Advanced Topics in Infinite Games

Seminar Summer Term 2014

Kickoff Meeting
April 16th, 2014

Outline

1. **A Short Introduction to Infinite Games**
2. **Organization**
3. **Paper Bidding**

Infinite Games: Motivation

- Model-checking for fixed-point logics.
- Synthesis of correct-by-construction controllers for reactive systems (non-terminating, interacting with antagonistic environment).
- Automata emptiness often expressible in terms of games.
- Semantics of alternating automata in terms of games.

Infinite Games: Motivation

- Model-checking for fixed-point logics.
- Synthesis of correct-by-construction controllers for reactive systems (non-terminating, interacting with antagonistic environment).
- Automata emptiness often expressible in terms of games.
- Semantics of alternating automata in terms of games.

Earliest appearance: **Church's problem (1957)**

Given requirement φ on input-output behavior of boolean circuits, compute a circuit that satisfies φ (or prove that none exists).

Infinite Games: Motivation

- Model-checking for fixed-point logics.
- Synthesis of correct-by-construction controllers for reactive systems (non-terminating, interacting with antagonistic environment).
- Automata emptiness often expressible in terms of games.
- Semantics of alternating automata in terms of games.

Earliest appearance: **Church's problem (1957)**

Given requirement φ on input-output behavior of boolean circuits, compute a circuit that satisfies φ (or prove that none exists).

Game theoretic formulation: two-player game

- Player 0 generates infinite stream of input bits,
- Player 1 has to answer each input bit by output bit such that
- combination of streams satisfies φ.

Church's Problem: Example

φ is conjunction of following properties:

1. Whenever the input bit is 1, then the output bit is 1, too.
2. If there are infinitely many 0's in the input stream, then there are infinitely many 0's in the output stream.
3. At least one out of every three consecutive output bits is a 1.

Church's Problem: Example

φ is conjunction of following properties:

1. Whenever the input bit is 1, then the output bit is 1, too.
2. If there are infinitely many 0's in the input stream, then there are infinitely many 0's in the output stream.
3. At least one out of every three consecutive output bits is a 1.

Winning strategy for the output player:

- Answer every 1 by a 1.
- Answer every 0 by a 0,

Church's Problem: Example

φ is conjunction of following properties:

1. Whenever the input bit is 1, then the output bit is 1, too.
2. If there are infinitely many 0's in the input stream, then there are infinitely many 0's in the output stream.
3. At least one out of every three consecutive output bits is a 1.

Winning strategy for the output player:

- Answer every 1 by a 1.
- Answer every 0 by a 0, unless it would the third 0 in a row. Then, answer by a 1.

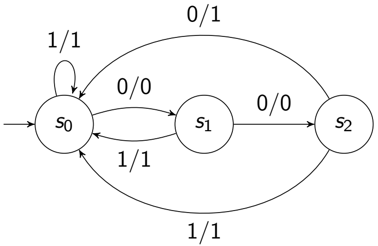
Church's Problem: Example

φ is conjunction of following properties:

1. Whenever the input bit is 1, then the output bit is 1, too.
2. If there are infinitely many 0's in the input stream, then there are infinitely many 0's in the output stream.
3. At least one out of every three consecutive output bits is a 1.

Winning strategy for the output player:

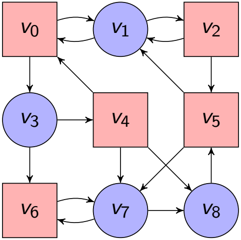
- Answer every 1 by a 1.
- Answer every 0 by a 0, unless it would the third 0 in a row. Then, answer by a 1.



Infinite Games: Arenas

One more level of abstraction: two-player games on finite graphs.

- An *arena* $\mathcal{A} = (V, V_0, V_1, E)$ consists of
  - a finite set V of vertices,
  - a set $V_0 \subseteq V$ of vertices owned by Player 0,
  - the set $V_1 = V \setminus V_0$ of vertices owned by Player 1,
  - a directed edge-relation $E \subseteq V \times V$.



Infinite Games: Plays

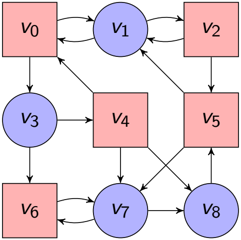
Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).

Infinite Games: Plays

Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).

