Delay Games

Martin Zimmermann

Saarland University

March 15th, 2018

Università degli studi di Napoli "Federico II", Naples, Italy

Motivation

Büchi & Landweber '69: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

Motivation

Büchi & Landweber '69: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

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

Motivation

Büchi & Landweber '69: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

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

I:
O:

Motivation

Büchi & Landweber '69: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

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

I: b
O:

Motivation

Büchi & Landweber '69: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

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

I:	b
O:	a

Motivation

Büchi & Landweber '69: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

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

I:	b	a
O:	a	

Motivation

Büchi & Landweber '69: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

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

I:	b	a
O:	a	a

Motivation

Büchi & Landweber '69: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

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

I:	b	a	b
O:	a	a	

Motivation

Büchi & Landweber '69: The winner of a zero-sum two-player game of infinite duration with ω-regular winning condition can be determined effectively.

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

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

I wins

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

I:

O:

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

I: b
O:

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

I: b a
O:

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

I: b a b
O:

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

I:	b	a	b
O:	b		

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

I:	b	a	b	b
O:	b			

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

I:	b	a	b	b
O:	b	b		

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

I:	b	a	b	b	a
O:	b	b			

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

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

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

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

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

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

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

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

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

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

Motivation

Hosch & Landweber '72: Let O delay her moves to obtain a lookahead on I's moves.

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

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

O **wins**

Outline

1. **Introduction**
2. **Lower Bounds on the Necessary Lookahead**
3. **Solving Delay Games**
4. **Conclusion**

Outline

1. **Introduction**
2. Lower Bounds on the Necessary Lookahead
3. Solving Delay Games
4. Conclusion

A Formal Definition

A delay game $\Gamma_f(L)$ consists of

- a delay function $f\colon \mathbb{N} \to \mathbb{N}_+$ and
- an ω-language $L \subseteq (\Sigma_I \times \Sigma_O)^\omega$.

A Formal Definition

A delay game $\Gamma_f(L)$ consists of

- a delay function $f\colon \mathbb{N} \to \mathbb{N}_+$ and
- an ω-language $L \subseteq (\Sigma_I \times \Sigma_O)^\omega$.

It is contested by players "Input" (I) and "Output" (O) as follows:

- In each round $i = 0, 1, 2, \ldots$
  - first, I picks word $u_i \in \Sigma_I^{f(i)}$,
  - then, O picks letter $v_i \in \Sigma_O$.
- O wins iff $\binom{u_0 u_1 u_2 \cdots}{v_0 v_1 v_2 \cdots} \in L$.

A Formal Definition

Questions we are interested in:

- Given L, is there an f such that O wins $\Gamma_f(L)$?
- How hard is the problem to solve?
- How *large* does f have to be?

A Formal Definition

Questions we are interested in:

- Given L, is there an f such that O wins $\Gamma_f(L)$?
- How hard is the problem to solve?
- How *large* does f have to be?

Definition A delay function f is constant, if $f(i) = 1$ for every $i > 0$.

Intuition: W.r.t. constant f, O has lookahead of size $f(0)$ in each round.

History

- **Hosch & Landweber '72:** ω-regular delay games restricted to constant delay functions decidable.

History

- **Hosch & Landweber '72:** ω-regular delay games restricted to constant delay functions decidable.
- **Kaiser, Holtmann & Thomas '10:** ω-regular delay games decidable in $2\textsc{ExpTime}$, doubly-exponential constant lookahead sufficient.

History

- **Hosch & Landweber '72:** ω-regular delay games restricted to constant delay functions decidable.
- **Kaiser, Holtmann & Thomas '10:** ω-regular delay games decidable in $2\textsc{ExpTime}$, doubly-exponential constant lookahead sufficient.
- **Fridman, Löding & Z. '11:** (very weak subclasses of) ω-contextfree delay games are undecidable w.r.t. constant delay functions, fast-growing unbounded lookahead necessary.

History

- **Hosch & Landweber '72:** ω-regular delay games restricted to constant delay functions decidable.
- **Kaiser, Holtmann & Thomas '10:** ω-regular delay games decidable in $2\textsc{ExpTime}$, doubly-exponential constant lookahead sufficient.
- **Fridman, Löding & Z. '11:** (very weak subclasses of) ω-contextfree delay games are undecidable w.r.t. constant delay functions, fast-growing unbounded lookahead necessary.
- **Klein & Z. '15:** ω-regular delay games $\textsc{ExpTime}$-complete, exponential constant lookahead sufficient and necessary.

