Games with Costs and Delays

Martin Zimmermann

Saarland University

June 20th, 2017

LICS 2017, Reykjavik, Iceland

Gale-Stewart Games

Büchi-Landweber: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

Gale-Stewart Games

Büchi-Landweber: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

Gale-Stewart Games

Büchi-Landweber: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I: b

O:

Gale-Stewart Games

Büchi-Landweber: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

***I*:**	b
***O*:**	a

Gale-Stewart Games

Büchi-Landweber: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	*b*	*a*
O:	*a*	

Gale-Stewart Games

Büchi-Landweber: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L \text{, if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a
O:	a	a

Gale-Stewart Games

Büchi-Landweber: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	*b*	*a*	*b*
O:	*a*	*a*	

Gale-Stewart Games

Büchi-Landweber: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

***I*:**	b	a	b	$\cdots$
***O*:**	a	a	$\cdots$	

I wins!

Gale-Stewart Games

Büchi-Landweber: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	$\cdots$
O:	a	a	$\cdots$	

I wins!

- Many possible extensions... we consider two:
  Interaction: one player may delay her moves.
  Winning condition: quantitative instead of qualitative.

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L \text{, if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I: b
O:

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I: b a
O:

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I: b a b
O:

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b
O:	b		

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b
O:	b			

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)} \cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b
O:	b	b		

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a
O:	b	b			

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a
O:	b	b	a		

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a	a
O:	b	b	a			

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a	a
O:	b	b	a	a		

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a	a	b
O:	b	b	a	a			

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a	a	b
O:	b	b	a	a	b		

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a	a	b	b
O:	b	b	a	a	b			

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a	a	b	b
O:	b	b	a	a	b	b		

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a	a	b	b	$\cdots$
O:	b	b	a	a	b	b	$\cdots$		

O wins!

Delay Games

Allow Player O to delay her moves:

$$\binom{\alpha(0)}{\beta(0)}\binom{\alpha(1)}{\beta(1)}\cdots \in L, \text{ if } \beta(i) = \alpha(i+2) \text{ for every } i$$

I:	b	a	b	b	a	a	b	b	$\cdots$
O:	b	b	a	a	b	b	$\cdots$		

O wins!

Typical questions:

- How often does Player O have to delay to win?
- How hard is determining the winner of a delay game?
- Does the ability to delay allow Player O to improve the quality of her strategies?

Previous Work

If winning conditions given by deterministic parity automata:

Theorem (Klein, Z. '15)

- *If Player O wins delay game induced by $\mathcal{A}$, then also by delaying at most $2^{|\mathcal{A}|^2}$ times.*

Previous Work

If winning conditions given by deterministic parity automata:

Theorem (Klein, Z. '15)

- *If Player O wins delay game induced by $\mathcal{A}$, then also by delaying at most $2^{|\mathcal{A}|^2}$ times.*
- *Lower bound $2^{|\mathcal{A}|}$ (already for safety automata).*

Previous Work

If winning conditions given by deterministic parity automata:

Theorem (Klein, Z. '15)

- *If Player O wins delay game induced by $\mathcal{A}$, then also by delaying at most $2^{|\mathcal{A}|^2}$ times.*
- *Lower bound $2^{|\mathcal{A}|}$ (already for safety automata).*
- *Determining the winner is* **EXPTIME***-complete (hardness already for safety automata).*

Previous Work

If winning conditions given by deterministic parity automata:

Theorem (Klein, Z. '15)

- *If Player O wins delay game induced by $\mathcal{A}$, then also by delaying at most $2^{|\mathcal{A}|^2}$ times.*
- *Lower bound $2^{|\mathcal{A}|}$ (already for safety automata).*
- *Determining the winner is* **EXPTIME***-complete (hardness already for safety automata).*

Note:
This improved similar results by Holtmann, Kaiser, and Thomas with doubly-exponential upper bounds and no lower bounds.

