The Complexity of Admissibility in ω-Regular Games

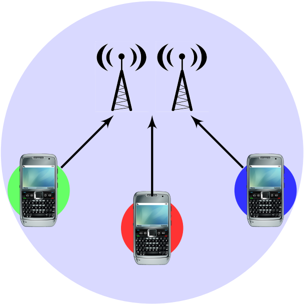
R. Brenguier J.-F. Raskin M. Sassolas

UNIVERSITÉ LIBRE DE BRUXELLES

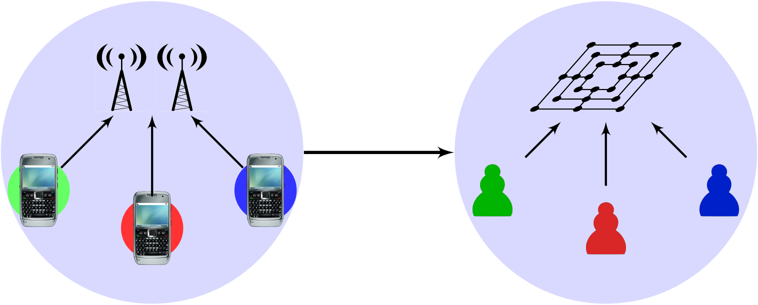
UPEC Connaissance - Action UNIVERSITÉ PARIS-EST CRÉTEIL VAL DE MARNE

Highlights of Logic, Games and Automata
21st of September 2013

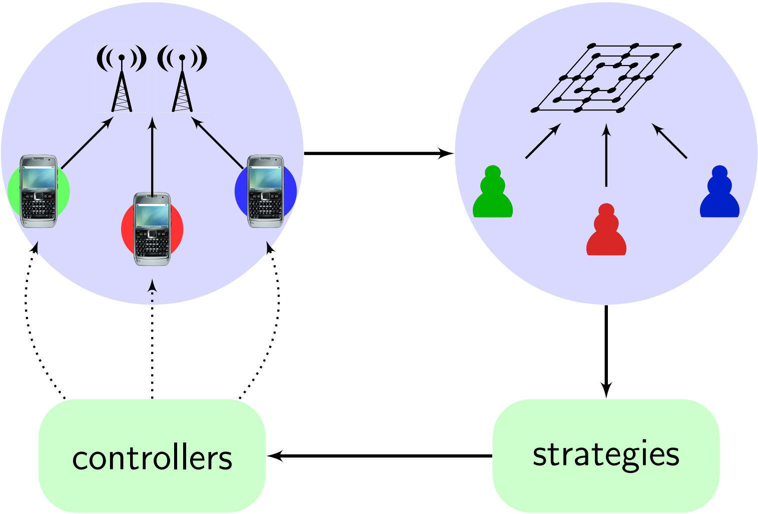
Controller synthesis



Controller synthesis



Controller synthesis



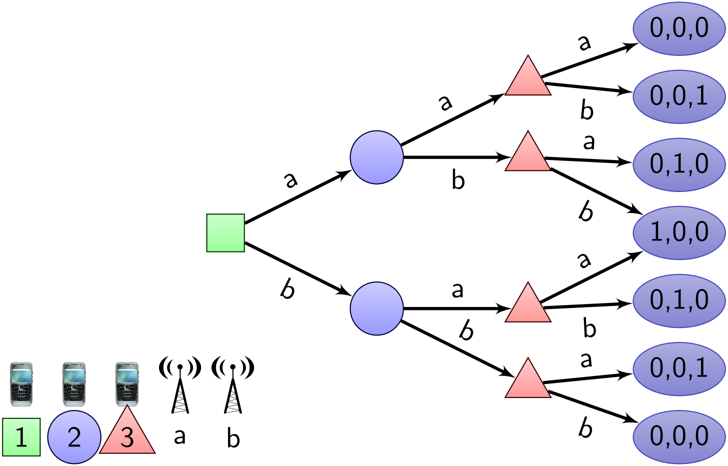
Models of rationality

- Nash equilibria $\rightsquigarrow$ no player has interest in deviating.
- Regret minimization $\rightsquigarrow$ players prefer moves that would induce less regret had they known the other players strategy.
- Elimination of dominated strategies $\rightsquigarrow$ players eliminate “bad” strategies

$\hookrightarrow$ In all cases it is assumed everybody knows and uses the model of rationality.

