Synthesis of Winning Strategies for Interaction under Partial Information

Oberseminar Informatik

Bernd Puchala

RWTH Aachen University

June 10th, 2013

Interaction: General Properties

Interaction: General Properties

- grand coalition of protagonists (players $1, \ldots, n$)
- single antagonist (player 0)

Interaction: General Properties

- grand coalition of protagonists (players $1, \ldots, n$)
- single antagonist (player 0)
- infinitely many rounds (nonterminating)
- synchronous and concurrent:
  in any of these rounds, all players simultaneously perform some action

Interaction: General Properties

- grand coalition of protagonists (players $1, \ldots, n$)
- single antagonist (player 0)
- infinitely many rounds (nonterminating)
- synchronous and concurrent:
  in any of these rounds, all players simultaneously perform some action
- finitely many actions

Interaction: General Properties

- grand coalition of protagonists (players $1, \ldots, n$)
- single antagonist (player 0)
- infinitely many rounds (nonterminating)
- synchronous and concurrent:
  in any of these rounds, all players simultaneously perform some action
- finitely many actions
- Partial Information about history of past events

Interaction: Formal Model

Interaction: Formal Model

Game graph with partial information and $n+1$ players:

$$\mathcal{G} = (V, \delta, (\sim_i)_{i=1,\ldots,n})$$

Interaction: Formal Model

Game graph with partial information and $n+1$ players:

$$\mathcal{G} = (V, \delta, (\sim_i)_{i=1,\dots,n})$$

- V finite set of positions

Interaction: Formal Model

Game graph with partial information and $n+1$ players:

$$\mathcal{G} = (V, \delta, (\sim_i)_{i=1,\dots,n})$$

- V finite set of positions
- move function $\delta : \mathsf{dom}(\delta) \subseteq V \times A \to V$
- finite set A of actions $a = (a_0, \dots, a_n)$

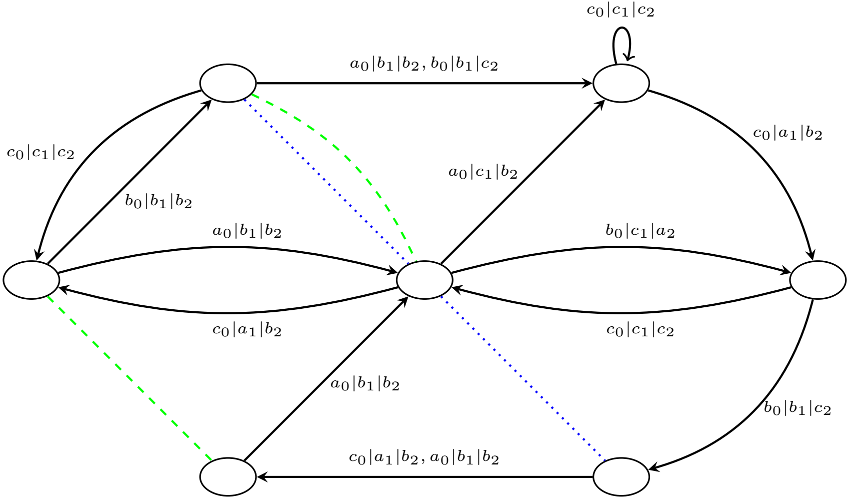
Interaction: Formal Model

Game graph with partial information and $n+1$ players:

$$\mathcal{G} = (V, \delta, (\sim_i)_{i=1,\ldots,n})$$

- V finite set of positions
- move function $\delta : \mathrm{dom}(\delta) \subseteq V \times A \to V$
- finite set A of actions $a = (a_0, \ldots, a_n)$
- equivalence relations $\sim_i \subseteq V \times V$

Interaction: Formal Model



Interaction: Strategies

Interaction: Strategies

Strategy for player i:

$$\sigma_i \colon V^* \to A_i$$

Interaction: Strategies

Strategy for player i:

$$\sigma_i \colon V^* \to A_i$$

such that for all $\pi = v_0 v_1 \ldots v_l$ and $\rho = w_0 w_1 \ldots w_l$:

$$v_j \sim_i w_j \text{ for all } j \quad \Longrightarrow \quad \sigma_i(\rho) = \sigma_i(\pi)$$

Interaction: Strategies

Strategy for player i:

$$\sigma_i \colon V^* \to A_i$$

such that for all $\pi = v_0 v_1 \dots v_l$ and $\rho = w_0 w_1 \dots w_l$:

$$v_j \sim_i w_j \text{ for all } j \implies \sigma_i(\rho) = \sigma_i(\pi)$$

Joint strategy for the grand coalition:

$$\sigma = (\sigma_1, \dots, \sigma_n)$$

Interaction: Objectives

Interaction: Objectives

Winning condition for the grand coalition:

$$W \subseteq V^{\omega}$$

Interaction: Objectives

Winning condition for the grand coalition:

$$W \subseteq V^{\omega}$$

Here:

- W regular
  (recognizable by deterministic parity automaton)
- W context-free
  (recognizable by parity automaton with stack memory)

Interaction: Objectives

Winning condition for the grand coalition:

$$W \subseteq V^\omega$$

Here:

- W regular
  (recognizable by deterministic parity automaton)
- W context-free
  (recognizable by parity automaton with stack memory)

joint winning strategy σ:
all plays that are played according to σ are in W

Interaction: Objectives

Winning condition for the grand coalition:

$$W \subseteq V^\omega$$

Here:

- W regular
  (recognizable by deterministic parity automaton)
- W context-free
  (recognizable by parity automaton with stack memory)

joint winning strategy σ:
all plays that are played according to σ are in W

(strategy that guarantees winning for the grand coalition against **all** possible behaviors of player 0)

Strategy Synthesis: Problem

Given: Game graph with partial information, winning condition.
Question: Does the grand coalition have a joint winning strategy?