Previous Work

If winning conditions given by formula in (quantitative) linear temporal logics:

Theorem (Klein, Z. '16)

- *If Player O wins delay game induced by φ, then also by delaying at most $2^{2^{2^{|\varphi|}}}$ times.*
- *There is a matching lower bound.*
- *Determining the winner is* **3EXPTIME***-complete.*

Note:
Quantitative conditions not harder than qualitative ones.

Uniformization of Relations

- A strategy σ for O in a game induces a mapping $f_\sigma \colon \Sigma_I^\omega \to \Sigma_O^\omega$
- σ is winning $\Leftrightarrow \{\binom{\alpha}{f_\sigma(\alpha)} \mid \alpha \in \Sigma_I^\omega\} \subseteq L$ (f_σ uniformizes L)

Uniformization of Relations

- A strategy σ for O in a game induces a mapping $f_\sigma \colon \Sigma_I^\omega \to \Sigma_O^\omega$
- σ is winning $\Leftrightarrow \{ \binom{\alpha}{f_\sigma(\alpha)} \mid \alpha \in \Sigma_I^\omega \} \subseteq L$ (f_σ uniformizes L)

Continuity in terms of strategies (in Cantor metric):

- Strategy without lookahead: i-th letter of $f_\sigma(\alpha)$ only depends on first i letters of α (very strong notion of continuity).

Uniformization of Relations

- A strategy σ for O in a game induces a mapping $f_\sigma \colon \Sigma_I^\omega \to \Sigma_O^\omega$
- σ is winning $\Leftrightarrow \{\binom{\alpha}{f_\sigma(\alpha)} \mid \alpha \in \Sigma_I^\omega\} \subseteq L$ (f_σ uniformizes L)

Continuity in terms of strategies (in Cantor metric):

- Strategy without lookahead: i-th letter of $f_\sigma(\alpha)$ only depends on first i letters of α (very strong notion of continuity).
- Strategy with bounded delay: f_σ Lipschitz-continuous.

Uniformization of Relations

- A strategy σ for O in a game induces a mapping $f_\sigma \colon \Sigma_I^\omega \to \Sigma_O^\omega$
- σ is winning $\Leftrightarrow \{\binom{\alpha}{f_\sigma(\alpha)} \mid \alpha \in \Sigma_I^\omega\} \subseteq L$ (f_σ uniformizes L)

Continuity in terms of strategies (in Cantor metric):

- Strategy without lookahead: i-th letter of $f_\sigma(\alpha)$ only depends on first i letters of α (very strong notion of continuity).
- Strategy with bounded delay: f_σ Lipschitz-continuous.
- Strategy with arbitrary (finite) delay: f_σ (uniformly) continuous.

Uniformization of Relations

- A strategy σ for O in a game induces a mapping $f_\sigma \colon \Sigma_I^\omega \to \Sigma_O^\omega$
- σ is winning $\Leftrightarrow \{ \binom{\alpha}{f_\sigma(\alpha)} \mid \alpha \in \Sigma_I^\omega \} \subseteq L$ (f_σ uniformizes L)

Continuity in terms of strategies (in Cantor metric):

- Strategy without lookahead: i-th letter of $f_\sigma(\alpha)$ only depends on first i letters of α (very strong notion of continuity).
- Strategy with bounded delay: f_σ Lipschitz-continuous.
- Strategy with arbitrary (finite) delay: f_σ (uniformly) continuous.

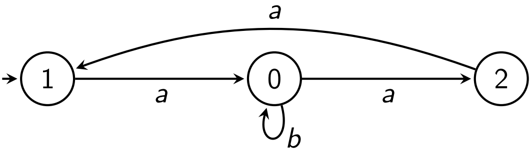
Holtmann, Kaiser, Thomas: for ω-regular L

L uniformizable by continuous function

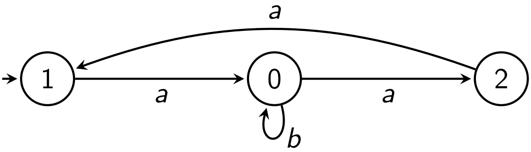
$\Leftrightarrow$

L uniformizable by Lipschitz-continuous function

Finitary Parity Automata



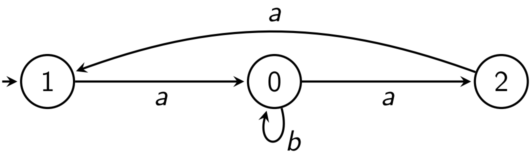
Finitary Parity Automata



Parity acceptance: Almost every odd priority is followed by a larger even one.

