Easy to Win, Hard to Master: Playing Parity Games with Costs Optimally

Joint work with Alexander Weinert (Saarland University)

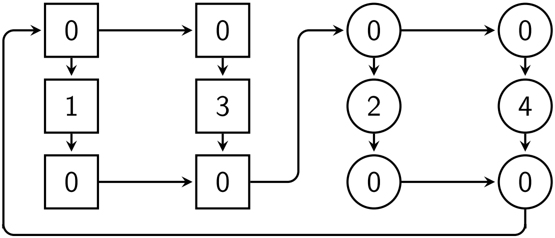
Martin Zimmermann

Saarland University

December 16th, 2016

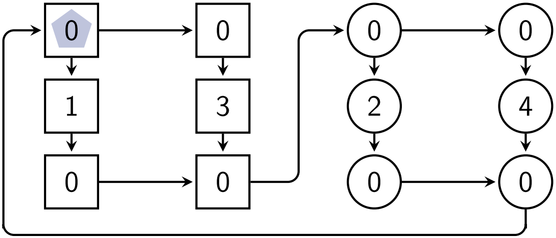
AVeRTS 2016, Chennai, India

Parity Games



Example due to Chatterjee & Fijalkow

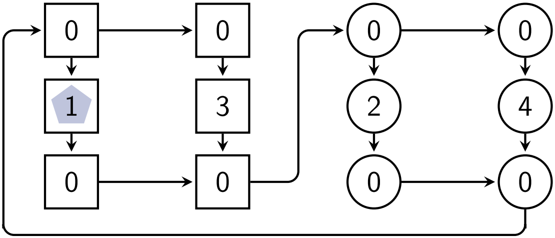
Parity Games



0

Example due to Chatterjee & Fijalkow

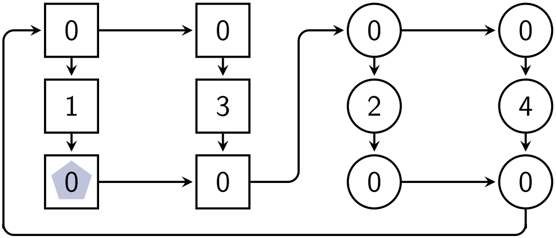
Parity Games



$0 \longrightarrow 1$

Example due to Chatterjee & Fijalkow

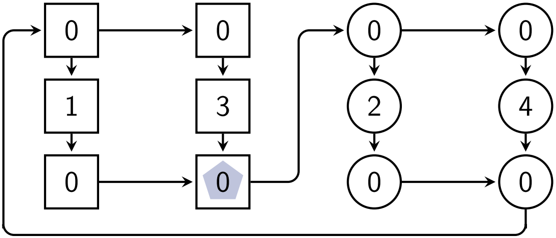
Parity Games



0 ⟶ 1 ⟶ 0

Example due to Chatterjee & Fijalkow

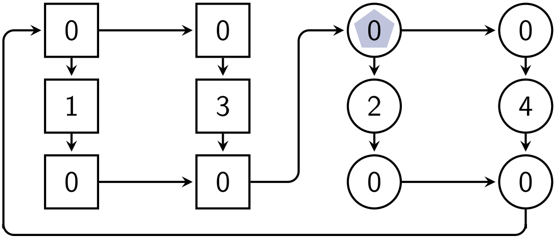
Parity Games



$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0$

Example due to Chatterjee & Fijalkow

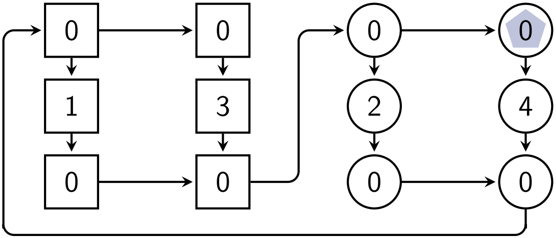
Parity Games



0 ⟶ 1 ⟶ 0 ⟶ 0 ⟶ 0

Example due to Chatterjee & Fijalkow

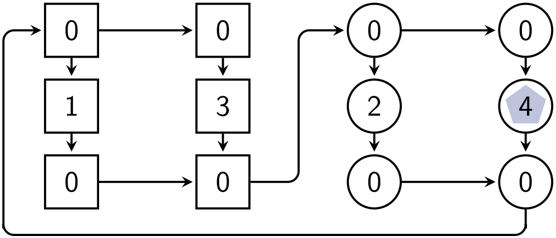
Parity Games



$$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0$$

Example due to Chatterjee & Fijalkow

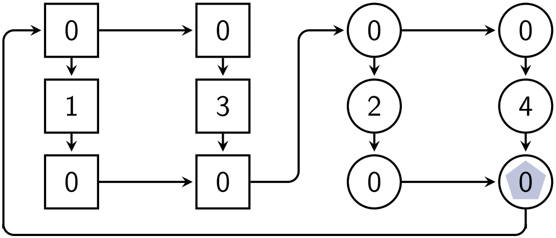
Parity Games



$$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4$$

Example due to Chatterjee & Fijalkow

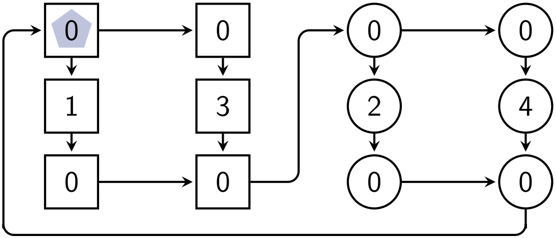
Parity Games



$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0$

Example due to Chatterjee & Fijalkow

Parity Games

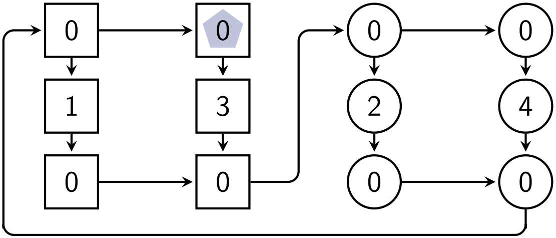


$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0$

0

Example due to Chatterjee & Fijalkow

Parity Games

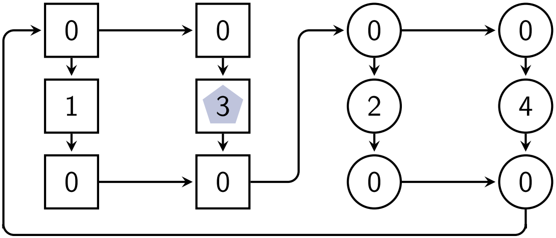


$$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0$$

$$\hookrightarrow 0 \longrightarrow 0$$

Example due to Chatterjee & Fijalkow

Parity Games