Strategy Synthesis: Problem

Given: Game graph with partial information, winning condition.
Question: Does the grand coalition have a joint winning strategy?

! We can assume that player 0 has full information

Strategy Synthesis: Problem

Given: Game graph with partial information, winning condition.
Question: Does the grand coalition have a joint winning strategy?

! We can assume that player 0 has full information

If there is such a strategy, construct one!

Strategy Synthesis: Problem

Given: Game graph with partial information, winning condition.
Question: Does the grand coalition have a joint winning strategy?

- ! We can assume that player 0 has full information

If there is such a strategy, construct one! (Decision-Method++)

Strategy Synthesis: Problem

Given: Game graph with partial information, winning condition.
Question: Does the grand coalition have a joint winning strategy?

! We can assume that player 0 has full information

If there is such a strategy, construct one! (Decision-Method++)

Implementation by

- Finite state machine
- Pushdown machine

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Given: plant $\mathcal{P}$, specification ϕ
Task: synthesize a controller $\mathcal{C}$ which ensures that all possible system behaviours satisfy ϕ

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Given:	plant $\mathcal{P}$, specification ϕ
Task:	synthesize a controller $\mathcal{C}$ which ensures that all possible system behaviours satisfy ϕ

parts of: traffic controls, power plants, fuel injection systems, ...

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Given: plant $\mathcal{P}$, specification ϕ
Task: synthesize a controller $\mathcal{C}$ which ensures that all possible system behaviours satisfy ϕ

parts of: traffic controls, power plants, fuel injection systems, ...

- Ongoing interplay between controller and plant

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Given: plant $\mathcal{P}$, specification ϕ
Task: synthesize a controller $\mathcal{C}$ which ensures that all possible system behaviours satisfy ϕ

parts of: traffic controls, power plants, fuel injection systems, ...

- Ongoing interplay between controller and plant
  $\rightsquigarrow$ nonterminating interaction

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Given: plant $\mathcal{P}$, specification ϕ

Task: synthesize a controller $\mathcal{C}$ which ensures that all possible system behaviours satisfy ϕ

parts of: traffic controls, power plants, fuel injection systems, ...

- Ongoing interplay between controller and plant
  $\rightsquigarrow$ nonterminating interaction
- ! Certain events in the system are uncontrollable

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Given: plant $\mathcal{P}$, specification ϕ

Task: synthesize a controller $\mathcal{C}$ which ensures that all possible system behaviours satisfy ϕ

parts of: traffic controls, power plants, fuel injection systems, ...

- Ongoing interplay between controller and plant
  $\rightsquigarrow$ nonterminating interaction

! Certain events in the system are uncontrollable
  $\rightsquigarrow$ antagonistic environment/nature

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Given: plant $\mathcal{P}$, specification ϕ

Task: synthesize a controller $\mathcal{C}$ which ensures that all possible system behaviours satisfy ϕ

parts of: traffic controls, power plants, fuel injection systems, ...

- Ongoing interplay between controller and plant
  $\rightsquigarrow$ nonterminating interaction
- ! Certain events in the system are uncontrollable
  $\rightsquigarrow$ antagonistic environment/nature
- Specification

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Given: plant $\mathcal{P}$, specification ϕ
Task: synthesize a controller $\mathcal{C}$ which ensures that all possible system behaviours satisfy ϕ

parts of: traffic controls, power plants, fuel injection systems, ...

- Ongoing interplay between controller and plant
  $\rightsquigarrow$ nonterminating interaction
- ! Certain events in the system are uncontrollable
  $\rightsquigarrow$ antagonistic environment/nature
- Specification $\rightsquigarrow$ winning condition

Strategy Synthesis: Application in Computer Science

Controller-Synthesis for open reactive systems:

Given: plant $\mathcal{P}$, specification ϕ
Task: synthesize a controller $\mathcal{C}$ which ensures that all possible system behaviours satisfy ϕ

parts of: traffic controls, power plants, fuel injection systems, ...

- Ongoing interplay between **controller** and plant
  $\rightsquigarrow$ nonterminating interaction
- ! Certain events in the system are uncontrollable
  $\rightsquigarrow$ antagonistic **environment/nature**
- **Specification** $\rightsquigarrow$ winning condition

Strategy Synthesis: Extensions

Church's Circuit Synthesis Problem, 1957:
Two-player games with full information and regular winning conditions written in MSO

Strategy Synthesis: Extensions

Church's Circuit Synthesis Problem, 1957:
Two-player games with full information and regular winning conditions written in MSO

Theorem (Büchi, Landweber 1969)

Given an MSO-formula ϕ, one can decide whether there exists a winning strategy σ for player 1. If so, a finite state winning strategy can be constructed effectively.

Strategy Synthesis: Extensions

Strategy Synthesis: Extensions

Extensions:

Strategy Synthesis: Extensions

Extensions:

- partial information
  controllers usually don't have full information about all events

Strategy Synthesis: Extensions

Extensions:

- partial information
  controllers usually don't have full information about all events
- multiple players
  certain single controller solutions may not be constructible

Strategy Synthesis: Extensions

Extensions:

- partial information
  controllers usually don't have full information about all events
- multiple players
  certain single controller solutions may not be constructible
- context-free winning conditions
  infinite state aspects

Strategy Synthesis: Extensions

Extensions:

- partial information
  controllers usually don't have full information about all events
- multiple players
  certain single controller solutions may not be constructible
- context-free winning conditions
  infinite state aspects

Other Extensions: Asynchronous Systems, Stochastic Systems, Hybrid Systems, . . .