$$L(\mathcal{A}) = a(b^*aaa)^*b^\omega + a(b^*aaa)^\omega$$

Finitary Parity Automata



Parity acceptance: Almost every odd priority is followed by a larger even one.

$$L(\mathcal{A}) = a(b^* aaa)^* b^\omega + a(b^* aaa)^\omega$$

Finitary parity acceptance: There is a bound n such that almost every odd priority is followed by a larger even one within n steps.

$$L(\mathcal{A}) = a(b^* aaa)^* b^\omega + \sum_{n \in \mathbb{N}} a(b^{\leq n} aaa)^\omega$$

Finitary Parity Automata

Remark

Safety automata can be transformed into finitary parity automata of the same size.

Finitary Parity Automata

Remark

Safety automata can be transformed into finitary parity automata of the same size.

Proof:

Turn all unsafe states into sinks with an odd color, all safe states get even color.

Finitary Parity Automata

Remark

Safety automata can be transformed into finitary parity automata of the same size.

Proof:

Turn all unsafe states into sinks with an odd color, all safe states get even color.

Thus: exponential lower bounds on complexity and necessary lookahead for delay games with finitary parity conditions.

Results

If winning conditions given by deterministic finitary parity automata:

Theorem

- *If Player O wins delay game induced by $\mathcal{A}$, then also by delaying at most $2^{|\mathcal{A}|^6}$ times.*

Results

If winning conditions given by deterministic finitary parity automata:

Theorem

- *If Player O wins delay game induced by $\mathcal{A}$, then also by delaying at most $2^{|\mathcal{A}|^6}$ times.*
- *Lower bound $2^{|\mathcal{A}|}$.*

Results

If winning conditions given by deterministic finitary parity automata:

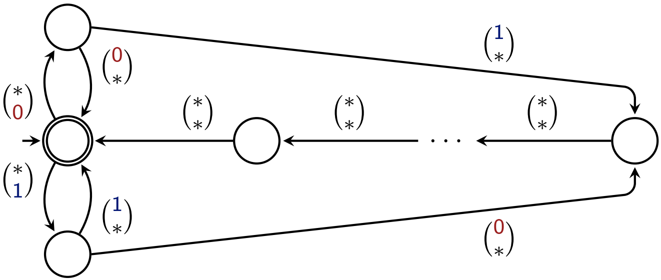
Theorem

- *If Player O wins delay game induced by $\mathcal{A}$, then also by delaying at most $2^{|\mathcal{A}|^6}$ times.*
- *Lower bound $2^{|\mathcal{A}|}$.*
- *Determining the winner is* **EXPTIME***-complete.*

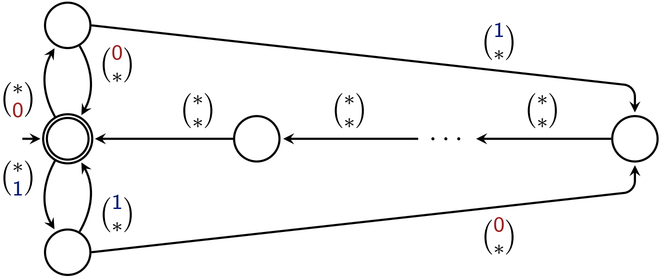
Note:

Again, quantitative conditions not harder than qualitative ones.

