation Let $s \prec_a s'$ stand for $\rho(s) \prec_a \rho(s')$.

Def Let s be a profile, then $s_a := s\,|_{V^*V_a}$ is a strategy for player a.

Nash equilibrium



Def $s : \mathcal{H} \to V$ is a strategy profile iff $h \cdot s(h) \in \mathcal{H}$ for all $h \in \mathcal{H}$.

E.g. $s(hv_3) \in \{v_1, v_3\}$

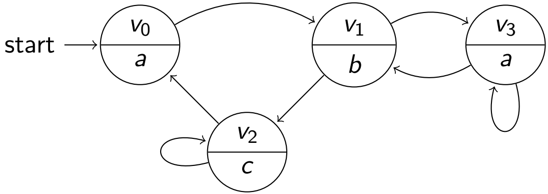
Def A strategy profile s induces a unique run $\rho(s)$.

Notation Let $s \prec_a s'$ stand for $\rho(s) \prec_a \rho(s')$.

Def Let s be a profile, then $s_a := s\,|_{V^* V_a}$ is a strategy for player a.

Def A profile $s = \cup_{b \in A} s_b$ is a Nash equilibrium iff s makes all the players stable, *i.e.* for all $a \in A$ we have

Nash equilibrium



Def $s : \mathcal{H} \to V$ is a strategy profile iff $h \cdot s(h) \in \mathcal{H}$ for all $h \in \mathcal{H}$.

E.g. $s(hv_3) \in \{v_1, v_3\}$

Def A strategy profile s induces a unique run $\rho(s)$.

Notation Let $s \prec_a s'$ stand for $\rho(s) \prec_a \rho(s')$.

Def Let s be a profile, then $s_a := s\,|_{V^* V_a}$ is a strategy for player a.

Def A profile $s = \cup_{b \in A} s_b$ is a Nash equilibrium iff
s makes all the players stable, *i.e.* for all $a \in A$ we have
$\forall s'_a,\ s \not\prec_a s'_a \cup (\cup_{b \in A \setminus \{a\}} s_b)$.

Some special cases

Usually, the preferences are defined in two stages :

1. by assigning a payoff tuple $A \to \mathbb{R}$ to each run.

Some special cases

Usually, the preferences are defined in two stages :

1. by assigning a payoff tuple $A \to \mathbb{R}$ to each run.
   If $A = \{a, b, c\}$ then $(2, 7, 4)$ means a gets 2, b gets 7...

Some special cases

Usually, the preferences are defined in two stages :

1. by assigning a payoff tuple $A \to \mathbb{R}$ to each run.
   If $A = \{a, b, c\}$ then $(2, 7, 4)$ means a gets 2, b gets 7...
2. and $(0, 2, 1) \prec_b (9, 3, 0)$.

Some special cases

Usually, the preferences are defined in two stages :

1. by assigning a payoff tuple $A \to \mathbb{R}$ to each run.
   If $A = \{a, b, c\}$ then $(2, 7, 4)$ means a gets 2, b gets 7...
2. and $(0, 2, 1) \prec_b (9, 3, 0)$.

Two-player win/lose games: only payoffs $(1, 0)$ or $(0, 1)$.

Some special cases

Usually, the preferences are defined in two stages :

1. by assigning a payoff tuple $A \to \mathbb{R}$ to each run.
   If $A = \{a, b, c\}$ then $(2, 7, 4)$ means a gets 2, b gets 7...
2. and $(0, 2, 1) \prec_b (9, 3, 0)$.

Two-player win/lose games: only payoffs $(1, 0)$ or $(0, 1)$.
Such games may have winning strategies.

Some special cases

Usually, the preferences are defined in two stages :

1. by assigning a payoff tuple $A \to \mathbb{R}$ to each run.
   If $A = \{a, b, c\}$ then $(2, 7, 4)$ means a gets 2, b gets 7...
2. and $(0, 2, 1) \prec_b (9, 3, 0)$.

Two-player win/lose games: only payoffs $(1, 0)$ or $(0, 1)$.
Such games may have winning strategies.
In such games $s = s_a \cup s_b$ is an NE iff s_a or s_b is winning.