Strategy Synthesis: Computational Hardness

Strategy Synthesis: Computational Hardness

Theorem (Reif 1979)

Strategy problem is EXPTIME*-hard (complete) for two players and safety winning conditions.*

Strategy Synthesis: Computational Hardness

Theorem (Reif 1979)

Strategy problem is EXPTIME*-hard (complete) for two players and safety winning conditions.*

Theorem (Pnueli, Rosner 1990)

Strategy problem is undecidable (not recursively enumerable) for three players and safety winning conditions.

Strategy Synthesis: Computational Hardness

Theorem (Reif 1979)

Strategy problem is EXPTIME*-hard (complete) for two players and safety winning conditions.*

Theorem (Pnueli, Rosner 1990)

Strategy problem is undecidable (not recursively enumerable) for three players and safety winning conditions.

- strategy problem for safety games with full information can be solved in linear time
- under full information, grand coalition reduces to single player

Strategy Synthesis: Refinements

What are the relevant parameters of strategy synthesis?

Strategy Synthesis: Refinements

What are the relevant parameters of strategy synthesis?

- number of (cooperating) players
- complexity of winning condition:
  - expressive complexity of winning condition
  - information complexity of winning condition
    (extent to which winning conditions may involve facts that the player(s) cannot observe)
- complexity of information flow between the players
- structural complexity of game graph

Topics

- Parity Games on Simple Graphs
  Joint work with Roman Rabinovich
- Locally decomposable winning conditions
  Joint work with Wladimir Fridman
- Knowledge tracking for multiplayer games
  Joint work with Dietmar Berwanger and Łukasz Kaiser

Topics

- Parity Games on Simple Graphs
  Joint work with Roman Rabinovich
- Locally decomposable winning conditions
  Joint work with Wladimir Fridman
- **Knowledge tracking for multiplayer games**
  Joint work with Dietmar Berwanger and Łukasz Kaiser

Parity Games on Simple Graphs

- **Two**-player games on finite graphs $\mathcal{G} = (V, \delta, \sim_1)$
- **Parity** conditions:
  col : $V \to \{0, \ldots, k\}$, $\min \mathrm{col} \inf v_0 v_1 v_2 \ldots$ even
- **Observable** coloring: $v \sim_1 w \quad \Rightarrow \quad \mathrm{col}(v) = \mathrm{col}(w)$

Parity Games on Simple Graphs: Full Information

- Strategy problem for parity games with full information is in NP $\cap$ co-NP
- Not known whether it is in $\textsc{Ptime}$

Parity Games on Simple Graphs: Full Information

- Strategy problem for parity games with full information is in NP $\cap$ co-NP
- Not known whether it is in $\textsc{Ptime}$

Theorem (Berwanger, Dawar, Hunter, Kreutzer 2006)

Strategy problem for parity games with full information can be solved in polynomial time on graphs of bounded DAG-width.

DAG-width is a measure of structural complexity for directed graphs:
How close is a graph to a directed acyclic graph (DAG)?

Parity Games on Simple Graphs: Partial Information

How about parity games on game graphs with partial information?

Theorem (Reif 1979)

Strategy problem for safety games on finite game graphs with partial information information is EXPTIME*-hard.*

Parity Games on Simple Graphs: Partial Information

How about parity games on game graphs with partial information?

Theorem (Reif 1979)

Strategy problem for safety games on finite game graphs with partial information information is EXPTIME*-hard.*

But, we still might hope that structural complexity of game graphs has a sufficiently large influence on the computational complexity ...

Parity Games on Simple Graphs: Partial Information

How about parity games on game graphs with partial information?

Theorem (Reif 1979)

Strategy problem for safety games on finite game graphs with partial information information is EXPTIME*-hard.*

But, we still might hope that structural complexity of game graphs has a sufficiently large influence on the computational complexity ...

Theorem (2010)

Strategy problem for safety games on finite game graphs with partial information information is

- EXPTIME*-hard on graphs of DAG-width at most* 3
- PSPACE*-hard on graphs of DAG-width at most* 1.

Parity Games on Simple Graphs: Bounded Partial Information

- The intrinsic complexity caused by partial information is high, even on very simple graphs

Parity Games on Simple Graphs: Bounded Partial Information

- The intrinsic complexity caused by partial information is high, even on very simple graphs
- What about bounded partial information?
- Each equivalence class of positions (w.r.t. $\sim_1$) has size at most **r**

Parity Games on Simple Graphs: Bounded Partial Information

- The intrinsic complexity caused by partial information is high, even on very simple graphs
- What about bounded partial information?
- Each equivalence class of positions (w.r.t. $\sim_1$) has size at most **r**

Theorem (2011)

Strategy problem for parity games on game graphs with bounded partial information and bounded DAG-width can be solved in polynomial time.

Locally Decomposable Winning Conditions

- Arbitrarily many players
- Regular and context-free winning conditions

Locally Decomposable Winning Conditions

- Arbitrarily many players
- Regular and context-free winning conditions

Theorem (Pnueli, Rosner 1990)

Strategy problem is undecidable for three players and safety winning conditions.