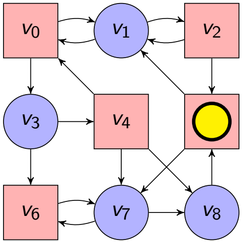


Example play:

Infinite Games: Plays

Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).

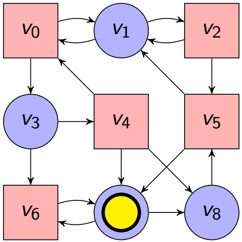


Example play: v_5

Infinite Games: Plays

Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).

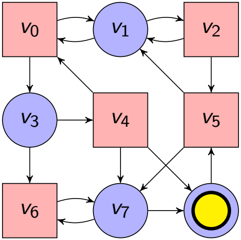


Example play: v_5 v_7

Infinite Games: Plays

Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).

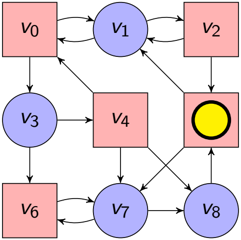


Example play: $v_5 \; v_7 \; v_8$

Infinite Games: Plays

Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).

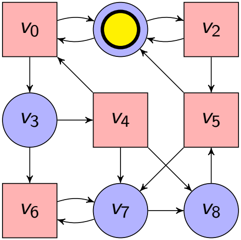


Example play: $v_5\ v_7\ v_8\ v_5$

Infinite Games: Plays

Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).

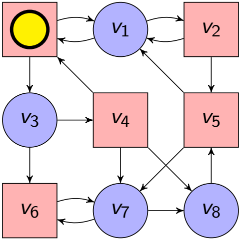


Example play: $v_5 \; v_7 \; v_8 \; v_5 \; v_1$

Infinite Games: Plays

Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).

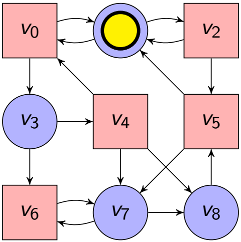


Example play: $v_5\ v_7\ v_8\ v_5\ v_1\ v_0$

Infinite Games: Plays

Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).

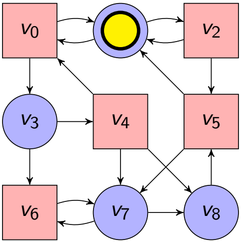


Example play: $v_5\ v_7\ v_8\ v_5\ v_1\ v_0\ v_1$

Infinite Games: Plays

Rules:

- A token is placed at an initial vertex.
- If the token is at vertex of Player i, he moves it to a successor.
- Thereby, the players construct a play (an infinite path in $\mathcal{A}$).



Example play: $v_5\ v_7\ v_8\ v_5\ v_1\ v_0\ v_1\ \cdots$

Infinite Games: Strategies

- A strategy for Player i in $\mathcal{A}$ is a mapping $\sigma\colon V^*V_i \to V$ satisfying $(v_n, \sigma(v_0 \cdots v_n)) \in E$ (only legal moves).

Infinite Games: Strategies

- A strategy for Player i in $\mathcal{A}$ is a mapping $\sigma\colon V^*V_i \to V$ satisfying $(v_n, \sigma(v_0 \cdots v_n)) \in E$ (only legal moves).
- A play $v_0 v_1 v_2 \cdots$ is consistent with σ, if $v_{n+1} = \sigma(v_0 \cdots v_n)$ for every n with $v_n \in V_i$.

Infinite Games: Strategies

- A strategy for Player i in $\mathcal{A}$ is a mapping $\sigma\colon V^*V_i \to V$ satisfying $(v_n, \sigma(v_0 \cdots v_n)) \in E$ (only legal moves).
- A play $v_0v_1v_2\cdots$ is consistent with σ, if $v_{n+1} = \sigma(v_0 \cdots v_n)$ for every n with $v_n \in V_i$.
- Note: if we fix an initial vertex and strategies σ and τ for Player 0 and Player 1, then there is a unique play that starts in v and is consistent with σ and τ.