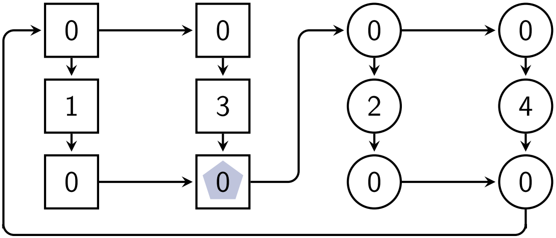


$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0$

$\longrightarrow 0 \longrightarrow 0 \longrightarrow 3$

Example due to Chatterjee & Fijalkow

Parity Games

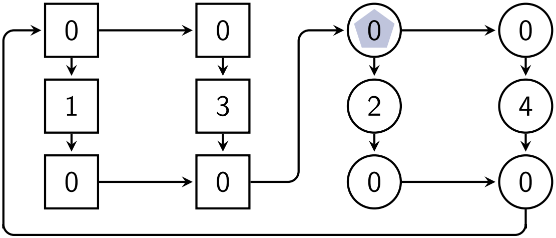


$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0$

$\longrightarrow 0 \longrightarrow 0 \longrightarrow 3 \longrightarrow 0$

Example due to Chatterjee & Fijalkow

Parity Games

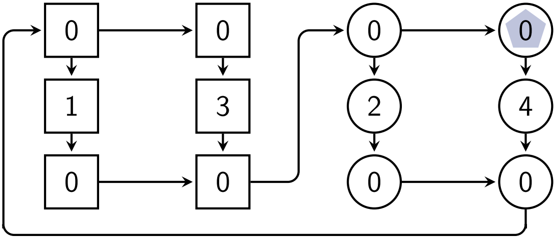


$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0$

$0 \longrightarrow 0 \longrightarrow 3 \longrightarrow 0 \longrightarrow 0$

Example due to Chatterjee & Fijalkow

Parity Games

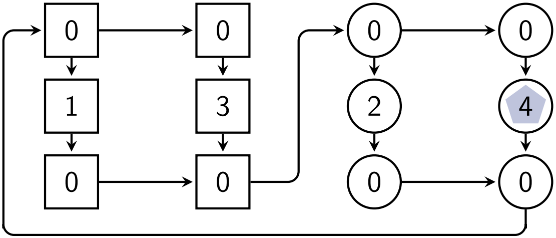


$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0$

$0 \longrightarrow 0 \longrightarrow 3 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0$

Example due to Chatterjee & Fijalkow

Parity Games

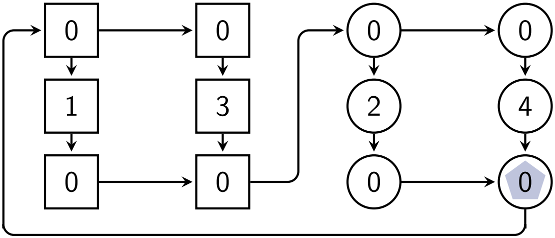


$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0$

$0 \longrightarrow 0 \longrightarrow 3 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4$

Example due to Chatterjee & Fijalkow

Parity Games

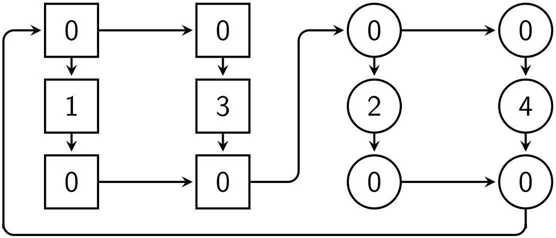


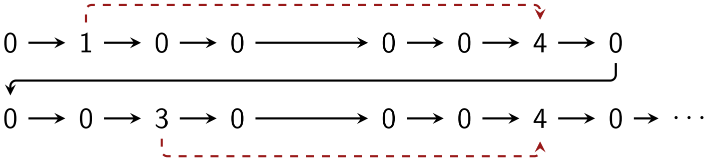
$0 \longrightarrow 1 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0$

$0 \longrightarrow 0 \longrightarrow 3 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 4 \longrightarrow 0 \longrightarrow \cdots$

Example due to Chatterjee & Fijalkow

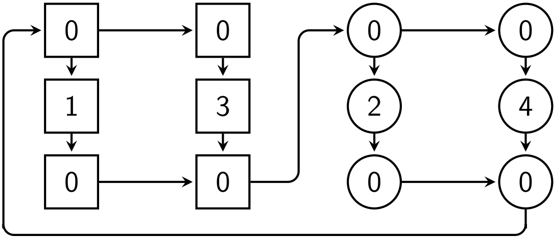
Parity Games

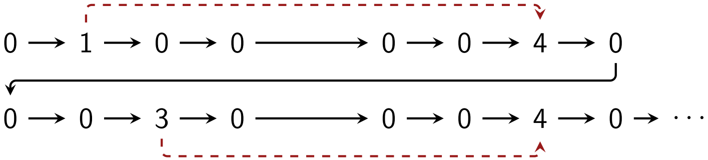




Example due to Chatterjee & Fijalkow

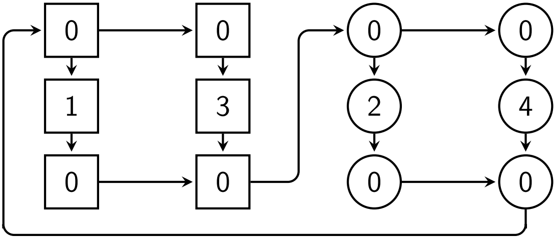
Parity Games





- Various applications: μ-calculus model checking, Rabin's theorem, reactive synthesis, alternating automata,...