Tradeoff Lookahead vs. Quality



Tradeoff Lookahead vs. Quality



Theorem

For every $n > 0$, there is a language L_n recognized by a finitary Büchi automaton with $n + 2$ states such that

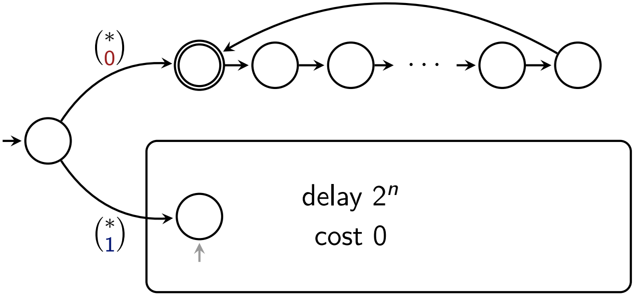
- *an optimal strategy without delay has cost n, but*
- *an optimal strategy delaying once has cost 1.*

Tradeoff Lookahead vs. Quality

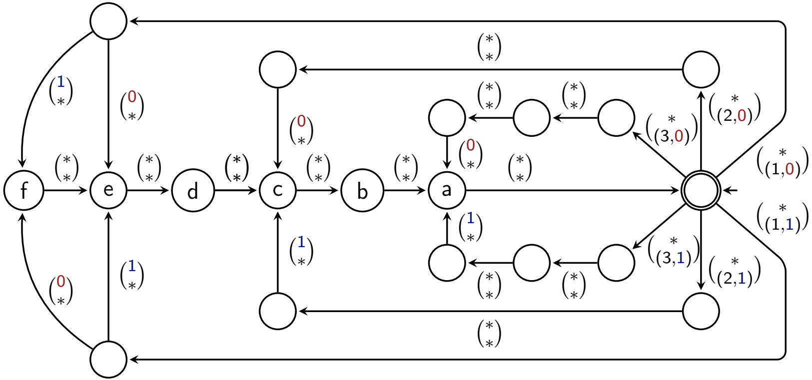
Theorem

For every $n > 0$, there is a language L'_n recognized by a finitary Büchi automaton with $\mathcal{O}(n)$ states such that

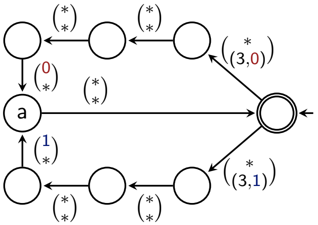
- *an optimal strategy delaying 2^n times has cost 0, and*
- *an optimal strategy delaying less than 2^n times has cost n.*