Infinite Games: Strategies

- A strategy for Player i in $\mathcal{A}$ is a mapping $\sigma \colon V^* V_i \to V$ satisfying $(v_n, \sigma(v_0 \cdots v_n)) \in E$ (only legal moves).
- A play $v_0 v_1 v_2 \cdots$ is consistent with σ, if $v_{n+1} = \sigma(v_0 \cdots v_n)$ for every n with $v_n \in V_i$.
- Note: if we fix an initial vertex and strategies σ and τ for Player 0 and Player 1, then there is a unique play that starts in v and is consistent with σ and τ.

Special types of strategies:

- Positional strategies: $\sigma(v_0 \cdots v_n) = \sigma(v_n)$ for all $v_0 \cdots v_n$: move only depends on position the token is at at the moment.

Infinite Games: Strategies

- A strategy for Player i in $\mathcal{A}$ is a mapping $\sigma\colon V^*V_i \rightarrow V$ satisfying $(v_n, \sigma(v_0 \cdots v_n)) \in E$ (only legal moves).
- A play $v_0 v_1 v_2 \cdots$ is consistent with σ, if $v_{n+1} = \sigma(v_0 \cdots v_n)$ for every n with $v_n \in V_i$.
- Note: if we fix an initial vertex and strategies σ and τ for Player 0 and Player 1, then there is a unique play that starts in v and is consistent with σ and τ.

Special types of strategies:

- Positional strategies: $\sigma(v_0 \cdots v_n) = \sigma(v_n)$ for all $v_0 \cdots v_n$: move only depends on position the token is at at the moment.
- Finite-state strategies: implemented by DFA with output reading play prefix $v_0 \cdots v_n$ and outputting $\sigma(v_0 \cdots v_n)$.

Infinite Games: Winning

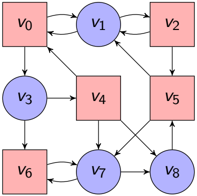
- A game $\mathcal{G} = (\mathcal{A}, \mathrm{Win})$ consists of an arena $\mathcal{A}$ and a set $\mathrm{Win} \subseteq V^\omega$ of winning plays for Player 0.
- Set of winning plays for Player 1: $V^\omega \setminus \mathrm{Win}$.

Infinite Games: Winning

- A game $\mathcal{G} = (\mathcal{A}, \mathrm{Win})$ consists of an arena $\mathcal{A}$ and a set $\mathrm{Win} \subseteq V^\omega$ of winning plays for Player 0.
- Set of winning plays for Player 1: $V^\omega \setminus \mathrm{Win}$.
- Strategy σ for Player i is winning strategy from v, if every play that starts in v and is consistent with σ is winning for him.

Infinite Games: Winning

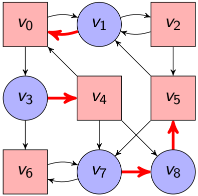
- A game $\mathcal{G} = (\mathcal{A}, \mathrm{Win})$ consists of an arena $\mathcal{A}$ and a set $\mathrm{Win} \subseteq V^\omega$ of winning plays for Player 0.
- Set of winning plays for Player 1: $V^\omega \setminus \mathrm{Win}$.
- Strategy σ for Player i is winning strategy from v, if every play that starts in v and is consistent with σ is winning for him.



$$\mathrm{Win} = \{\, \rho \in V^\omega \mid \rho \text{ does not visit all vertices} \,\}$$

Infinite Games: Winning

- A game $\mathcal{G} = (\mathcal{A}, \mathrm{Win})$ consists of an arena $\mathcal{A}$ and a set $\mathrm{Win} \subseteq V^\omega$ of winning plays for Player 0.
- Set of winning plays for Player 1: $V^\omega \setminus \mathrm{Win}$.
- Strategy σ for Player i is winning strategy from v, if every play that starts in v and is consistent with σ is winning for him.

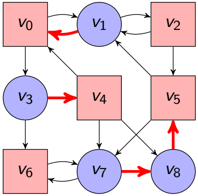


$$\mathrm{Win} = \{ \rho \in V^\omega \mid \rho \text{ does not visit all vertices} \}$$

Player 0 wins from every vertex

Infinite Games: Winning

- A game $\mathcal{G} = (\mathcal{A}, \mathrm{Win})$ consists of an arena $\mathcal{A}$ and a set $\mathrm{Win} \subseteq V^\omega$ of winning plays for Player 0.
- Set of winning plays for Player 1: $V^\omega \setminus \mathrm{Win}$.
- Strategy σ for Player i is winning strategy from v, if every play that starts in v and is consistent with σ is winning for him.



$$\mathrm{Win} = \{\, \rho \in V^\omega \mid \rho \text{ does not visit all vertices} \,\}$$

Player 0 wins from every vertex with positional strategies.