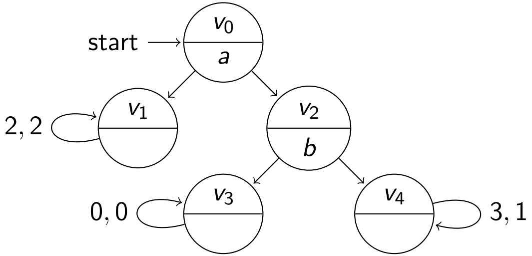
Some special cases

Usually, the preferences are defined in two stages :

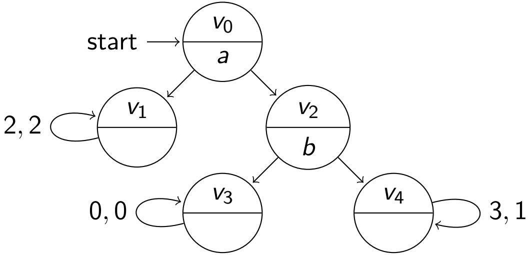
1. by assigning a payoff tuple $A \to \mathbb{R}$ to each run.
   If $A = \{a, b, c\}$ then $(2, 7, 4)$ means a gets 2, b gets 7...
2. and $(0, 2, 1) \prec_b (9, 3, 0)$.

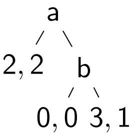
Two-player win/lose games: only payoffs $(1, 0)$ or $(0, 1)$.
Such games may have winning strategies.
In such games $s = s_a \cup s_b$ is an NE iff s_a or s_b is winning.
If a game has a winning strategy, it is said to be determined.

Finite games in extensive form with $\mathbb{R}$-valued payoffs

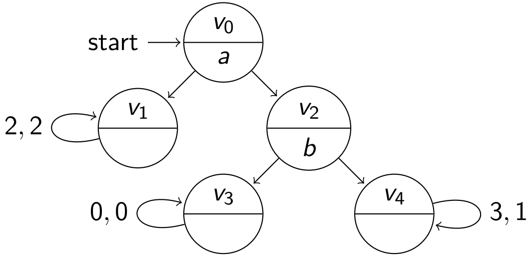


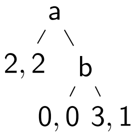
Finite games in extensive form with $\mathbb{R}$-valued payoffs





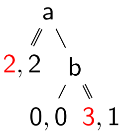
Finite games in extensive form with $\mathbb{R}$-valued payoffs



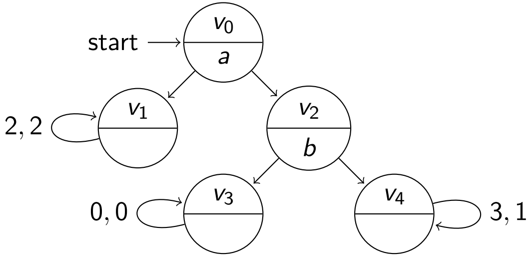


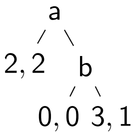
The double lines below represent the strategical choices.

$\neg$NE

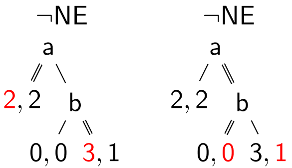


Finite games in extensive form with $\mathbb{R}$-valued payoffs

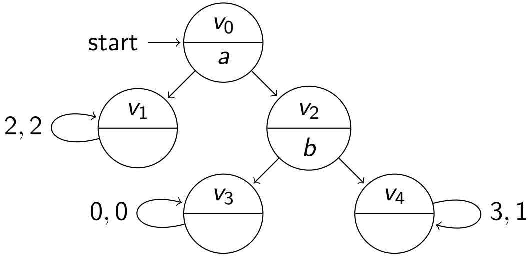


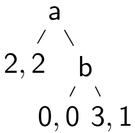


The double lines below represent the strategical choices.