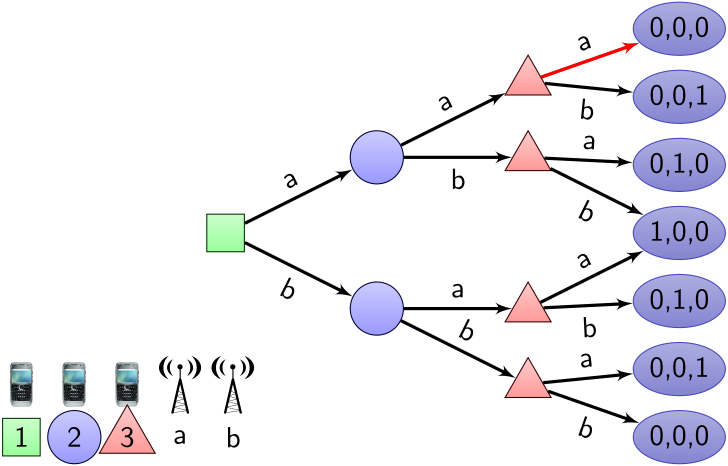
Iterative elimination of dominated strategies

- What is a "bad" strategy? σ is strictly dominated by σ' if
  - for all profiles of the other players, if σ wins, so does σ'.
  - for some profile of the other players, σ loses while σ' wins.



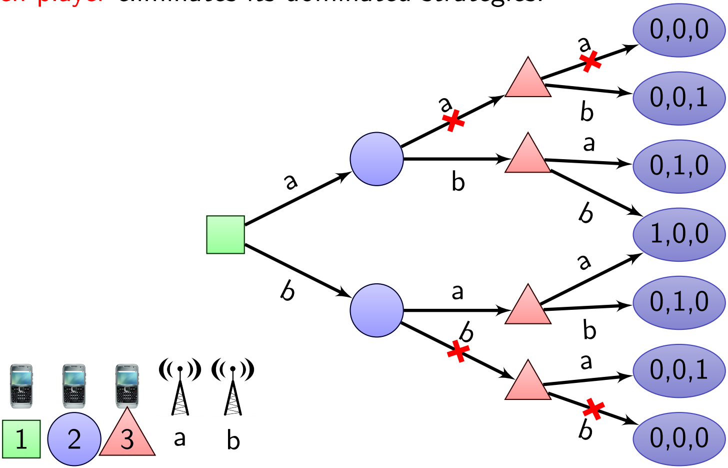
Iterative elimination of dominated strategies

- What is a "bad" strategy? σ is strictly dominated by σ' if
  - for all profiles of the other players, if σ wins, so does σ'.
  - for some profile of the other players, σ loses while σ' wins.



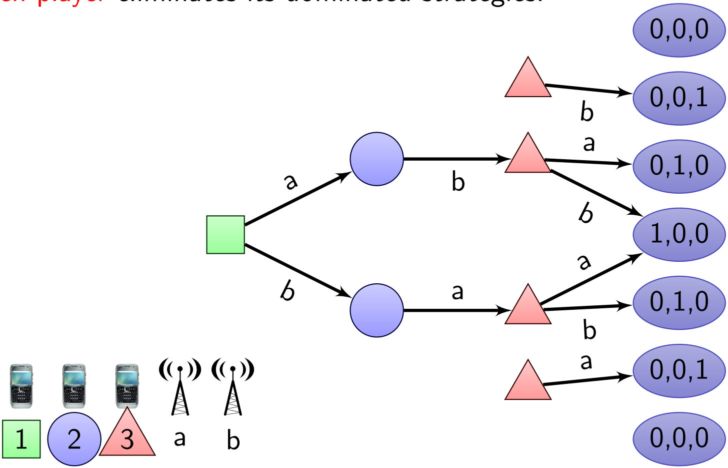
Iterative elimination of dominated strategies

- What is a "bad" strategy? σ is strictly dominated by σ' if
  - for all profiles of the other players, if σ wins, so does σ'.
  - for some profile of the other players, σ loses while σ' wins.
- Each player eliminates its dominated strategies.