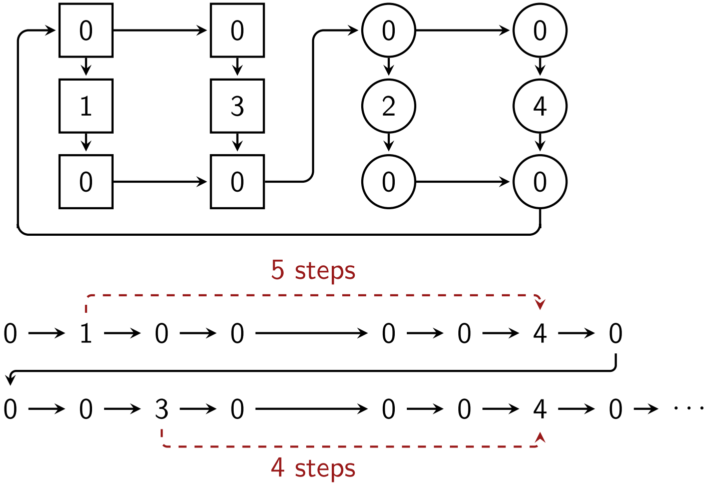
Finitary Parity Games



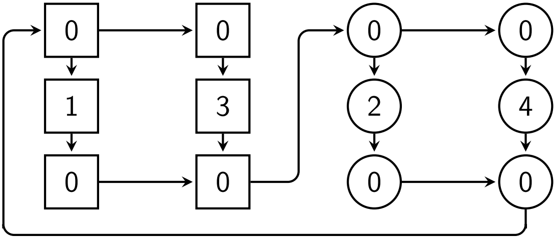
$0 \rightarrow 1 \rightarrow 0 \rightarrow 0 \longrightarrow 0 \rightarrow 0 \rightarrow 4 \rightarrow 0$

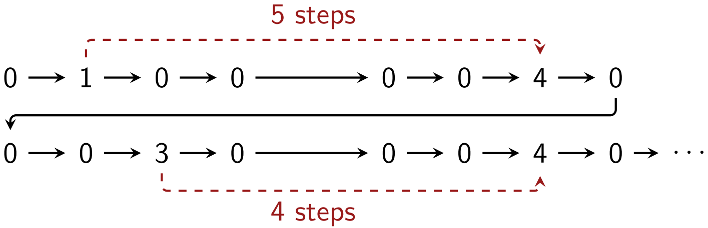
$0 \rightarrow 0 \rightarrow 3 \rightarrow 0 \longrightarrow 0 \rightarrow 0 \rightarrow 4 \rightarrow 0 \rightarrow \cdots$

Finitary Parity Games



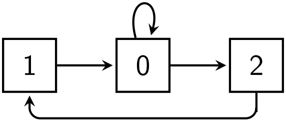
Finitary Parity Games



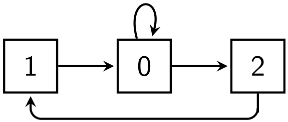


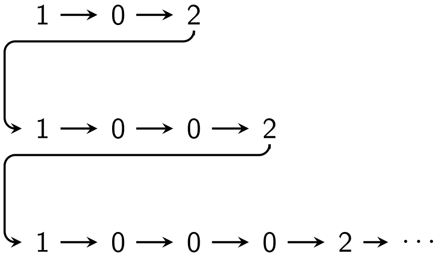
- A quantitative strengthening of parity games.

Another Example

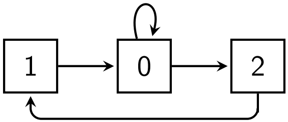


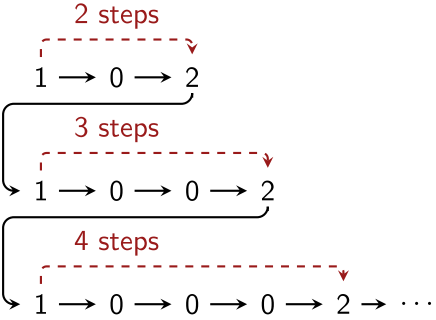
Another Example



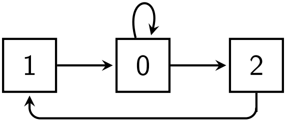


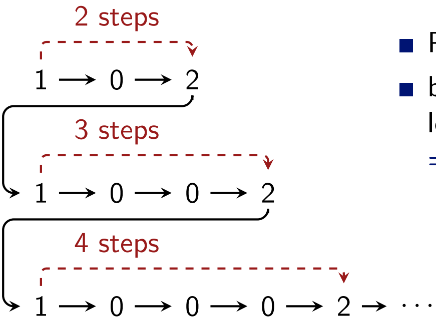
Another Example





Another Example





- Player 1 wins from every vertex,
- but needs to stay longer and longer in vertex of color 0.
  $\Rightarrow$ requires infinite memory.

Previous Work

- **Parity:** Almost all requests are answered.
- **Finitary Parity:** There is a bound b such that almost all requests are answered within b steps.

Previous Work

- **Parity:** Almost all requests are answered.
- **Finitary Parity:** There is a bound b such that almost all requests are answered within b steps.

Condition	Complexity	Memory Pl. 0	Memory Pl. 1
Parity	$\mathrm{UP} \cap \textsc{co}\text{-}\mathrm{UP}$	Memoryless	Memoryless
Finitary Parity	$\mathrm{P}\textsc{Time}$	Memoryless	Infinite

Previous Work

- **Parity:** Almost all requests are answered.
- **Finitary Parity:** There is a bound b such that almost all requests are answered within b steps.