Infinite Games: Winning

- A game $\mathcal{G} = (\mathcal{A}, \mathrm{Win})$ consists of an arena $\mathcal{A}$ and a set $\mathrm{Win} \subseteq V^\omega$ of winning plays for Player 0.
- Set of winning plays for Player 1: $V^\omega \setminus \mathrm{Win}$.
- Strategy σ for Player i is winning strategy from v, if every play that starts in v and is consistent with σ is winning for him.

- Winning region $W_i(\mathcal{G})$: set of vertices from which Player i has a winning strategy.
- Always: $W_0(\mathcal{G}) \cap W_1(\mathcal{G}) = \emptyset$.
- $\mathcal{G}$ determined, if $W_0(\mathcal{G}) \cup W_1(\mathcal{G}) = V$.
- Solving a game: determine the winning regions and winning strategies.

Three Types of Winning Conditions

Win is an (possibly) infinite set of infinite words, and therefore unsuitable as input to an algorithm $\Rightarrow$ need finite representation.

Three Types of Winning Conditions

Win is an (possibly) infinite set of infinite words, and therefore unsuitable as input to an algorithm $\Rightarrow$ need finite representation.

- Reachability games: for $R \subseteq V$ define

$$\textsc{reach}(R) = \{\, \rho \in V^\omega \mid \rho \text{ visits } R \text{ at least once} \,\}$$

Three Types of Winning Conditions

Win is an (possibly) infinite set of infinite words, and therefore unsuitable as input to an algorithm $\Rightarrow$ need finite representation.

- Reachability games: for $R \subseteq V$ define

$$\textsc{reach}(R) = \{\, \rho \in V^\omega \mid \rho \text{ visits } R \text{ at least once} \,\}$$

- Parity games: for $\Omega\colon V \to \mathbb{N}$ define

$$\textsc{parity}(\Omega) = \{\, \rho \in V^\omega \mid \text{minimal priority seen infinitely often during } \rho \text{ is even} \,\}$$

Three Types of Winning Conditions

Win is an (possibly) infinite set of infinite words, and therefore unsuitable as input to an algorithm $\Rightarrow$ need finite representation.

- Reachability games: for $R \subseteq V$ define

$$\textsc{reach}(R) = \{\, \rho \in V^\omega \mid \rho \text{ visits } R \text{ at least once} \,\}$$

- Parity games: for $\Omega \colon V \to \mathbb{N}$ define

$$\textsc{parity}(\Omega) = \{\, \rho \in V^\omega \mid \text{minimal priority seen infinitely often during } \rho \text{ is even} \,\}$$

- Muller games: for $\mathcal{F} \subseteq 2^V$ define

$$\textsc{muller}(\mathcal{F}) = \{\, \rho \in V^\omega \mid \text{set of vertices seen infinitely often during } \rho \text{ is in } \mathcal{F} \,\}$$

Three Types of Winning Conditions

Win is an (possibly) infinite set of infinite words, and therefore unsuitable as input to an algorithm $\Rightarrow$ need finite representation.

- Reachability games: for $R \subseteq V$ define

$$\textsc{Reach}(R) = \{\, \rho \in V^\omega \mid \rho \text{ visits } R \text{ at least once} \,\}$$

- Parity games: for $\Omega\colon V \to \mathbb{N}$ define

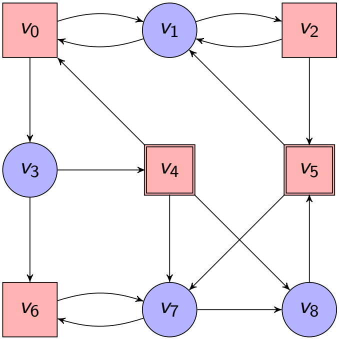
$$\textsc{Parity}(\Omega) = \{\, \rho \in V^\omega \mid \text{minimal priority seen infinitely often during } \rho \text{ is even} \,\}$$

- Muller games: for $\mathcal{F} \subseteq 2^V$ define

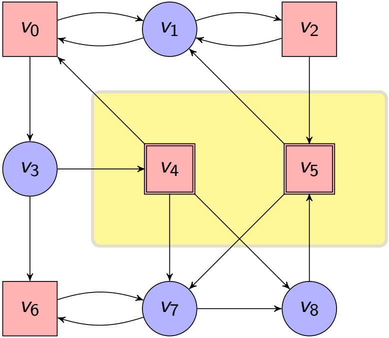
$$\textsc{Muller}(\mathcal{F}) = \{\, \rho \in V^\omega \mid \text{set of vertices seen infinitely often during } \rho \text{ is in } \mathcal{F} \,\}$$

There are many other winning conditions.