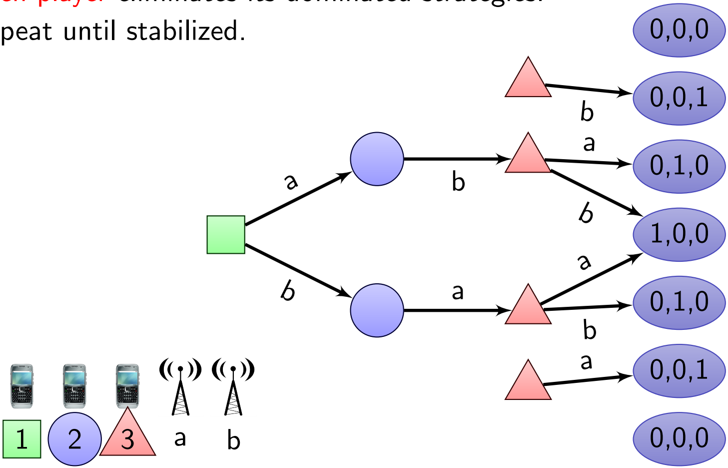
Iterative elimination of dominated strategies

- What is a "bad" strategy? σ is strictly dominated by σ' if
  - for all profiles of the other players, if σ wins, so does σ'.
  - for some profile of the other players, σ loses while σ' wins.
- Each player eliminates its dominated strategies.



Iterative elimination of dominated strategies

- What is a "bad" strategy? σ is strictly dominated by σ' if
  - for all profiles of the other players, if σ wins, so does σ'.
  - for some profile of the other players, σ loses while σ' wins.
- Each player eliminates its dominated strategies.
- Repeat until stabilized.



Our setting

- Turn based games on graphs.
- Objective of player i: $\textsc{Win}_i \subseteq V^\omega$.

Our setting

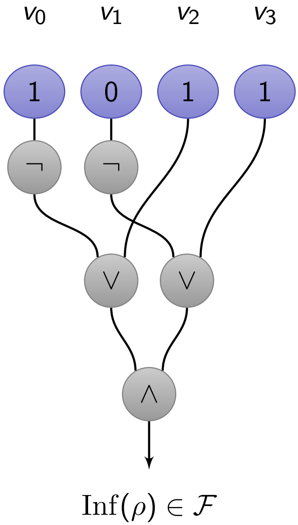
- Turn based games on graphs.
- Objective of player i: $\textsc{Win}_i \subseteq V^\omega$.

- Muller objectives:
  $\rho \in \textsc{Win}_i$ iff $\mathrm{Inf}(\rho) \in \mathcal{F}$.

$\rightsquigarrow$ Generalizes Büchi and parity conditions.

- Weak Muller objectives:
  $\rho \in \textsc{Win}_i$ iff $\mathrm{Occ}(\rho) \in \mathcal{F}$.

$\rightsquigarrow$ Generalizes safety and reachability conditions.



Admissibility

- Dominance: $\sigma'_i \succ_{\mathcal{S}^n} \sigma_i$ if σ'_i strictly dominates σ_i w.r.t $\mathcal{S}^n$.
- Iterative admissibility: $\mathcal{S}_i^0 = \mathcal{S}_i$ and
$$\mathcal{S}_i^{n+1} := \mathcal{S}_i^n \setminus \left\{\sigma_i \mid \exists \sigma'_i \in \mathcal{S}_i^n, \sigma'_i \succ_{\mathcal{S}^n} \sigma_i\right\}.$$
- Set of iteratively admissible strategies: $\mathcal{S}^* = \bigcap_{n\in\mathbb{N}} \mathcal{S}^n$

Admissibility

- Dominance: $\sigma'_i \succ_{\mathcal{S}^n} \sigma_i$ if σ'_i strictly dominates σ_i w.r.t $\mathcal{S}^n$.
- Iterative admissibility: $\mathcal{S}_i^0 = \mathcal{S}_i$ and
$$\mathcal{S}_i^{n+1} := \mathcal{S}_i^n \setminus \left\{\sigma_i \;\middle|\; \exists \sigma'_i \in \mathcal{S}_i^n, \sigma'_i \succ_{\mathcal{S}^n} \sigma_i\right\}.$$
- Set of iteratively admissible strategies: $\mathcal{S}^* = \bigcap_{n\in\mathbb{N}} \mathcal{S}^n$

$\hookrightarrow$ Goal: compute $\mathcal{S}^*$ or at least decide properties thereof.

Admissibility

- Dominance: $\sigma'_i \succ_{\mathcal{S}^n} \sigma_i$ if σ'_i strictly dominates σ_i w.r.t $\mathcal{S}^n$.
- Iterative admissibility: $\mathcal{S}_i^0 = \mathcal{S}_i$ and
$$\mathcal{S}_i^{n+1} := \mathcal{S}_i^n \setminus \left\{\sigma_i \;\middle|\; \exists \sigma'_i \in \mathcal{S}_i^n, \sigma'_i \succ_{\mathcal{S}^n} \sigma_i\right\}.$$
- Set of iteratively admissible strategies: $\mathcal{S}^* = \bigcap_{n\in\mathbb{N}} \mathcal{S}^n$

$\hookrightarrow$ Goal: compute $\mathcal{S}^*$ or at least decide properties thereof.