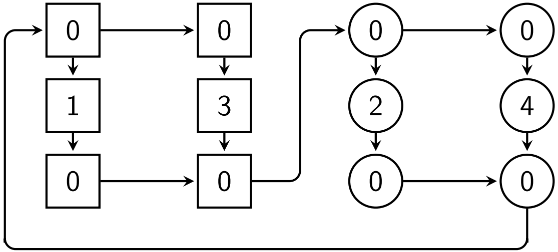
Condition	Complexity	Memory Pl. 0	Memory Pl. 1
Parity	$\mathrm{UP} \cap \textsc{co}\text{-}\mathrm{UP}$	Memoryless	Memoryless
Finitary Parity	$\mathrm{PT}\textsc{ime}$	Memoryless	Infinite

Corollary

If Player 0 wins a finitary parity game $\mathcal{G}$, then a uniform bound $b \leq |\mathcal{G}|$ suffices.

A trivial example shows that the upper bound $|\mathcal{G}|$ is tight.

Back to the Example

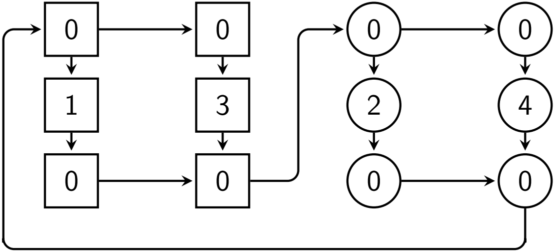


Answering requests as soon as possible requires memory.

- Every request can be answered within four steps:
  - a 1 by a 2
  - a 3 by a 4

$\Rightarrow$ requires one bit of memory.

Back to the Example



Answering requests as soon as possible requires memory.

- Every request can be answered within four steps:
  - a 1 by a 2
  - a 3 by a 4

  ⇒ requires one bit of memory.
- But answering a 1 by a 4 takes five steps.

  ⇒ every memoryless strategy has at least *cost* 5.

Playing Finitary Parity Games Optimally

Questions

1. How much memory is needed to play finitary parity games optimally?
2. How hard is it to determine the optimal bound b for a finitary parity game?
3. There is a tradeoff between size and cost of strategies! What is its extent?

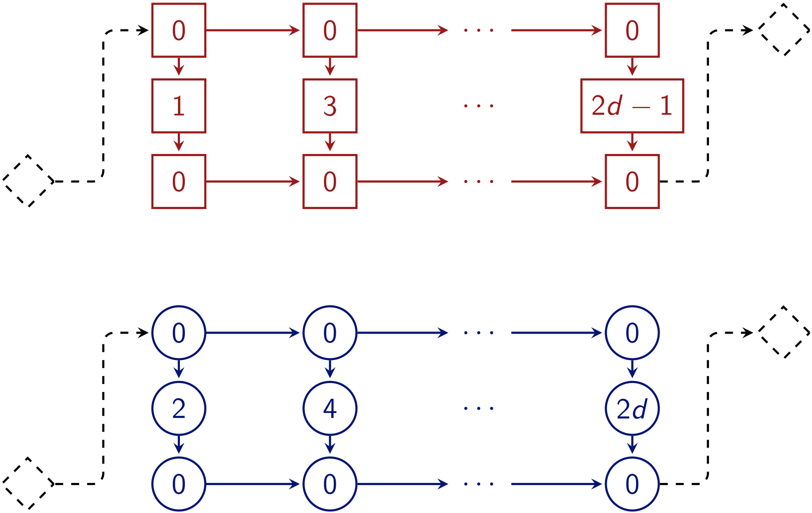
Outline

1. Memory Requirements of Optimal Strategies
2. Determining Optimal Bounds is Hard
3. Trading Memory for Quality and Vice Versa
4. Generalizations
5. Conclusion

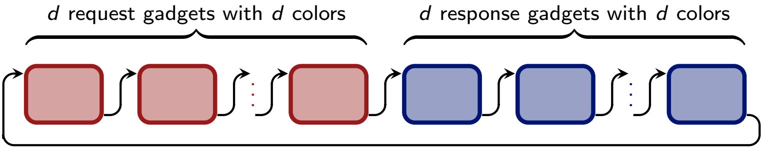
Outline

1. **Memory Requirements of Optimal Strategies**
2. Determining Optimal Bounds is Hard
3. Trading Memory for Quality and Vice Versa
4. Generalizations
5. Conclusion

Memory Requirements

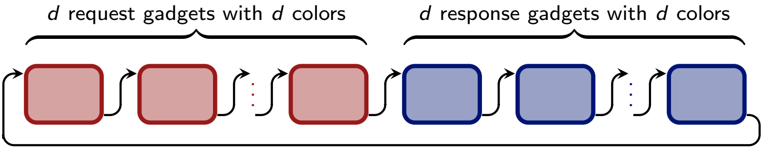


Memory Requirements



- Player 0 has winning strategy with cost $d^2 + 2d$: answer j-th unique request in j-th response-gadget.
  $\Rightarrow$ requires exponential memory (in d).
- Against a smaller strategy Player 1 can enforce a larger cost, as Player 0 cannot store every sequence of requests.

Memory Requirements



- Player 0 has winning strategy with cost $d^2 + 2d$: answer j-th unique request in j-th response-gadget.
  $\Rightarrow$ requires exponential memory (in d).
- Against a smaller strategy Player 1 can enforce a larger cost, as Player 0 cannot store every sequence of requests.

Theorem

For every $d > 1$, there exists a finitary parity game $\mathcal{G}_d$ such that

- $|\mathcal{G}_d| \in \mathcal{O}(d^2)$ *and* $\mathcal{G}_d$ *has d odd colors, and*
- *every optimal strategy for Player* 0 *has at least size* $2^d - 2$.