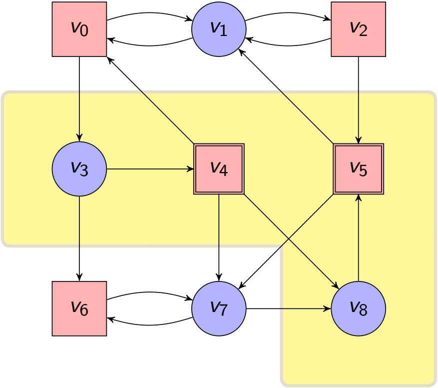
Reachability Games



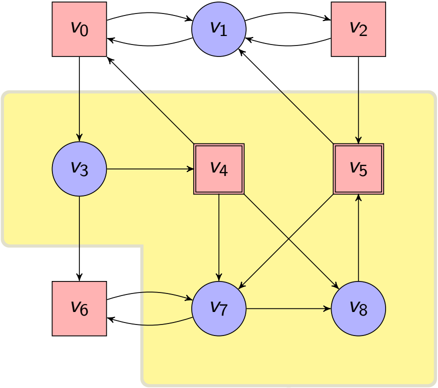
Reachability Games



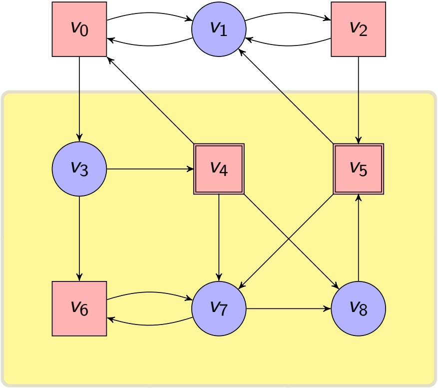
Reachability Games



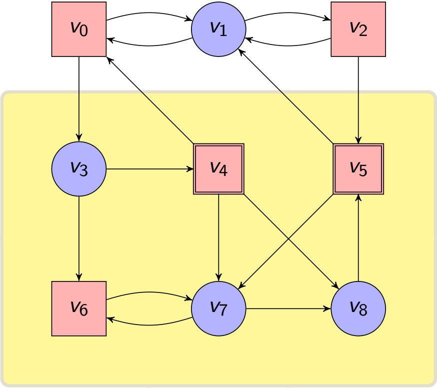
Reachability Games



Reachability Games

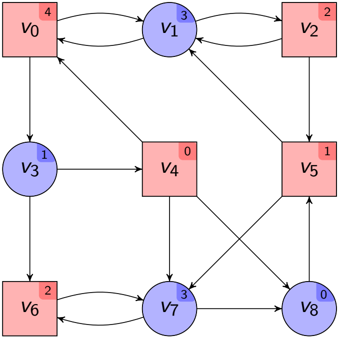


Reachability Games

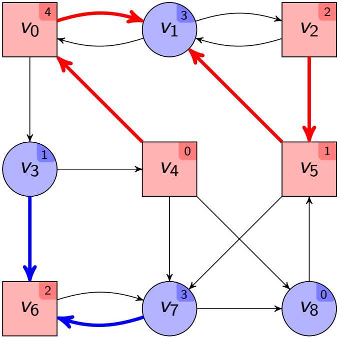


- Both players have positional winning strategies.
- Reachability games can be solved in linear time.

Parity Games

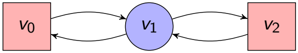


Parity Games



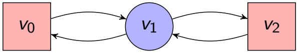
- Both players have positional winning strategies.
- Solving: in $\mathrm{NP} \cap \mathrm{Co\text{-}NP}$, not known to be in $\mathrm{PT{\scriptsize IME}}$.