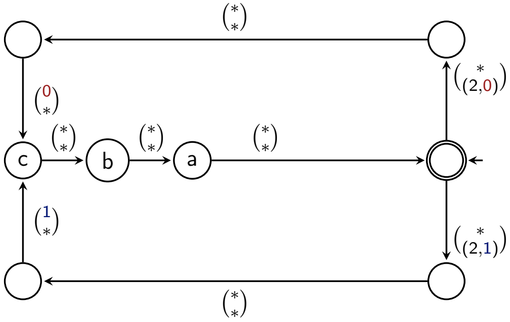
Tradeoff Lookahead vs. Quality



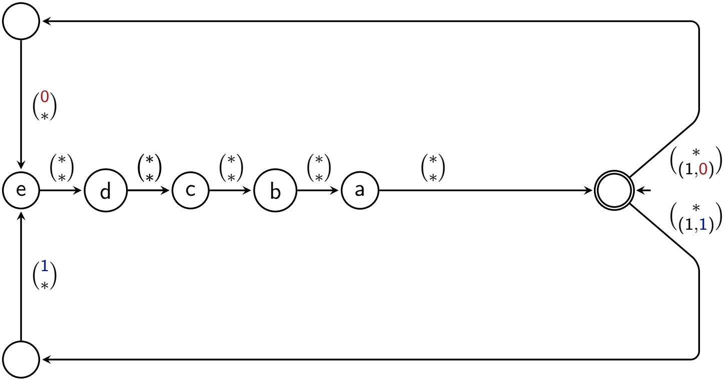
Tradeoff Lookahead vs. Quality



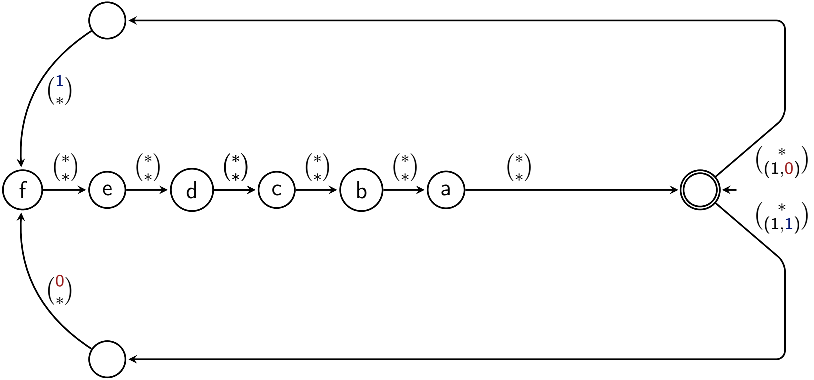
Tradeoff Lookahead vs. Quality



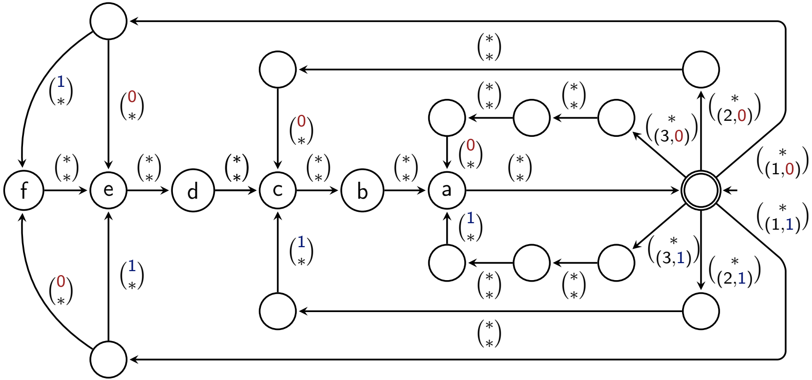
Tradeoff Lookahead vs. Quality



Tradeoff Lookahead vs. Quality



Tradeoff Lookahead vs. Quality



Theorem

For every $n > 0$, there is a language L''_n recognized by a finitary Büchi automaton with $\mathcal{O}(n^2)$ states such that for every $0 \leq j \leq n$: an optimal strategy delaying j times has cost $2(n+1) - j$.

More Results

acceptance	lookahead	complexity

More Results

acceptance	lookahead	complexity
parity	exp.	**EXPTIME**-complete
finitary parity	exp.	**EXPTIME**-complete

More Results

acceptance	lookahead	complexity
parity	exp.	**EXPTIME**-complete
finitary parity	exp.	**EXPTIME**-complete
parity w. costs	exp.	**EXPTIME**-complete

More Results

acceptance	lookahead	complexity
parity	exp.	**EXPTIME**-complete
finitary parity	exp.	**EXPTIME**-complete
parity w. costs	exp.	**EXPTIME**-complete
finitary Streett	exp./doubly-exp.	**EXPTIME**/**2EXPTIME**
Streett w. costs	exp./doubly-exp.	**EXPTIME**/**2EXPTIME**

More Results

acceptance	lookahead	complexity
parity	exp.	**EXPTIME**-complete
finitary parity	exp.	**EXPTIME**-complete
parity w. costs	exp.	**EXPTIME**-complete
finitary Streett	exp./doubly-exp.	**EXPTIME/2EXPTIME**
Streett w. costs	exp./doubly-exp.	**EXPTIME/2EXPTIME**

Theorem

Optimal strategies in delay games with Streett conditions with costs may require doubly-exponential lookahead.

Conclusion

- Quantitative delay games with parity conditions are not harder than qualitative ones.
- Lookahead allows to improve the quality of strategies.

Conclusion

- Quantitative delay games with parity conditions are not harder than qualitative ones.
- Lookahead allows to improve the quality of strategies.

Open Problems

- Close the gaps for Streett conditions (qualitative and quantitative).
- Study other tradeoffs, e.g., lookahead vs. memory size.
- Determine the complexity of finding optimal strategies (smallest cost or smallest lookahead).