Outline

PSPACE-Hardness

Lemma

The following problem is PSPACE*-hard: "Given a finitary parity game* $\mathcal{G}$ *and a bound* $b \in \mathbb{N}$*, does Player* 0 *have a strategy for* $\mathcal{G}$ *whose cost is at most* b*?"*

PSPACE-Hardness

Lemma

The following problem is $\mathrm{PS}\textsc{pace}$*-hard: "Given a finitary parity game* $\mathcal{G}$ *and a bound* $b \in \mathbb{N}$*, does Player* 0 *have a strategy for* $\mathcal{G}$ *whose cost is at most* b*?"*

Proof

- By a reduction from QBF (w.l.o.g. in CNF).
- Checking the truth of $\varphi = \forall x \exists y.\ (x \vee \neg y) \wedge (\neg x \vee y)$ as a two-player game (Player 0 wants to prove truth of φ):

PSPACE-Hardness

Lemma

The following problem is $\textsc{PSpace}$*-hard: "Given a finitary parity game* $\mathcal{G}$ *and a bound* $b \in \mathbb{N}$*, does Player* 0 *have a strategy for* $\mathcal{G}$ *whose cost is at most* b*?"*

Proof

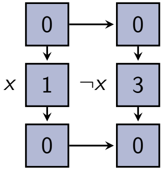
- By a reduction from QBF (w.l.o.g. in CNF).
- Checking the truth of $\varphi = \forall x \exists y.\ (x \vee \neg y) \wedge (\neg x \vee y)$ as a two-player game (Player 0 wants to prove truth of φ):
  1. Player 1 picks truth value for x.
  2. Player 0 picks truth value for y.
  3. Player 1 picks clause C.
  4. Player 0 picks literal ℓ from C.
  5. Player 0 wins $\Leftrightarrow$ ℓ is picked to be satisfied in step 1 or 2.

The Reduction

$$\varphi = \forall x \ \exists y \ . \ \overbrace{(\, x \ \vee \ \neg y \,) \wedge (\, \neg x \ \vee \ y \,)}^{\psi}$$

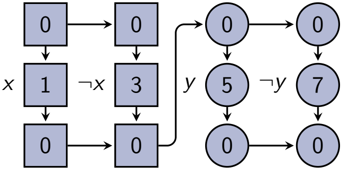
The Reduction

$$\varphi = \forall x \; \exists y \, . \; \overbrace{(\, x \;\vee\; \neg y \,) \wedge (\, \neg x \;\vee\; y \,)}^{\psi}$$



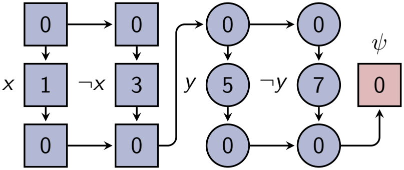
The Reduction

$$\varphi = \forall x\ \exists y.\ \overbrace{(x \vee \neg y) \wedge (\neg x \vee y)}^{\psi}$$



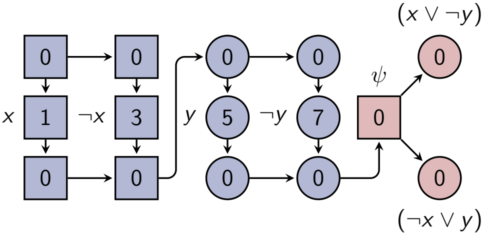
The Reduction

$$\varphi = \forall x\ \exists y\ .\ \overbrace{(\ x\ \vee\ \neg y\ ) \wedge (\ \neg x\ \vee\ y\ )}^{\psi}$$



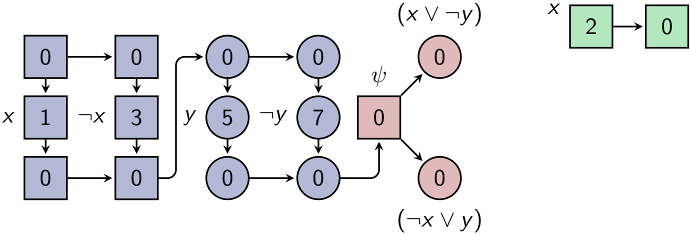
The Reduction

$$\varphi = \forall x\ \exists y\ .\ \overbrace{(\,x \vee \neg y\,) \wedge (\,\neg x \vee y\,)}^{\psi}$$



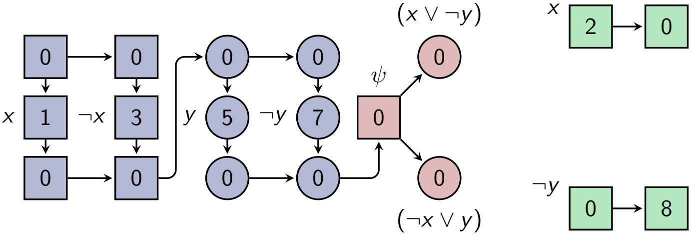
The Reduction

$$\varphi = \forall x \ \exists y \ . \ \overbrace{(x \lor \neg y) \land (\neg x \lor y)}^{\psi}$$



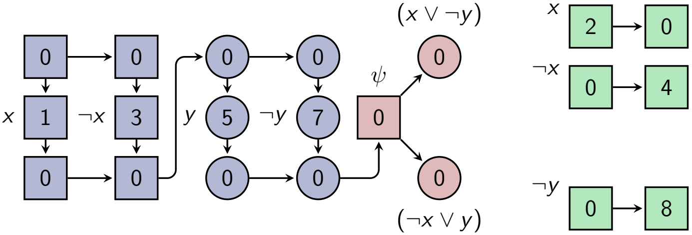
The Reduction

$$\varphi = \forall x \ \exists y \ . \ \overbrace{(\, x \ \vee \ \neg y \,) \wedge (\, \neg x \ \vee \ y \,)}^{\psi}$$



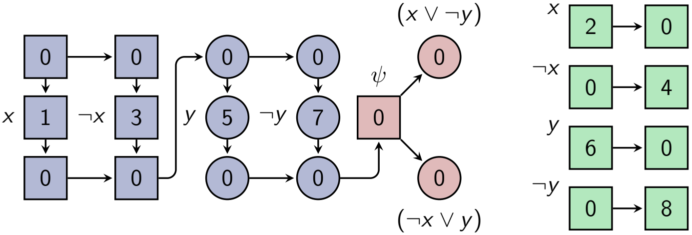
The Reduction

$$\varphi = \forall x \; \exists y \, . \; \overbrace{(\, x \vee \neg y \,) \wedge (\, \neg x \vee y \,)}^{\psi}$$