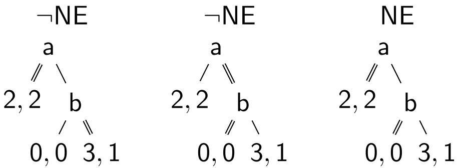


Finite games in extensive form with $\mathbb{R}$-valued payoffs

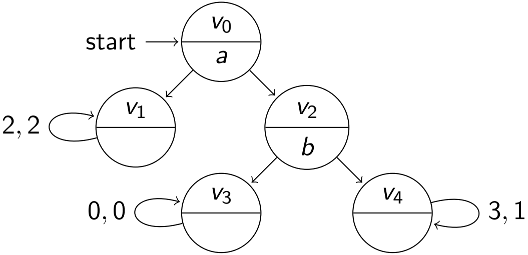


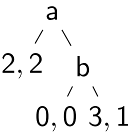


The double lines below represent the strategical choices.

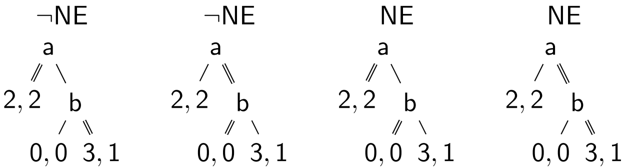


Finite games in extensive form with $\mathbb{R}$-valued payoffs





The double lines below represent the strategical choices.



Towards the transfer theorem

for turn-based games on finite graphs

Theorem (Gurevich and Harrington 1982)

Two-player win/lose Muller games are finite-memory determined

Theorem (Paul and Simon 2009)

Multi-player multi-outcome Muller games have finite-memory NE.

Towards the transfer theorem

for turn-based games on finite graphs

Theorem (Gurevich and Harrington 1982)

Two-player win/lose Muller games are finite-memory determined

Theorem (Paul and Simon 2009)

Multi-player multi-outcome Muller games have finite-memory NE.

Theorem (still a bit vague)

A game g played on a finite graph has a finite-memory NE if

1. *some win/lose derived games are finite-memory determined,*

Towards the transfer theorem

for turn-based games on finite graphs

Theorem (Gurevich and Harrington 1982)

Two-player win/lose Muller games are finite-memory determined

Theorem (Paul and Simon 2009)

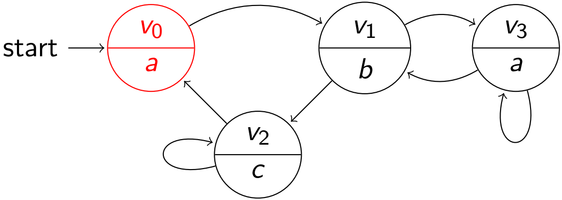
Multi-player multi-outcome Muller games have finite-memory NE.

Theorem (still a bit vague)

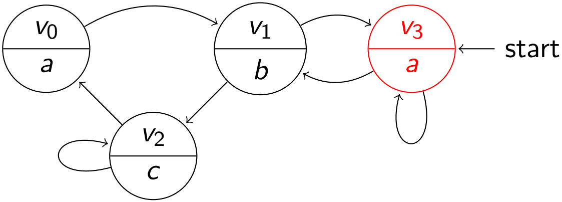
A game g played on a finite graph has a finite-memory NE if

1. *some win/lose derived games are finite-memory determined,*
2. *and the preferences satisfy three conditions.*

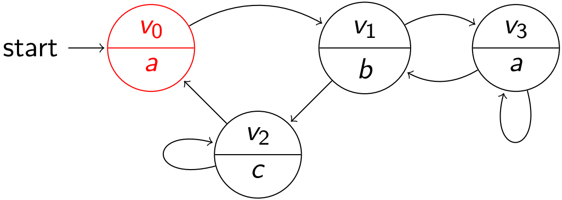
Future games



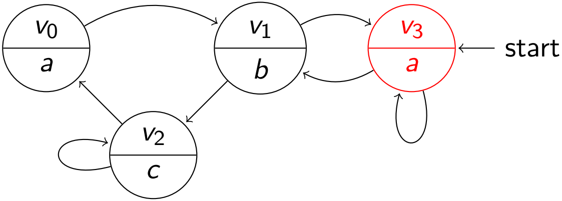
Below: future game after the "imposed history" $v_0 v_1 v_3$:



Future games

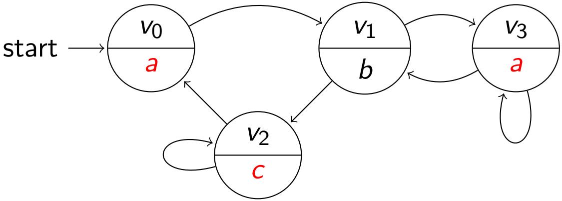


Below: future game after the "imposed history" $v_0 v_1 v_3$:

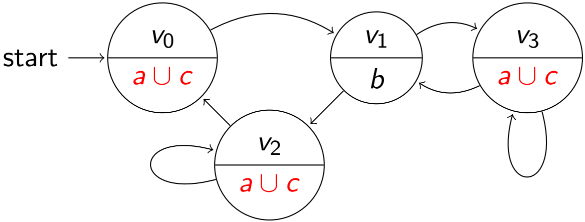


Define $v_3 h \prec_b^{future} v_3 h'$ iff $v_0 v_1 v_3 h \prec_b v_0 v_1 v_3 h'$.

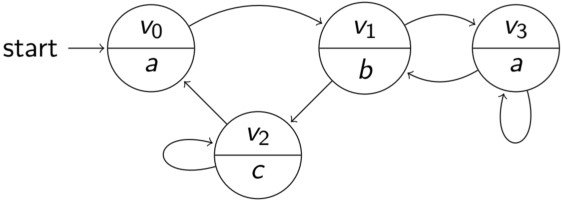
Threshold games



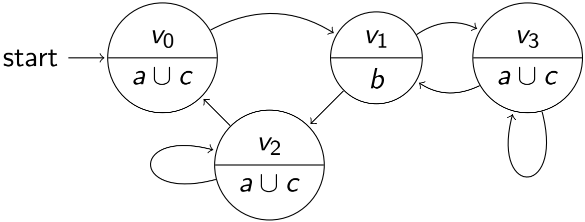
Below: game for b and threshold run $v_0 v_1 v_3^\omega$



Threshold games



Below: game for b and threshold run $v_0 v_1 v_3^\omega$



Player b wins if the run $\rho \succ_b v_0 v_1 v_3^\omega$, else $a \cup c$ wins.

Strict weak order

existing concept

A relation $\prec$ is a strict partial order if it is irreflexive and transitive.

It is a strict weak order if in addition its complement is transitive.

Strict weak order

existing concept

A relation $\prec$ is a strict partial order if it is irreflexive and transitive.

It is a strict weak order if in addition its complement is transitive.

- a strict linear order is a strict weak order,

Strict weak order

existing concept

A relation $\prec$ is a strict partial order if it is irreflexive and transitive.

It is a strict weak order if in addition its complement is transitive.

- a strict linear order is a strict weak order,
- so is the usual order over payoffs, *e.g.* $(0, 2, 1) \prec_b (9, 3, 0)$.

Strict weak order

existing concept

A relation $\prec$ is a strict partial order if it is irreflexive and transitive.

It is a strict weak order if in addition its complement is transitive.

- a strict linear order is a strict weak order,
- so is the usual order over payoffs, *e.g.* $(0, 2, 1) \prec_b (9, 3, 0)$.
- The strict weak order $(\mathbb{R} \times \{0, 1\}, <_{lex})$ cannot be simulated by payoff tuples.

Automatic-piecewise prefix linearity

Usual preferences depend either fully on finite prefixes of the run, or only on its tail. (Apart from discounted payoffs.)

Automatic-piecewise prefix linearity

Usual preferences depend either fully on finite prefixes of the run, or only on its tail. (Apart from discounted payoffs.)

A preference relation $\prec$ is prefix-linear if $h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ for all h, h', ρ, ρ'.

Automatic-piecewise prefix linearity

Usual preferences depend either fully on finite prefixes of the run, or only on its tail. (Apart from discounted payoffs.)

A preference relation $\prec$ is prefix-linear if $h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ for all h, h', ρ, ρ'.

The lexicographic order on $\{0,1\}^\omega$ is prefix-linear.

Automatic-piecewise prefix linearity

Usual preferences depend either fully on finite prefixes of the run, or only on its tail. (Apart from discounted payoffs.)