History

- **Hosch & Landweber '72:** ω-regular delay games restricted to constant delay functions decidable.
- **Kaiser, Holtmann & Thomas '10:** ω-regular delay games decidable in $2\textsc{ExpTime}$, doubly-exponential constant lookahead sufficient.
- **Fridman, Löding & Z. '11:** (very weak subclasses of) ω-contextfree delay games are undecidable w.r.t. constant delay functions, fast-growing unbounded lookahead necessary.
- **Klein & Z. '15:** ω-regular delay games $\textsc{ExpTime}$-complete, exponential constant lookahead sufficient and necessary.
- **Z. '15:** max-regular delay games restricted to constant delay functions decidable, in general unbounded lookahead necessary, but no lower bound on growth rate. If constant lookahead suffices, then doubly-exponential one is sufficient.

More History

- **Klein & Z. '15:** Borel determinacy for delay games.

More History

- **Klein & Z. '15:** Borel determinacy for delay games.
- **Klein & Z. '16:** LTL delay games 3ExpTime-complete, triply-exponential constant lookahead sufficient and necessary. Same holds for parameterized variants of LTL, e.g., Prompt-LTL.

More History

- **Klein & Z. '15:** Borel determinacy for delay games.
- **Klein & Z. '16:** LTL delay games 3ExpTime-complete, triply-exponential constant lookahead sufficient and necessary. Same holds for parameterized variants of LTL, e.g., Prompt-LTL.
- **Z. '17:** Delay games over parity automata with costs ExpTime-complete, exponential constant lookahead sufficient and necessary. Tradeoffs between lookahead and quality of strategies.

More History

- **Klein & Z. '15:** Borel determinacy for delay games.
- **Klein & Z. '16:** LTL delay games 3EXPTIME-complete, triply-exponential constant lookahead sufficient and necessary. Same holds for parameterized variants of LTL, e.g., Prompt-LTL.
- **Z. '17:** Delay games over parity automata with costs EXPTIME-complete, exponential constant lookahead sufficient and necessary. Tradeoffs between lookahead and quality of strategies.
- **Z. '17:** A general framework to solve delay games and compute finite-state strategies for them.

More History

- **Klein & Z. '15:** Borel determinacy for delay games.
- **Klein & Z. '16:** LTL delay games 3ExpTime-complete, triply-exponential constant lookahead sufficient and necessary. Same holds for parameterized variants of LTL, e.g., Prompt-LTL.
- **Z. '17:** Delay games over parity automata with costs ExpTime-complete, exponential constant lookahead sufficient and necessary. Tradeoffs between lookahead and quality of strategies.
- **Z. '17:** A general framework to solve delay games and compute finite-state strategies for them.
- **Winter & Z.:** Tradeoffs between lookahead and memory size.

Uniformization of Relations

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

Uniformization of Relations

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

Continuity in terms of strategies:

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

Uniformization of Relations

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

Continuity in terms of strategies:

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

Uniformization of Relations

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

Continuity in terms of strategies:

- Strategy without delay: i-th letter of $g_\sigma(\alpha)$ only depends on first i letters of α (very strong notion of continuity).
- Strategy with constant delay: g_σ Lipschitz-continuous.
- Strategy with arbitrary (finite) delay: g_σ continuous.

Uniformization of Relations

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

Continuity in terms of strategies:

- Strategy without delay: i-th letter of $g_\sigma(\alpha)$ only depends on first i letters of α (very strong notion of continuity).
- Strategy with constant delay: g_σ Lipschitz-continuous.
- Strategy with arbitrary (finite) delay: g_σ continuous.

Holtmann, Kaiser, Thomas: for ω-regular L

L uniformizable by continuous function

$\Leftrightarrow$

L uniformizable by Lipschitz-continuous function

Outline

1. Introduction

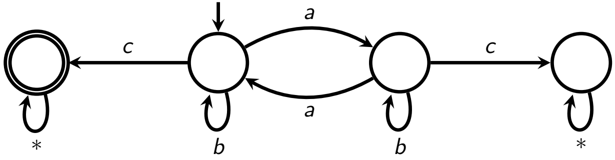
2. **Lower Bounds on the Necessary Lookahead**

3. Solving Delay Games

4. Conclusion

Lower Bounds on Lookahead

A reachability automaton accepts if an accepting state is reached at least once.



$$L(\mathcal{A}) = \{\alpha \in \{a, b, c\}^\omega \mid$$
α contains a c and has an even number of a's before the first one$\}$

Lower Bounds on Lookahead

A reachability automaton accepts if an accepting state is reached at least once.

Theorem

For every $n > 1$ there is a language L_n recognized by a deterministic reachability automaton $\mathcal{A}_n$ with $|\mathcal{A}_n| \in \mathcal{O}(n)$ s.t.