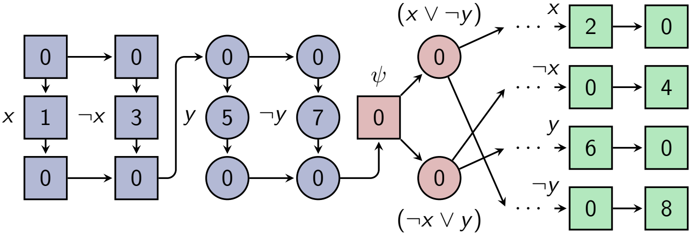
The Reduction

$$\varphi = \forall x \ \exists y \ . \ \overbrace{(\ x \ \vee \ \neg y\ ) \wedge (\ \neg x \ \vee \ y\ )}^{\psi}$$



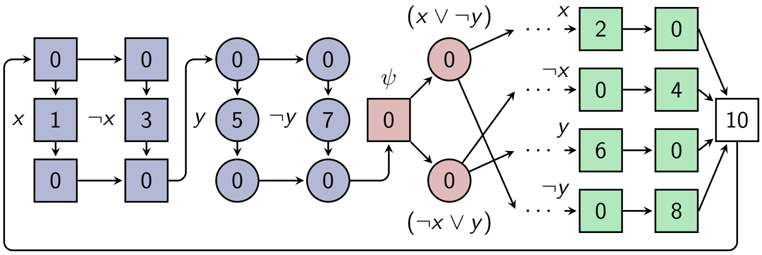
The Reduction

$$\varphi = \forall x\ \exists y\ .\ \overbrace{(\ x\ \vee\ \neg y\ ) \wedge (\ \neg x\ \vee\ y\ )}^{\psi}$$



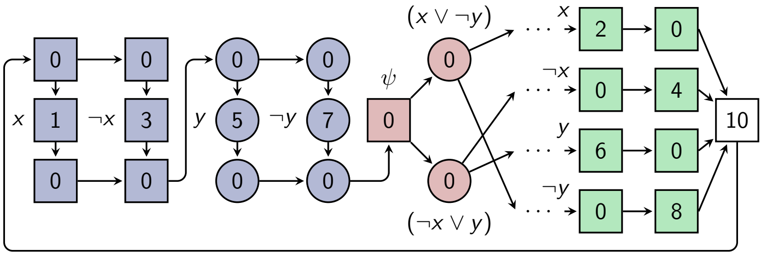
The Reduction

$$\varphi = \forall x \ \exists y \ . \ \overbrace{(\ x \ \vee \ \neg y\ ) \wedge (\ \neg x \ \vee \ y\ )}^{\psi}$$



The Reduction

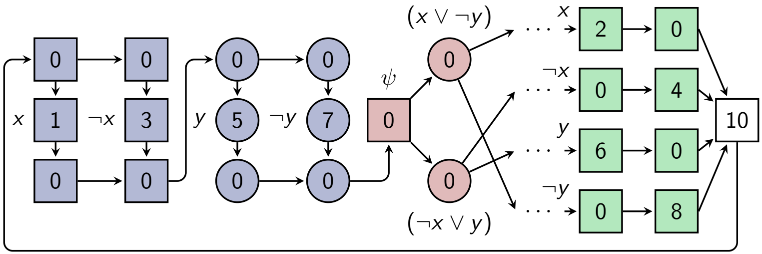
$$\varphi = \forall x \ \exists y \ . \ \overbrace{(\ x \ \vee \ \neg y\ ) \wedge (\ \neg x \ \vee \ y\ )}^{\psi}$$



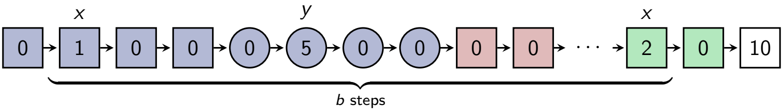
For a well-chosen bound b, a strategy for Player 0 with cost at most b witnesses the truth of φ and vice versa.

The Reduction

$$\varphi = \forall x \ \exists y \ . \ \overbrace{(\ x \ \vee \ \neg y\ ) \wedge (\ \neg x \ \vee \ y\ )}^{\psi}$$

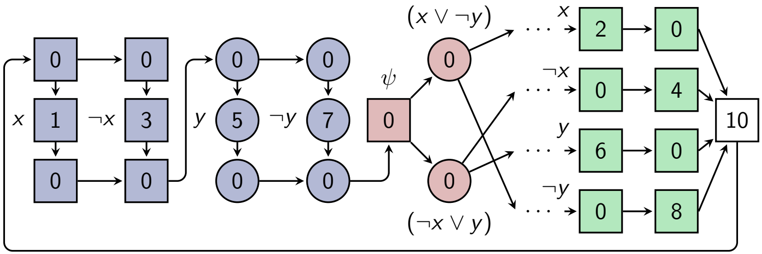


For a well-chosen bound b, a strategy for Player 0 with cost at most b witnesses the truth of φ and vice versa.

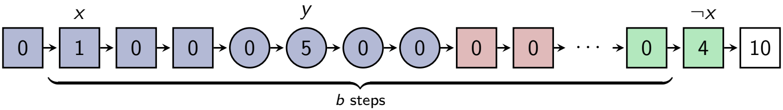


The Reduction

$$\varphi = \forall x\ \exists y\ .\ \overbrace{(\,x \vee \neg y\,) \wedge (\,\neg x \vee y\,)}^{\psi}$$

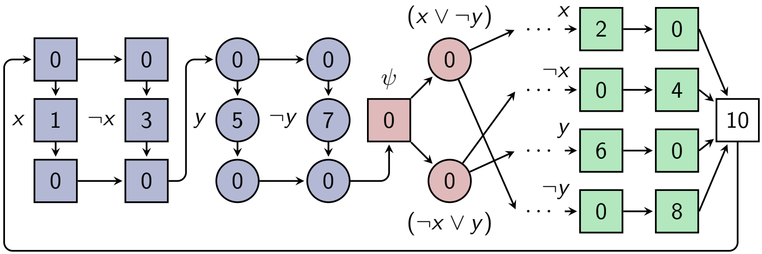


For a well-chosen bound b, a strategy for Player 0 with cost at most b witnesses the truth of φ and vice versa.