A preference relation $\prec$ is prefix-linear if
$h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ for all h, h', ρ, ρ'.

The lexicographic order on $\{0,1\}^\omega$ is prefix-linear.

More general: $h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ if $\overline{h'} = \overline{h} \in \overline{\mathcal{H}}$,
where $\overline{\mathcal{H}}$ are the classes of an equivalence relation on $\mathcal{H}$.

Automatic-piecewise prefix linearity

Usual preferences depend either fully on finite prefixes of the run, or only on its tail. (Apart from discounted payoffs.)

A preference relation $\prec$ is prefix-linear if
$h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ for all h, h', ρ, ρ'.

The lexicographic order on $\{0,1\}^\omega$ is prefix-linear.

More general: $h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ if $\overline{h'} = \overline{h} \in \overline{\mathcal{H}}$,
where $\overline{\mathcal{H}}$ are the classes of an equivalence relation on $\mathcal{H}$.
If the classes are decidable by a finite automaton, $\prec$ is automatic-piecewise prefix-linear.

Automatic-piecewise prefix linearity

Usual preferences depend either fully on finite prefixes of the run, or only on its tail. (Apart from discounted payoffs.)

A preference relation $\prec$ is prefix-linear if
$h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ for all h, h', ρ, ρ'.

The lexicographic order on $\{0,1\}^\omega$ is prefix-linear.

More general: $h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ if $\overline{h'} = \overline{h} \in \overline{\mathcal{H}}$,
where $\overline{\mathcal{H}}$ are the classes of an equivalence relation on $\mathcal{H}$.
If the classes are decidable by a finite automaton, $\prec$ is automatic-piecewise prefix-linear.

On $\{0,1\}^\omega$ let $0\rho \prec 0\rho' \Leftrightarrow \rho <_{lex} \rho'$ and $1\rho \prec 1\rho' \Leftrightarrow \rho >_{lex} \rho'$.
Then $\prec$ is automatic-piecewise prefix-linear (with two classes),

Automatic-piecewise prefix linearity

Usual preferences depend either fully on finite prefixes of the run, or only on its tail. (Apart from discounted payoffs.)

A preference relation $\prec$ is prefix-linear if $h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ for all h, h', ρ, ρ'.

The lexicographic order on $\{0,1\}^\omega$ is prefix-linear.

More general: $h\rho \prec h\rho' \Leftrightarrow h'\rho \prec h'\rho'$ if $\overline{h'} = \overline{h} \in \overline{\mathcal{H}}$, where $\overline{\mathcal{H}}$ are the classes of an equivalence relation on $\mathcal{H}$. If the classes are decidable by a finite automaton, $\prec$ is automatic-piecewise prefix-linear.

On $\{0,1\}^\omega$ let $0\rho \prec 0\rho' \Leftrightarrow \rho <_{lex} \rho'$ and $1\rho \prec 1\rho' \Leftrightarrow \rho >_{lex} \rho'$. Then $\prec$ is automatic-piecewise prefix-linear (with two classes), but $\prec$ is not prefix-linear: $010 \prec 011$ but $10 \succ 11$.

The Mont condition

A relation $\prec \subseteq V^\omega \times V^\omega$ is Mont if $\forall h_0, h_1, h_2, \dots \in V^*$ we have: $h_0 \dots h_n \rho \prec h_0 \dots h_n h_{n+1} \rho$ for all $n \in \mathbb{N}$ implies $h_0 \rho \prec h_0 h_1 h_2 \dots$

The Mont condition

A relation $\prec \subseteq V^\omega \times V^\omega$ is Mont if $\forall h_0, h_1, h_2, \dots \in V^*$ we have:
$h_0 \dots h_n \rho \prec h_0 \dots h_n h_{n+1} \rho$ for all $n \in \mathbb{N}$ implies $h_0 \rho \prec h_0 h_1 h_2 \dots$

Prefix independent, irreflexive relations are Mont:
$h_0 \dots h_n \rho \prec h_0 \dots h_n h_{n+1} \rho$ implies $\rho \prec \rho$.

Our result

Theorem

Let a game be played by players in A on a graph over finite V s.t.