$\rightsquigarrow$ Decidability rather than complexity

(Un-)Decidability of Strategy Problem

(Un-)Decidability of Strategy Problem

Theorem (Pnueli, Rosner 1990)

Strategy problem is decidable for regular winning conditions in the case of hierarchical information: $\sim_1 \subseteq \ldots \subseteq \sim_n$.

(Un-)Decidability of Strategy Problem

Theorem (Pnueli, Rosner 1990)

Strategy problem is decidable for regular winning conditions in the case of hierarchical information: $\sim_1 \subseteq \ldots \subseteq \sim_n$.

Theorem (Finkbeiner, Schewe 2005)

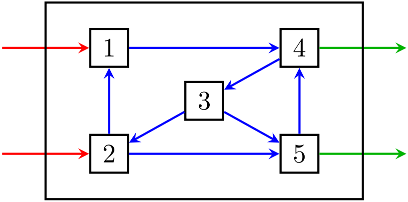
Strategy problem is undecidable for regular winning conditions **iff** *the communication graph contains incomparably informed players.*

Communication Graphs

Different representation of interactive scenarios:
No predefined game graph!

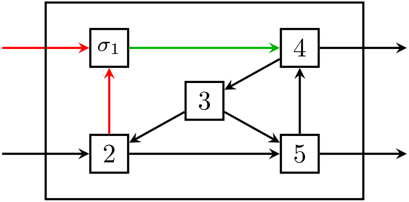
Communication Graphs

Different representation of interactive scenarios:
No predefined game graph!



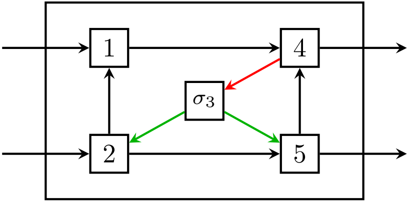
Communication Graphs

Different representation of interactive scenarios:
No predefined game graph!



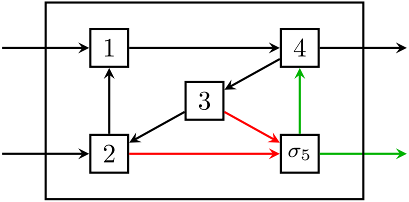
Communication Graphs

Different representation of interactive scenarios:
No predefined game graph!



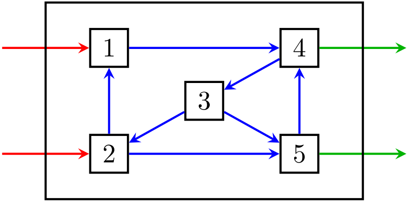
Communication Graphs

Different representation of interactive scenarios:
No predefined game graph!



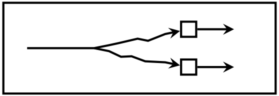
Communication Graphs

Different representation of interactive scenarios:
No predefined game graph!



Theorem (Finkbeiner, Schewe 2005)

Strategy problem is undecidable for regular winning conditions iff the communication graph contains incomparably informed players.



Locally Decomposable Winning Conditions

In this setting, the information-complexity of the winning condition is not limited!

Locally Decomposable Winning Conditions

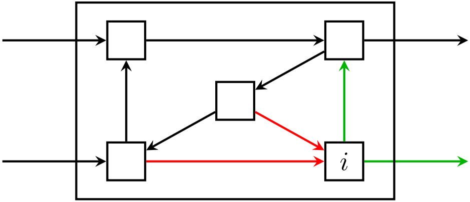
In this setting, the information-complexity of the winning condition is not limited!

Madhusudan, Thiagarajan 2001:
Locally Decomposable Winning Conditions

Locally Decomposable Winning Conditions

In this setting, the information-complexity of the winning condition is not limited!

Madhusudan, Thiagarajan 2001:
Locally Decomposable Winning Conditions

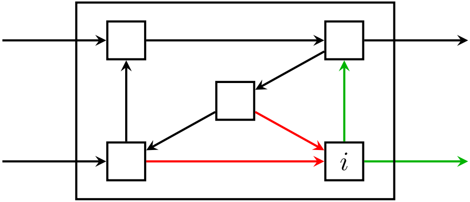


- Local Winning Conditions: W_i for player i

Locally Decomposable Winning Conditions

In this setting, the information-complexity of the winning condition is not limited!

Madhusudan, Thiagarajan 2001:
Locally Decomposable Winning Conditions



- Local Winning Conditions: W_i for player i
- Global Winning Condition $W = \bigcap W_i$

Locally Decomposable Winning Conditions: Regular Case

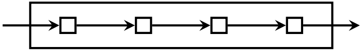
Theorem (Madhusudan, Thiagarajan 2001)

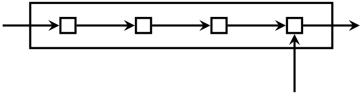
Strategy problem is decidable for locally decomposable regular winning conditions **iff** *each connected component of the communication graph is a pipeline or a two-flanked pipeline.*

Locally Decomposable Winning Conditions: Regular Case

Theorem (Madhusudan, Thiagarajan 2001)

Strategy problem is decidable for locally decomposable regular winning conditions **iff** *each connected component of the communication graph is a pipeline or a two-flanked pipeline.*

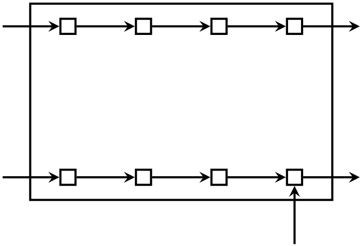




Locally Decomposable Winning Conditions: Regular Case

Theorem (Madhusudan, Thiagarajan 2001)

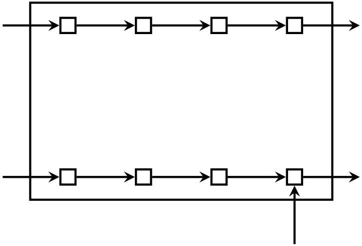
Strategy problem is decidable for locally decomposable regular winning conditions **iff** *each connected component of the communication graph is a pipeline or a two-flanked pipeline.*



Locally Decomposable Winning Conditions: Regular Case

Theorem (Madhusudan, Thiagarajan 2001)

Strategy problem is decidable for locally decomposable regular winning conditions **iff** *each connected component of the communication graph is a pipeline or a two-flanked pipeline.*



They considered only acyclic communication graphs!