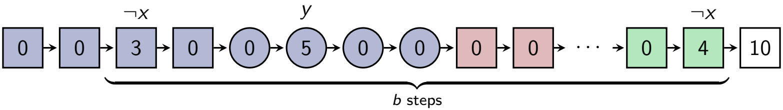


The Reduction

$$\varphi = \forall x\ \exists y\ .\ \overbrace{(x \vee \neg y) \wedge (\neg x \vee y)}^{\psi}$$



For a well-chosen bound b, a strategy for Player 0 with cost at most b witnesses the truth of φ and vice versa.



PSPACE-Membership

Lemma

The following problem is in $\textsc{PSpace}$*: "Given a finitary parity game* $\mathcal{G}$ *and a bound* $b \in \mathbb{N}$*, does Player* 0 *have a strategy for* $\mathcal{G}$ *whose cost is at most* b*?"*

PSPACE-Membership

Lemma

The following problem is in $\textsc{PSpace}$*: "Given a finitary parity game* $\mathcal{G}$ *and a bound* $b \in \mathbb{N}$*, does Player* 0 *have a strategy for* $\mathcal{G}$ *whose cost is at most* b*?"*

Proof Sketch

Fix $\mathcal{G}$ and b (w.l.o.g. $b \leq |\mathcal{G}|$).

1. Construct equivalent parity game $\mathcal{G}'$ storing the costs of open requests (up to bound b) and the number of overflows (up to bound $|\mathcal{G}|$) $\Rightarrow$ $|\mathcal{G}'| \in |\mathcal{G}|^{\mathcal{O}(d)}$.

PSPACE-Membership

Lemma

The following problem is in $\textsc{PSpace}$*: "Given a finitary parity game* $\mathcal{G}$ *and a bound* $b \in \mathbb{N}$*, does Player* 0 *have a strategy for* $\mathcal{G}$ *whose cost is at most* b*?"*

Proof Sketch

Fix $\mathcal{G}$ and b (w.l.o.g. $b \leq |\mathcal{G}|$).

1. Construct equivalent parity game $\mathcal{G}'$ storing the costs of open requests (up to bound b) and the number of overflows (up to bound $|\mathcal{G}|$) $\Rightarrow |\mathcal{G}'| \in |\mathcal{G}|^{\mathcal{O}(d)}$.
2. Define equivalent finite-duration variant $\mathcal{G}'_f$ of $\mathcal{G}'$ with polynomial play-length.
3. $\mathcal{G}'_f$ can be solved on alternating polynomial-time Turing machine.
4. $\textsc{APTime} = \textsc{PSpace}$ concludes the proof.

Upper Bounds on Memory

Equivalence between finitary parity game $\mathcal{G}$ w.r.t. bound b and parity game $\mathcal{G}'$ yields upper bounds on memory requirements.

Corollary

Let $\mathcal{G}$ be a finitary parity game with costs with d odd colors. If Player 0 has a strategy for $\mathcal{G}$ with cost b, then she also has a strategy with cost b and size $(b+2)^d = 2^{d \log(b+2)}$.

Upper Bounds on Memory

Equivalence between finitary parity game $\mathcal{G}$ w.r.t. bound b and parity game $\mathcal{G}'$ yields upper bounds on memory requirements.

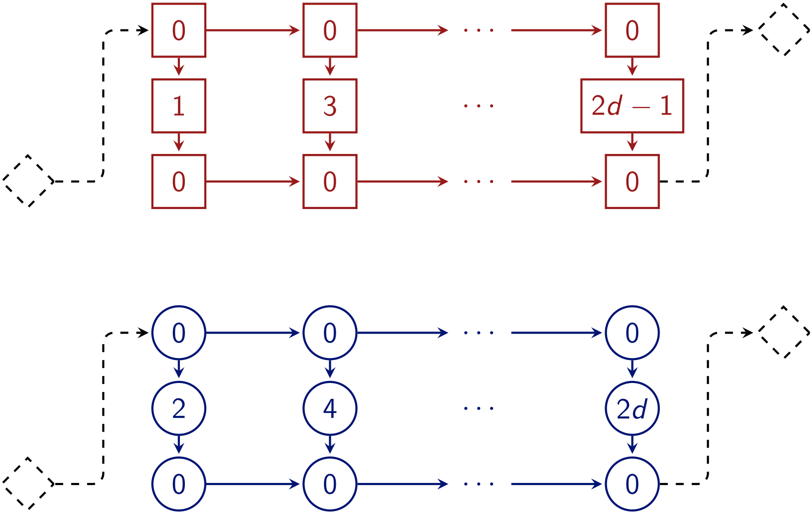
Corollary

Let $\mathcal{G}$ be a finitary parity game with costs with d odd colors. If Player 0 has a strategy for $\mathcal{G}$ with cost b, then she also has a strategy with cost b and size $(b+2)^d = 2^{d\log(b+2)}$.

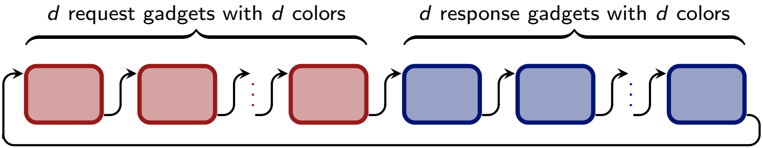
- Recall: lower bound 2^d.
- The same bounds hold for Player 1.