Remark

$\mathcal{S}^*$ is well defined and is reached after a finite number of iterations.

"Admissibility in Infinite Games" [Berwanger, STACS'07]

Admissibility

- Dominance: $\sigma'_i \succ_{\mathcal{S}^n} \sigma_i$ if σ'_i strictly dominates σ_i w.r.t $\mathcal{S}^n$.
- Iterative admissibility: $\mathcal{S}^0_i = \mathcal{S}_i$ and
$$\mathcal{S}^{n+1}_i := \mathcal{S}^n_i \setminus \left\{\sigma_i \;\middle|\; \exists \sigma'_i \in \mathcal{S}^n_i, \sigma'_i \succ_{\mathcal{S}^n} \sigma_i\right\}.$$
- Set of iteratively admissible strategies: $\mathcal{S}^* = \bigcap_{n\in\mathbb{N}} \mathcal{S}^n$

$\hookrightarrow$ Goal: compute $\mathcal{S}^*$ or at least decide properties thereof.

Remark

$\mathcal{S}^*$ is well defined and is reached after a finite number of iterations.

"Admissibility in Infinite Games" [Berwanger, STACS'07]

Decision problems on $\mathcal{S}^*$

The *winning coalition problem*: Given $W, L \subseteq P$, does there exists $\sigma_P \in \mathcal{S}^*$ such that all players of W win the game, and all players of L lose.
The *model-checking under admissibility problem*: Given φ an LTL formula, is it the case that for any profile $\sigma_P \in \mathcal{S}^*$, $Out(\sigma_P) \models \varphi$?

Values

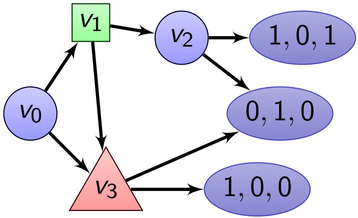
Introduced in [Berwanger, STACS'07]

- If there is a winning strategy

$\rightsquigarrow$ admissible strategies are the winning ones.

- It is impossible to win

$\rightsquigarrow$ all strategies are admissible.



Values

Introduced in [Berwanger, STACS'07]

- If there is a winning strategy

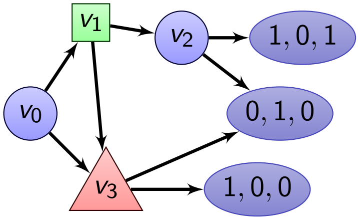
$\rightsquigarrow$ admissible strategies are the winning ones.

- It is impossible to win

$\rightsquigarrow$ all strategies are admissible.

- Otherwise: it is possible to win, but only with the help of others

$\hookrightarrow$ What are the admissible strategies in this case?



Values

Introduced in [Berwanger, STACS'07]

- If there is a winning strategy: value 1.

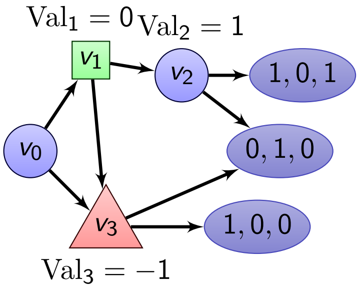
$\rightsquigarrow$ admissible strategies are the winning ones.

- It is impossible to win: value -1.

$\rightsquigarrow$ all strategies are admissible.

- Otherwise: it is possible to win, but only with the help of others: value 0.

$\hookrightarrow$ What are the admissible strategies in this case?



Values

Introduced in [Berwanger, STACS'07]

- If there is a winning strategy: value 1.

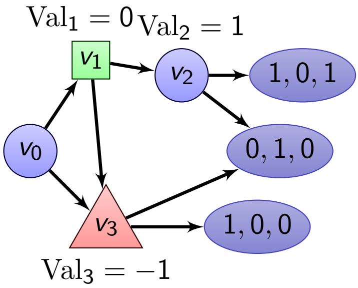
$\rightsquigarrow$ admissible strategies are the winning ones.

- It is impossible to win: value -1.

$\rightsquigarrow$ all strategies are admissible.

- Otherwise: it is possible to win, but only with the help of others: value 0.

$\hookrightarrow$ What are the admissible strategies in this case?