Locally Decomposable Winning Conditions: Extensions

Extensions:

- (Deterministic) Context-free winning conditions
- Communication graphs may contain cycles

Locally Decomposable Winning Conditions: Extensions

Extensions:

- (Deterministic) Context-free winning conditions
- Communication graphs may contain cycles

Main Result

Characterization of communication graphs with decidable strategy problem.

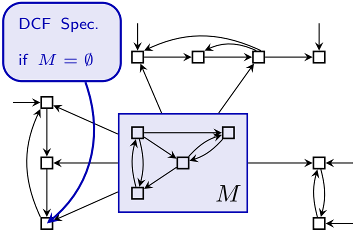
Locally Decomposable Winning Conditions: Extensions

Extensions:

- (Deterministic) Context-free winning conditions
- Communication graphs may contain cycles

Main Result

Characterization of communication graphs with decidable strategy problem.



Knowledge Tracking for Multiplayer Games

Knowledge Tracking for Multiplayer Games

Observation:
Information flow between the players is a crucial parameter

Knowledge Tracking for Multiplayer Games

Observation:
Information flow between the players is a crucial parameter

Questions:
Why does complex information flow cause the strategy problem to be computationally hard?

Knowledge Tracking for Multiplayer Games

Observation:
Information flow between the players is a crucial parameter

Questions:
Why does complex information flow cause the strategy problem to be computationally hard?
Can we handle the information flow in a generic way?

Knowledge Tracking for Multiplayer Games

Observation:
Information flow between the players is a crucial parameter

Questions:
Why does complex information flow cause the strategy problem to be computationally hard?
Can we handle the information flow in a generic way?

$\rightsquigarrow$ Represent all the possible states of mind of the players of the grand coalition and their dynamics explicitly

Knowledge Tracking for Multiplayer Games

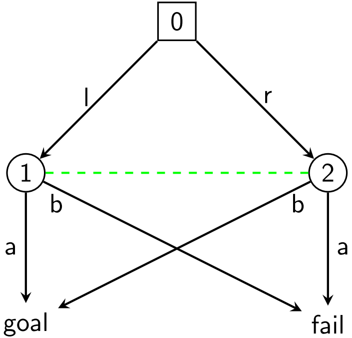
Observation:
Information flow between the players is a crucial parameter

Questions:
Why does complex information flow cause the strategy problem to be computationally hard?
Can we handle the information flow in a generic way?

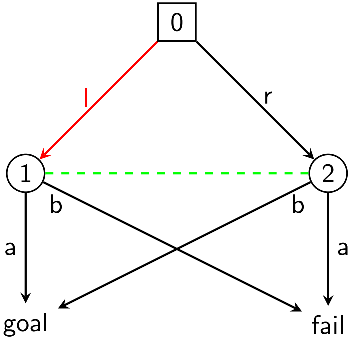
$\rightsquigarrow$ Represent all the possible states of mind of the players of the grand coalition and their dynamics explicitly

- Knowledge in multiplayer games:
  strategic dependencies and higher order knowledge
- Knowledge tracking:
  Epistemic Unfolding