Outline

Tradeoffs



Tradeoffs

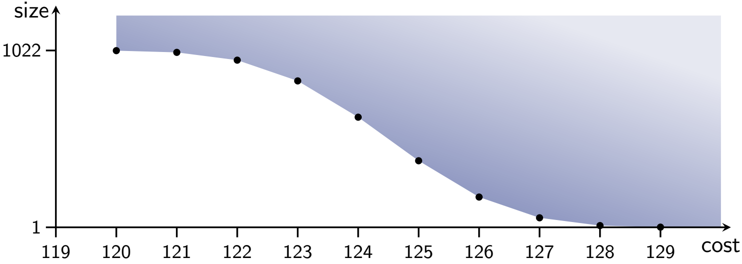


- Recall: Player 0 has winning strategy with cost $d^2 + 2d$: answer j-th unique request in j-th response-gadget, which requires memory of size $2^d - 2$.
- Only store first i unique requests, then go to largest answer in next gadget.
  $\Rightarrow$ achieves cost $d^2 + 3d - i$ and size $\sum_{j=1}^{i-1} \binom{d}{j}$.
- Against a smaller strategy Player 1 can enforce a larger cost, as Player 0 cannot store every sequence of i requests.

Tradeoffs

Theorem

Fix some finitary parity game $\mathcal{G}_d$ as before. For every i with $1 \leq i \leq d$ there exists a strategy σ_i for Player 0 in $\mathcal{G}_d$ such that σ_i has cost $d^2 + 3d - i$ and size $\sum_{j=1}^{i-1} \binom{d}{j}$.
Also, every strategy σ' for Player 0 in $\mathcal{G}_d$ whose cost is at most the cost of σ_i has at least the size of σ_i.



Outline

1. Memory Requirements of Optimal Strategies
2. Determining Optimal Bounds is Hard
3. Trading Memory for Quality and Vice Versa
4. **Generalizations**
5. Conclusion

Generalizations 1: Cost

Parity Games with costs

- In a finitary parity game, every edge has unit cost.
- In parity games with costs, allow arbitrary weights from $\mathbb{N}$.
- Subsumes parity games (cost zero for every edge) and finitary parity games (cost one for every edge) as special cases.

Generalizations 1: Cost

Parity Games with costs

- In a finitary parity game, every edge has unit cost.
- In parity games with costs, allow arbitrary weights from $\mathbb{N}$.
- Subsumes parity games (cost zero for every edge) and finitary parity games (cost one for every edge) as special cases.

New challenges:

- Arbitrarily long infixes of cost zero have to be dealt with.
  $\Rightarrow$ Use techniques for parity games.
- A binary encoding of the weights only allows an exponential upper bound on the cost of an optimal strategy.
  $\Rightarrow$ Adapt finite-duration game $\mathcal{G}'_f$ accordingly.

Generalizations 2: Streett

Streett Games

- In parity games, requests and responses are hierarchical.
- In Streett games, use a finite collection $(Q_j, P_j)_j$ of sets of vertices, requests Q_j and responses P_j of condition j.

Generalizations 2: Streett

Streett Games

- In parity games, requests and responses are hierarchical.
- In Streett games, use a finite collection $(Q_j, P_j)_j$ of sets of vertices, requests Q_j and responses P_j of condition j.

Finitary Streett Games / Streett Games with Costs

- Streett condition and weights from $\{1\}$ / $\mathbb{N}$.

Generalizations 2: Streett

Streett Games

- In parity games, requests and responses are hierarchical.
- In Streett games, use a finite collection $(Q_j, P_j)_j$ of sets of vertices, requests Q_j and responses P_j of condition j.

Finitary Streett Games / Streett Games with Costs

- Streett condition and weights from $\{1\}$ / $\mathbb{N}$.

New relief:

- Finitary Streett games are already ExpTime-complete and exponential memory is necessary
  $\Rightarrow$ Appropriate adaption of $\mathcal{G}'$ can be solved straightaway in exponential time, yielding exponential upper bounds on memory

More Results

Condition	Complexity	Mem. Pl. 0	Mem. Pl. 1
Parity	$\mathrm{UP} \cap \textsc{co}\text{-}\mathrm{UP}$	Memoryless	Memoryless
Finitary Parity	$\mathrm{P}\textsc{Time}$	Memoryless	Infinite

More Results

Condition	Complexity	Mem. Pl. 0	Mem. Pl. 1
Parity	UP ∩ co-UP	Memoryless	Memoryless
Finitary Parity	PTime	Memoryless	Infinite
Parity with Cost	UP ∩ co-UP	Memoryless	Infinite
Streett	co-NP-complete	Exponential	Memoryless
Finitary Streett	ExpTime-compl.	Exponential	Infinite
Streett with Cost	ExpTime-compl.	Exponential	Infinite

More Results

Condition	Complexity	Mem. Pl. 0	Mem. Pl. 1
Parity	UP ∩ co-UP	Memoryless	Memoryless
Finitary Parity	PTime	Memoryless	Infinite
Parity with Cost	UP ∩ co-UP	Memoryless	Infinite
Streett	co-NP-complete	Exponential	Memoryless
Finitary Streett	ExpTime-compl.	Exponential	Infinite
Streett with Cost	ExpTime-compl.	Exponential	Infinite
Opt. Finitary Parity	PSpace-compl.	Exponential	Exponential
Opt. Parity with Cost*	PSpace-compl.	Exponential	Exponential
Opt. Finitary Streett	ExpTime-compl.	Exponential	Exponential
Opt. Streett with Cost*	ExpTime-compl.	Exponential	Exponential

* Holds for binary encoding of the weights.

Outline

1. Memory Requirements of Optimal Strategies
2. Determining Optimal Bounds is Hard
3. Trading Memory for Quality and Vice Versa
4. Generalizations
5. **Conclusion**

Conclusion

Results

- Playing finitary games/games with costs optimally is harder than just winning them.
- Both in terms of memory requirements and computational complexity.
- Quality can (gradually) be traded for memory and vice versa.

Conclusion

Results

- Playing finitary games/games with costs optimally is harder than just winning them.
- Both in terms of memory requirements and computational complexity.
- Quality can (gradually) be traded for memory and vice versa.

Open problems

- Parity games with mutiple cost functions
- Multi-dimensional games
- Tradeoffs in other games (first results for parametric LTL and energy games)