Muller Games



$$\mathcal{F} = \{\{v_0, v_1, v_2\}\}$$

Muller Games



$$\mathcal{F} = \{\{v_0, v_1, v_2\}\}$$

- Both players need finite-state strategies.
- Complexity depends on encoding of $\mathcal{F}$: PTime, NP $\cap$ Co-NP, Pspace.

Outline

1. A Short Introduction to Infinite Games
2. **Organization**
3. Paper Bidding

Your Grade

- Seminar paper (8-10 pages)
- Presentation (40 minutes + discussion)
- 3-Minute-Madness
- Participation in discussions
- Meet all deadlines

Schedule

- Kickoff Meeting — Right now
- Meeting with your tutor — asap
- **3-Minute-Madness — Friday, May 9th**
- 1st Draft of seminar paper — before June, 29th
- **Deadline for seminar paper — July, 13th**
- 1st Draft of slides — before July, 27th
- Practice talk (optional, but strongly encouraged)
- **Presentations — August, 4th & 5th**

3-Minute-Madness

Present topic of your paper in three minutes!

3-Minute-Madness

Present topic of your paper in three minutes!

Focus on concepts and intuition:

- How do games in your paper extend the basic setup described here (interaction between players, winning condition, type of strategies,..)?
- Maybe give an example explaining the extensions.
- No theorems, proofs, etc.
- Preferably: even no formal definitions.

3-Minute-Madness

Present topic of your paper in three minutes!

Focus on concepts and intuition:

- How do games in your paper extend the basic setup described here (interaction between players, winning condition, type of strategies,..)?
- Maybe give an example explaining the extensions.
- No theorems, proofs, etc.
- Preferably: even no formal definitions.

Rules:

- Three minute presentation (strict)
- As many slides as you need (the less the better)
- Format: pdf
- Slides have to be submitted 24 hours before meeting.

Outline

1. A Short Introduction to Infinite Games

2. Organization

3. **Paper Bidding**

Paper Bidding - Rules

- Short introduction to each paper on the next slides.
- Sheets where you can rank your favorite papers.
- Papers are assigned trying to accommodate your preferences as much as possible.
- Notification by email as soon as possible.

Paper Bidding - Rules

- Short introduction to each paper on the next slides.
- Sheets where you can rank your favorite papers.
- Papers are assigned trying to accommodate your preferences as much as possible.
- Notification by email as soon as possible.

Topics:

1. Quantitative winning conditions
2. Delay
3. Imperfect information/Concurrent games
4. Further topics

Quantitative winning conditions

- **Mean-payoff Games: Positional Determinacy**
  Andrzej Ehrenfeucht and Jan Mycielski. **Positional Strategies for Mean Payoff Games.** International Journal of Game Theory, 8(2):109-113 (1979)

Quantitative winning conditions

- **Mean-payoff Games: Positional Determinacy**
  Andrzej Ehrenfeucht and Jan Mycielski. **Positional Strategies for Mean Payoff Games.** International Journal of Game Theory, 8(2):109-113 (1979)
- **Mean-payoff Games: Algorithms**
  Uri Zwick, Mike Paterson. **The Complexity of Mean Payoff Games on Graphs.** Theor. Comput. Sci. 158(1&2): 343-359 (1996)

Quantitative winning conditions

- **Mean-payoff Games: Positional Determinacy**
  Andrzej Ehrenfeucht and Jan Mycielski. **Positional Strategies for Mean Payoff Games.** International Journal of Game Theory, 8(2):109-113 (1979)
- **Mean-payoff Games: Algorithms**
  Uri Zwick, Mike Paterson. **The Complexity of Mean Payoff Games on Graphs.** Theor. Comput. Sci. 158(1&2): 343-359 (1996)
- **Finitary Games**
  Krishnendu Chatterjee, Thomas A. Henzinger, Florian Horn. **Finitary Winning in omega-regular Games.** ACM Trans. Comput. Log. 11(1) (2009)