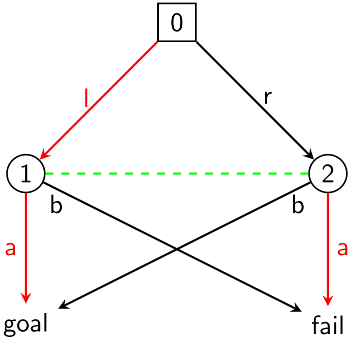
Strategic Dependencies



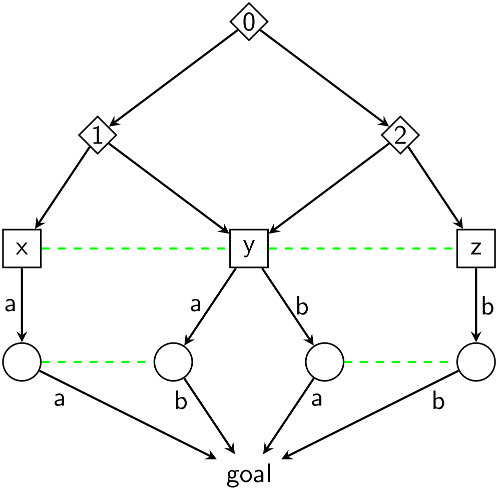
Strategic Dependencies



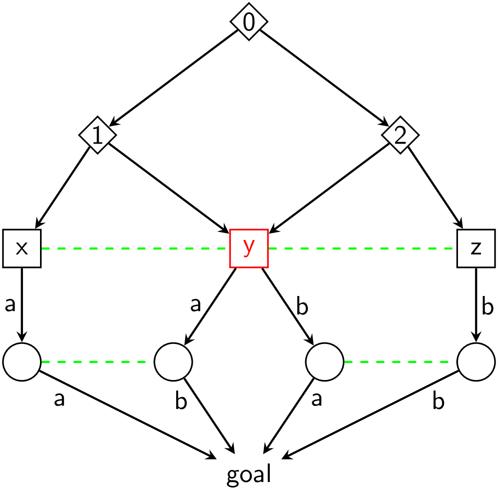
Strategic Dependencies



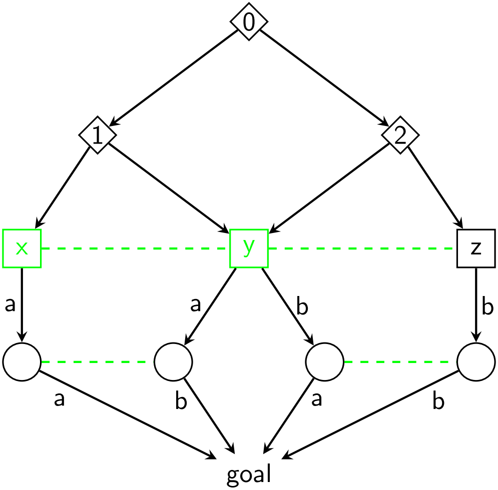
Higher Order Knowledge



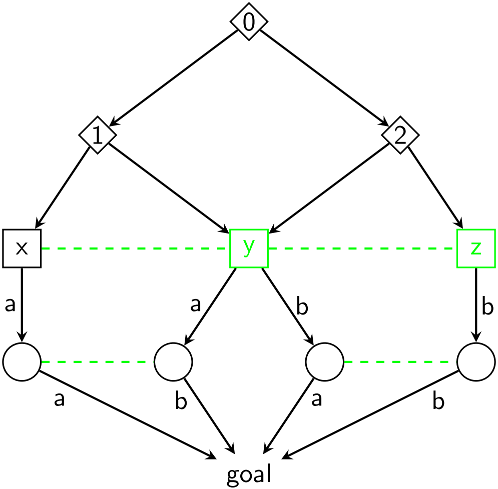
Higher Order Knowledge



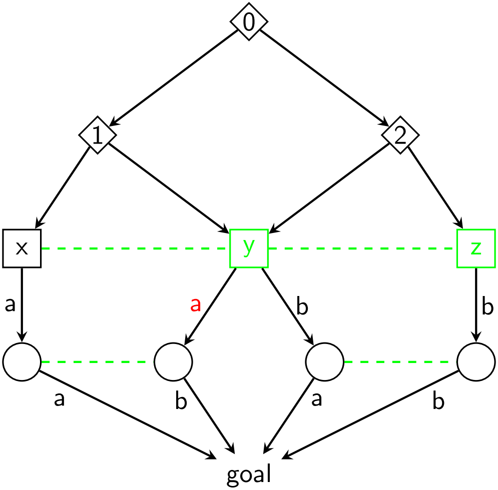
Higher Order Knowledge



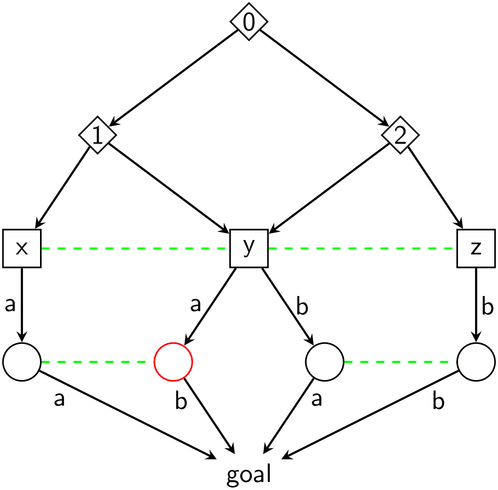
Higher Order Knowledge



Higher Order Knowledge



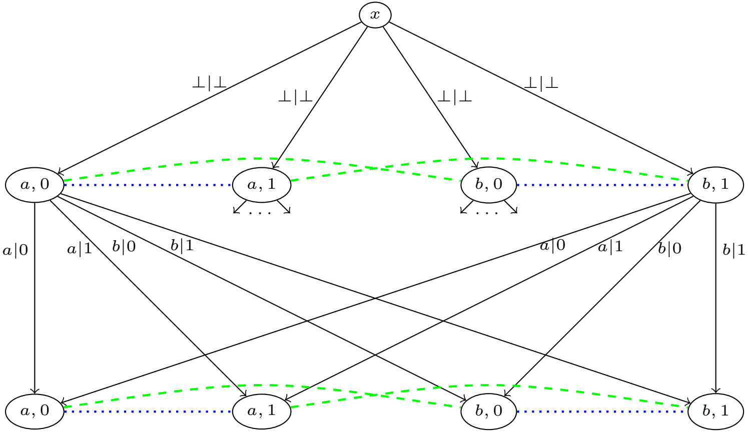
Higher Order Knowledge



Epistemic Unfolding

Knowledge tracking construction has to take both these aspects into account!