Remark

- *A player should never decrease its own value.*
- *The value depends on $\mathcal{S}^n$.*

$\hookrightarrow$ How to compute those values?

Safety objectives: a local notion of dominance

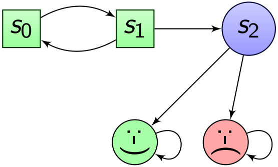
- Objective: avoid *Bad* states
- Existence of a winning strategy depends only on:
  - the current state
  - *Bad* states visited
    - $\rightsquigarrow$ *unfold* the graph to keep this information
    - $\rightsquigarrow$ size: $|V| \times 2^{|P|}$.
- In unfolded safety games the rule to never decrease one's own value is sufficient for admissibility.
- The structure of the unfolding avoid explosion in complexity.

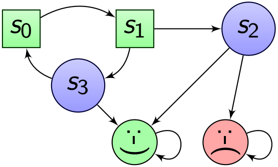
Theorem

The winning coalition problem is PSPACE*-complete for safety.*

Prefix-independent objectives

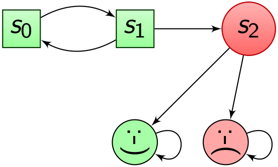
$\hookrightarrow$ In general: the local condition is not sufficient

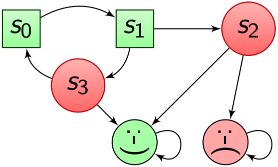




Prefix-independent objectives

$\hookrightarrow$ In general: the local condition is not sufficient



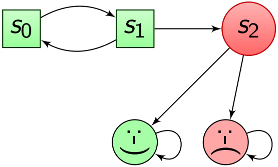


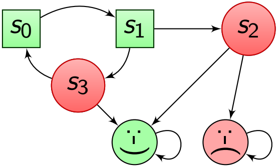
- In case the value is 0, need to allow other players to help.
- "Help!"-state for i: a state where $j \neq i$ has several choices with value ≥ 0 for i, while not changing the value for j.

$\hookrightarrow$ Admissible strategies should be winning if the other players played fairly in those states.

Prefix-independent objectives

$\hookrightarrow$ In general: the local condition is not sufficient





- In case the value is 0, need to allow other players to help.
- "Help!"-state for i: a state where $j \neq i$ has several choices with value ≥ 0 for i, while not changing the value for j.

$\hookrightarrow$ Admissible strategies should be winning if the other players played fairly in those states.

$\rightsquigarrow$ Give rises to automaton $\mathcal{A}_n$ recognizing $Out(\mathcal{S}^n)$ with circuit winning condition.

Prefix-independent objectives

$\hookrightarrow$ In general: the local condition is not sufficient

- In case the value is 0, need to allow other players to help.
- "Help!"-state for i: a state where $j \neq i$ has several choices with value ≥ 0 for i, while not changing the value for j.

$\hookrightarrow$ Admissible strategies should be winning if the other players played fairly in those states.

$\rightsquigarrow$ Give rises to automaton $\mathcal{A}_n$ recognizing $Out(\mathcal{S}^n)$ with circuit winning condition.

- In turn, $\mathcal{A}_n$ is used to compute the values at the next step.

Complexity for Objectives defined by Circuits

Theorem (Winning coalition problem)

- *The winning coalition problem* PSPACE*-complete for circuits.*
- *The winning coalition problem with Büchi objectives is in* $\mathsf{NP} \cap \mathsf{coNP}$
- *The winning coalition problem for weak circuit is* PSPACE*-complete.*

Theorem (Model-checking under admissibility problem)

The model-checking under admissibility problem is PSPACE*-complete for games where the winning condition of each player is given by a circuit condition.*

Summary

- Automata representing all outcomes of admissible strategies.
- Algorithms with tight complexity bounds to compute the set of all outcomes of iteratively admissible strategies.
- Application to model-checking of LTL assuming all players follow rationality.

Summary

- Automata representing all outcomes of admissible strategies.
- Algorithms with tight complexity bounds to compute the set of all outcomes of iteratively admissible strategies.
- Application to model-checking of LTL assuming all players follow rationality.

Future work

- Extension to quantitative games.
- Implementation.

Summary

- Automata representing all outcomes of admissible strategies.
- Algorithms with tight complexity bounds to compute the set of all outcomes of iteratively admissible strategies.
- Application to model-checking of LTL assuming all players follow rationality.

Future work

- Extension to quantitative games.
- Implementation.

Thank you