- *O wins $\Gamma_f(L_n)$ for some constant delay function f, but*
- *I wins $\Gamma_f(L_n)$ for every delay function f with $f(0) \leq 2^n$.*

Proof

- Fix $\Sigma_I = \Sigma_O = \{1, \ldots, n\}$.
- $w \in \Sigma_I^*$ contains bad j-pair ($j \in \Sigma_I$) if there are two occurrences of j in w such that no $j' > j$ occurs in between.

Proof

- Fix $\Sigma_I = \Sigma_O = \{1, \ldots, n\}$.
- $w \in \Sigma_I^*$ contains bad j-pair ($j \in \Sigma_I$) if there are two occurrences of j in w such that no $j' > j$ occurs in between.

Claim: $w \in \Sigma_I^{\geq 2^n} \Rightarrow w$ has a bad j-pair for some j.

Proof

- Fix $\Sigma_I = \Sigma_O = \{1, \ldots, n\}$.
- $w \in \Sigma_I^*$ contains bad j-pair ($j \in \Sigma_I$) if there are two occurrences of j in w such that no $j' > j$ occurs in between.

Claim: $w \in \Sigma_I^{\geq 2^n} \Rightarrow w$ has a bad j-pair for some j.

Proof by induction over n:

$n = 1$**:** $w = 1^k$ for $k \geq 2^1$ has bad 1-pair.

$n > 1$**:** Consider two cases:

- If w has more than one letter n, then it contains bad n-pair.
- Otherwise, w has infix $w' \in \{1, \ldots, n-1\}^{\geq 2^{n-1}}$. Then, w' has bad j-pair for some $j < n$ by induction hypothesis.

Proof

- Fix $\Sigma_I = \Sigma_O = \{1, \ldots, n\}$.
- $w \in \Sigma_I^*$ contains bad j-pair ($j \in \Sigma_I$) if there are two occurrences of j in w such that no $j' > j$ occurs in between.

Claim: There is a $w_n \in \Sigma_I^{2^n - 1}$ without bad j-pairs for every j.

Proof

- Fix $\Sigma_I = \Sigma_O = \{1, \ldots, n\}$.
- $w \in \Sigma_I^*$ contains bad j-pair ($j \in \Sigma_I$) if there are two occurrences of j in w such that no $j' > j$ occurs in between.

Claim: There is a $w_n \in \Sigma_I^{2^n-1}$ without bad j-pairs for every j.

Construction by induction over n:

$n = 1$**:** $w_1 = 1$.

$n > 1$**:** $w_n = w_{n-1}\, n\, w_{n-1}$.

Proof

- Fix $\Sigma_I = \Sigma_O = \{1, \ldots, n\}$.
- $w \in \Sigma_I^*$ contains bad j-pair ($j \in \Sigma_I$) if there are two occurrences of j in w such that no $j' > j$ occurs in between.

Claim: There is a $w_n \in \Sigma_I^{2^n - 1}$ without bad j-pairs for every j.

Construction by induction over n:

$n = 1$: $w_1 = 1$.

$n > 1$: $w_n = w_{n-1}\, n\, w_{n-1}$.

$$1\ 2\ 1\ 3\ 1\ 2\ 1\ 4\ 1\ 2\ 1\ 3\ 1\ 2\ 1$$

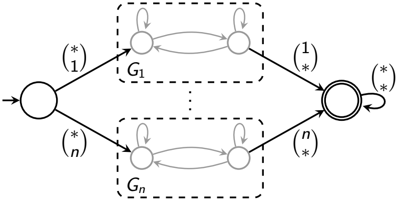
Proof continued

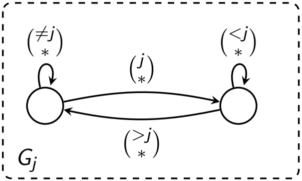
$\binom{\alpha}{\beta} \in L_n$ iff $\alpha(1)\alpha(2)\alpha(3)\cdots$ contains bad $\beta(0)$ pair, i.e., O has to find a bad j-pair in I's moves and play j as first move.

Proof continued

$\binom{\alpha}{\beta} \in L_n$ iff $\alpha(1)\alpha(2)\alpha(3)\cdots$ contains bad $\beta(0)$ pair, i.e., O has to find a bad j-pair in I's moves and play j as first move.

L_n is recognized by the following deterministic reachability automaton:





Proof continued

$\binom{\alpha}{\beta} \in L_n$ iff $\alpha(1)\alpha(2)\alpha(3)\cdots$ contains bad $\beta(0)$ pair, i.e., O has to find a bad j-pair in I's moves and play j as first move.

Claim: O wins $\Gamma_f(L_n)$ for some constant delay function f.

Proof continued

$\binom{\alpha}{\beta} \in L_n$ iff $\alpha(1)\alpha(2)\alpha(3)\cdots$ contains bad $\beta(0)$ pair, i.e., O has to find a bad j-pair in I's moves and play j as first move.