Epistemic Unfolding



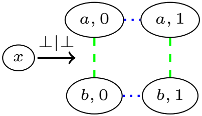
Epistemic Unfolding

x

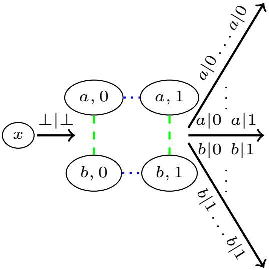
Epistemic Unfolding

$$x \xrightarrow{\perp|\perp}$$

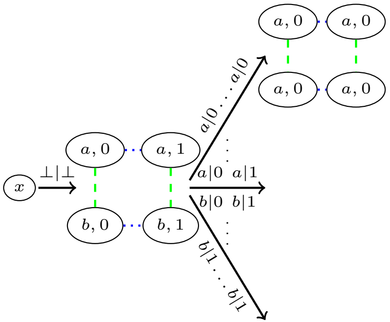
Epistemic Unfolding



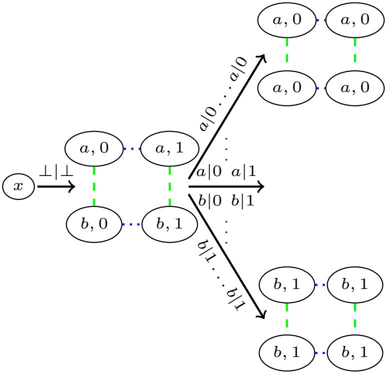
Epistemic Unfolding



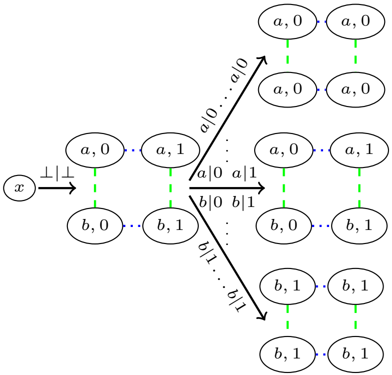
Epistemic Unfolding



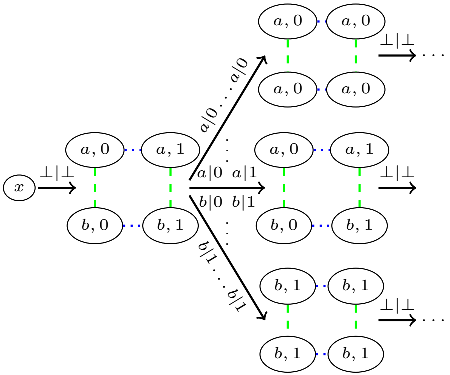
Epistemic Unfolding



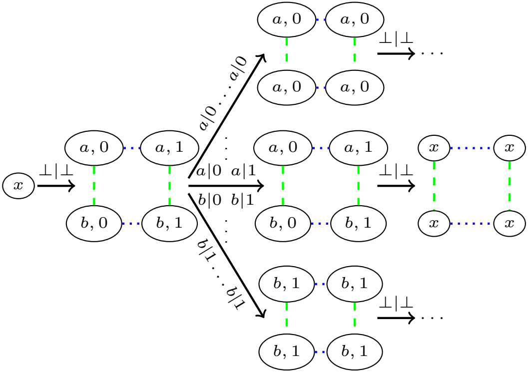
Epistemic Unfolding



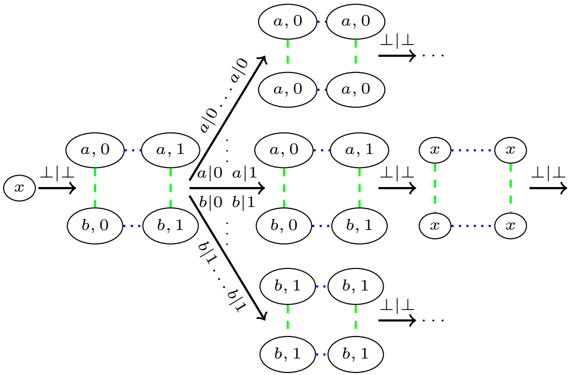
Epistemic Unfolding



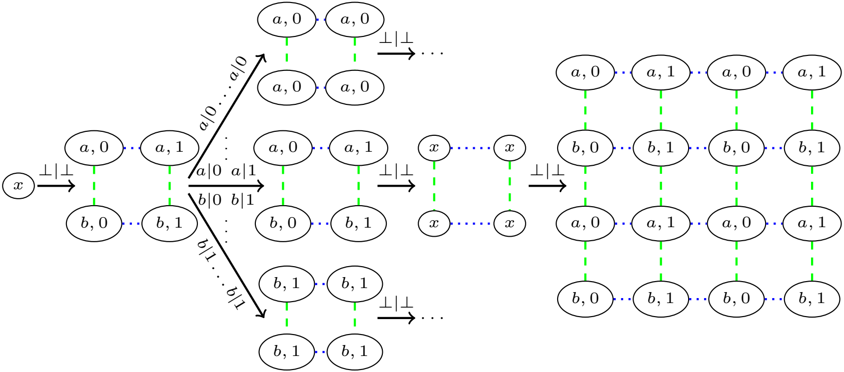
Epistemic Unfolding



Epistemic Unfolding



Epistemic Unfolding



Epistemic Unfolding

Succinct Representation: Homomorphic Equivalence!

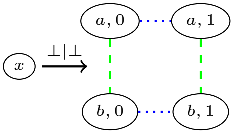
Epistemic Unfolding

Succinct Representation: Homomorphic Equivalence!

$(x) \xrightarrow{\perp|\perp}$

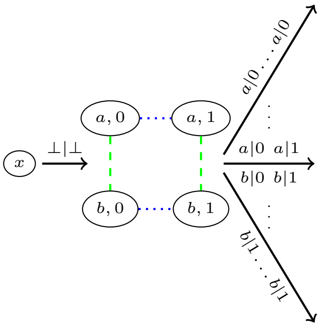
Epistemic Unfolding

Succinct Representation: Homomorphic Equivalence!