Quantitative winning conditions

- **Mean-payoff Games: Positional Determinacy**
  Andrzej Ehrenfeucht and Jan Mycielski. **Positional Strategies for Mean Payoff Games.** International Journal of Game Theory, 8(2):109-113 (1979)
- **Mean-payoff Games: Algorithms**
  Uri Zwick, Mike Paterson. **The Complexity of Mean Payoff Games on Graphs.** Theor. Comput. Sci. 158(1&2): 343-359 (1996)
- **Finitary Games**
  Krishnendu Chatterjee, Thomas A. Henzinger, Florian Horn. **Finitary Winning in omega-regular Games.** ACM Trans. Comput. Log. 11(1) (2009)
- **Energy Parity Games**
  Krishnendu Chatterjee, Laurent Doyen. **Energy Parity Games.** Theor. Comput. Sci. 458: 49-60 (2012)

Delay

- **Delay in Regular Games**
  Michael Holtmann, Łukasz Kaiser, Wolfgang Thomas.
  Degrees of Lookahead in Regular Infinite Games. Logical Methods in Computer Science 8(3) (2012)

Delay

- **Delay in Regular Games**
  Michael Holtmann, Łukasz Kaiser, Wolfgang Thomas.
  Degrees of Lookahead in Regular Infinite Games. Logical Methods in Computer Science 8(3) (2012)
- **Delay in Context-free Games**
  Wladimir Fridman, Christof Löding, Martin Zimmermann.
  Degrees of Lookahead in Context-free Infinite Games.
  CSL 2011: 264-276

Imperfect Information/Concurrent Games

- **Games with Imperfect Information**
  Laurent Doyen and Jean-François Raskin. **Games with Imperfect Information: Theory and Algorithms.** Lectures in Game Theory for Computer Scientists. Cambridge University Press, pp. 185-212 (2011)

Imperfect Information/Concurrent Games

- **Games with Imperfect Information**
  Laurent Doyen and Jean-François Raskin. **Games with Imperfect Information: Theory and Algorithms.** Lectures in Game Theory for Computer Scientists. Cambridge University Press, pp. 185-212 (2011)
- **Concurrent Reachability Games**
  Luca de Alfaro, Thomas A. Henzinger, Orna Kupferman. **Concurrent Reachability Games.** Theor. Comput. Sci. 386(3): 188-217 (2007)

Imperfect Information/Concurrent Games

- **Games with Imperfect Information**
  Laurent Doyen and Jean-François Raskin. **Games with Imperfect Information: Theory and Algorithms.** Lectures in Game Theory for Computer Scientists. Cambridge University Press, pp. 185-212 (2011)
- **Concurrent Reachability Games**
  Luca de Alfaro, Thomas A. Henzinger, Orna Kupferman. **Concurrent Reachability Games.** Theor. Comput. Sci. 386(3): 188-217 (2007)
- **Distributed Games**
  Swarup Mohalik, Igor Walukiewicz. **Distributed Games.** Proceedings of FSTTCS'03, LNCS 2914: 338-351 (2003)

Further Topics

- **Zielonka Trees**
  Stefan Dziembowski, Marcin Jurdziński, Igor Walukiewicz.
  How Much Memory is Needed to Win Infinite Games?
  LICS 1997: 99-110

Further Topics

- **Zielonka Trees**
  Stefan Dziembowski, Marcin Jurdziński, Igor Walukiewicz.
  How Much Memory is Needed to Win Infinite Games?
  LICS 1997: 99-110
- **Games of Ordinal Length**
  Julien Cristau, Florian Horn. **Graph Games on Ordinals.**
  FSTTCS 2008: 143-154

Further Topics

- **Zielonka Trees**
  Stefan Dziembowski, Marcin Jurdziński, Igor Walukiewicz. **How Much Memory is Needed to Win Infinite Games?** LICS 1997: 99-110
- **Games of Ordinal Length**
  Julien Cristau, Florian Horn. **Graph Games on Ordinals.** FSTTCS 2008: 143-154
- **Permissive Strategies**
  Julien Bernet, David Janin, Igor Walukiewicz. **Permissive Strategies: from Parity Games to Safety Games.** ITA 36(3): 261-275 (2002)

Your Choice

1. Mean-payoff Games: Positional Determinacy
2. Mean-payoff Games: Algorithms
3. Finitary Games
4. Energy Parity Games
5. Delay in Regular Games
6. Delay in Context-free Games
7. Games with Imperfect Information
8. Concurrent Reachability Games
9. Distributed Games
10. Zielonka Trees
11. Games of Ordinal Length
12. Permissive Strategies