1. *All one-vs-all threshold games of all future games are determined via strategies using m bits of memory.*

Our result

Theorem

Let a game be played by players in A on a graph over finite V s.t.

1. *All one-vs-all threshold games of all future games are determined via strategies using m bits of memory.*
2. *The $\prec_a$ are automatic-piecewise (with k classes) prefix-linear Mont strict weak orders.*

Our result

Theorem

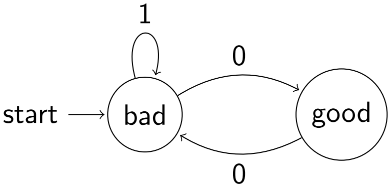
Let a game be played by players in A on a graph over finite V s.t.

1. *All one-vs-all threshold games of all future games are determined via strategies using m bits of memory.*
2. *The $\prec_a$ are automatic-piecewise (with k classes) prefix-linear Mont strict weak orders.*

Then the game has an NE in finite-memory strategies requiring $|A|(m + 2\log\max(k, |V|)) + 1$ bits of memory.

Counterexamples

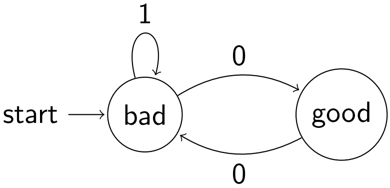
Why "All one-vs-all threshold games of all future games are determined via strategies using m bits of memory"?



If finitely many "good" then payoff 0, else lim sup average 0 and 1.

Counterexamples

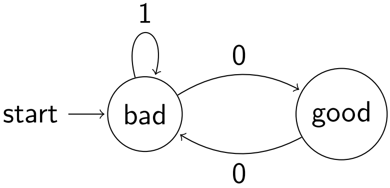
Why "All one-vs-all threshold games of all future games are determined via strategies using *m* bits of memory"?



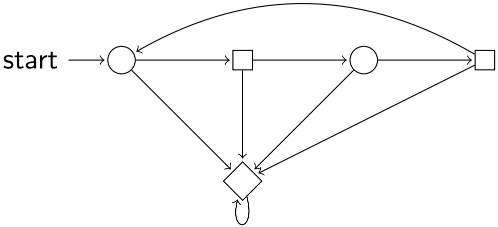
If finitely many "good" then payoff 0, else $\limsup$ average 0 and 1.

The unique player wins all the strict thresholds < 1 and can do so with finite memory, but the game has no finite-memory NE.

Counterexamples

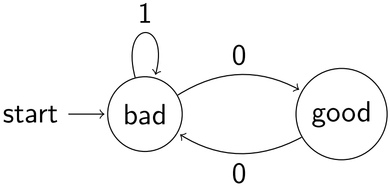


Why Mont preferences?

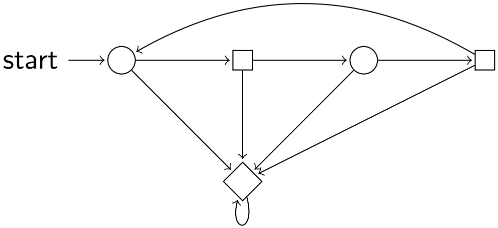


Payoff for Player "circle": if the diamond is never visited then -1, else number of visited squares.

Counterexamples

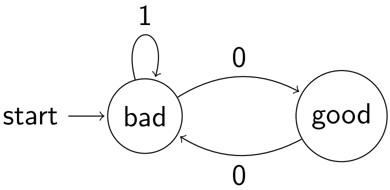


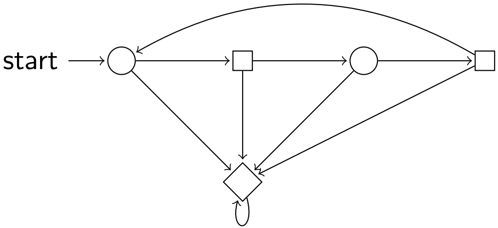
Why Mont preferences?



Payoff for Player "circle": if the diamond is never visited then -1, else number of visited squares. The threshold games are all memoryless determined! but there is not even an NE.

Counterexamples





Gurvich and Oudalov (2014) constructed a four-player 13-state one-cycle game with no positional NE. So, no transfer theorem with memoryless determinacy.