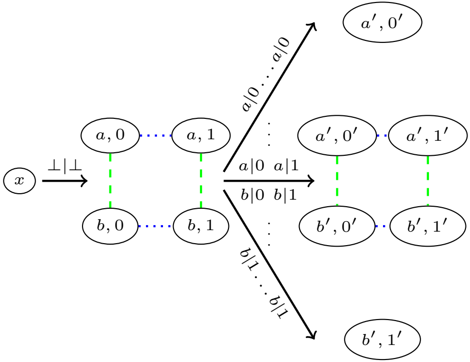
Epistemic Unfolding

Succinct Representation: Homomorphic Equivalence!



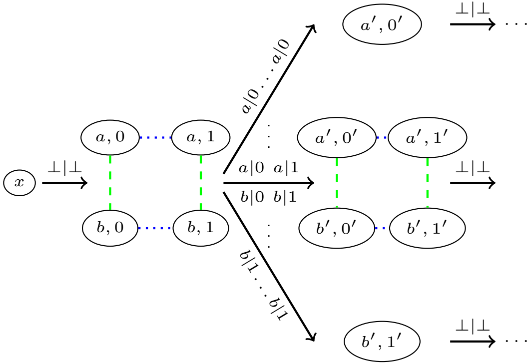
Epistemic Unfolding

Succinct Representation: Homomorphic Equivalence!



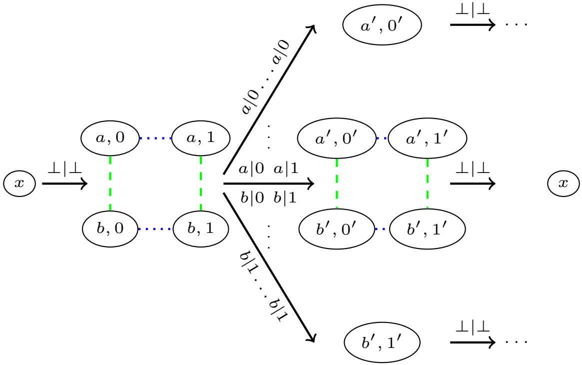
Epistemic Unfolding

Succinct Representation: Homomorphic Equivalence!



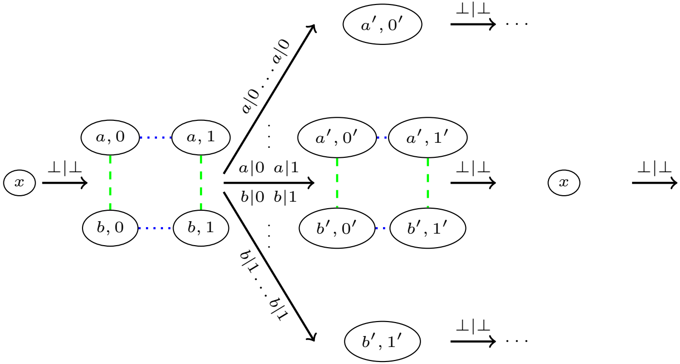
Epistemic Unfolding

Succinct Representation: Homomorphic Equivalence!



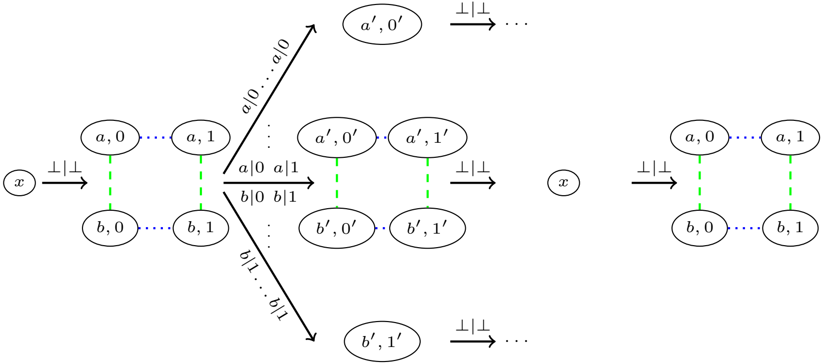
Epistemic Unfolding

Succinct Representation: Homomorphic Equivalence!



Epistemic Unfolding

Succinct Representation: Homomorphic Equivalence!



Epistemic Unfolding

For winning conditions with unrestricted information complexity, the quotient by homomorphic equivalence is not sound

Epistemic Unfolding

For winning conditions with unrestricted information complexity, the quotient by homomorphic equivalence is not sound but for observable ones it is!

Epistemic Unfolding

For winning conditions with unrestricted information complexity, the quotient by homomorphic equivalence is not sound but for observable ones it is!

Theorem

For game graphs with hierarchical partial information, the quotient modulo homomorphic equivalence is finite.

Corollary

Srategy problem for hierarchical games with observable deterministic contextfree winning conditions is decidable.

Conclusion

Parity Games on Simple Graphs:

- Safety games on finite game graphs with partial information and DAG-width at most 3 are EXPTIME-hard
- Parity games on finite game graphs with bounded partial information and bounded DAG-width can be solved in polynomial time

Locally Decomposable Specifications:

- Most communication graphs have undecidable strategy problems, even for regular winning conditions
- There are relevant cases of communication graphs, where even (deterministic) contextfree winning conditions can be allowed

Knowledge in Multiplayer Games:

- Strategic Dependencies and higher order knowledge make reasoning about knowledge cumbersome in multiplayer games
- Knowledge tracking is possible and for observable winning conditions, homomorphic equivalence yields a sound quotient