Claim: O wins $\Gamma_f(L_n)$ for some constant delay function f.

- Pick any f with $f(0) \geq 2^n + 1$, i.e., I has to pick a word $u \in \Sigma_I^{\geq 2^n+1}$ in round 0.
- Thus, u without its first letter contains a bad j-pair for some j.
- O picks such a j in round 0.
- The resulting play is winning for O, no matter how it is continued.

Proof continued

$\binom{\alpha}{\beta} \in L_n$ iff $\alpha(1)\alpha(2)\alpha(3)\cdots$ contains bad $\beta(0)$ pair, i.e., O has to find a bad j-pair in I's moves and play j as first move.

Claim: I wins $\Gamma_f(L_n)$ for every delay function f with $f(0) \leq 2^n$.

Proof continued

$\binom{\alpha}{\beta} \in L_n$ iff $\alpha(1)\alpha(2)\alpha(3)\cdots$ contains bad $\beta(0)$ pair, i.e., O has to find a bad j-pair in I's moves and play j as first move.

Claim: I wins $\Gamma_f(L_n)$ for every delay function f with $f(0) \leq 2^n$.

- Let f be a delay function with $f(0) \leq 2^n$.
- In round 0, I picks the prefix of $1w_n$ of length $f(0)$.
- Then, O has to pick some $j \in \Sigma_O$ in round 0.
- I completes w_n (if necessary) and then plays $j' \neq j$ ad infinitum.
- I wins the resulting play, as $w_n(j')^\omega$ does not contain a bad j-pair.

Remarks

The bad j-pair construction is very general:

- A similar construction witnesses an exponential lower bound for deterministic safety automata.
- Thus, exponential lookahead is necessary for any formalism that subsumes deterministic reachability or safety automata, in particular deterministic parity automata.

Remarks

The bad j-pair construction is very general:

- A similar construction witnesses an exponential lower bound for deterministic safety automata.
- Thus, exponential lookahead is necessary for any formalism that subsumes deterministic reachability or safety automata, in particular deterministic parity automata.
- Using the alphabet $\{1, \ldots, 2^n\}$ (encoded in binary) and some tricks yield doubly-exponential lower bounds for non-deterministic and universal automata.

Remarks

The bad j-pair construction is very general:

- A similar construction witnesses an exponential lower bound for deterministic safety automata.
- Thus, exponential lookahead is necessary for any formalism that subsumes deterministic reachability or safety automata, in particular deterministic parity automata.
- Using the alphabet $\{1, \ldots, 2^n\}$ (encoded in binary) and some tricks yield doubly-exponential lower bounds for non-deterministic and universal automata.
- Using the alphabet $\{1, \ldots, 2^{2^n}\}$ (encoded in binary) and even more tricks yield triply-exponential lower bounds for LTL and alternating automata.

Outline

1. Introduction

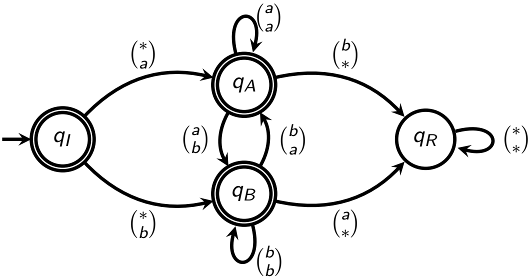
2. Lower Bounds on the Necessary Lookahead

3. **Solving Delay Games**

4. Conclusion

Solving Delay Games

We consider the special case of safety automata, which accept if only safe states are visited.



$$L(\mathcal{A}) = \{ \binom{\alpha(0)}{\beta(0)} \binom{\alpha(1)}{\beta(1)} \binom{\alpha(2)}{\beta(2)} \cdots \mid \beta(i) = \alpha(i+1) \text{ for every } i \}$$

Solving Delay Games

We consider the special case of safety automata, which accept if only safe states are visited.

Theorem

The following problem is in EXPTIME: *"Given a deterministic safety automaton $\mathcal{A}$, is there a delay function f such that O wins $\Gamma_f(L(\mathcal{A}))$?"*

Solving Delay Games

We consider the special case of safety automata, which accept if only safe states are visited.

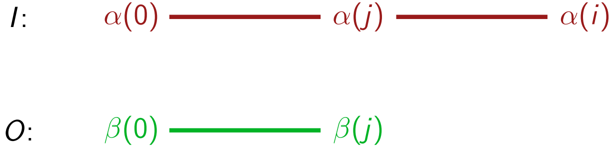
Theorem

The following problem is in ExpTime*: "Given a deterministic safety automaton $\mathcal{A}$, is there a delay function f such that O wins $\Gamma_f(L(\mathcal{A}))$?"*

W.l.o.g.: Every unsafe state of $\mathcal{A}$ is a sink.

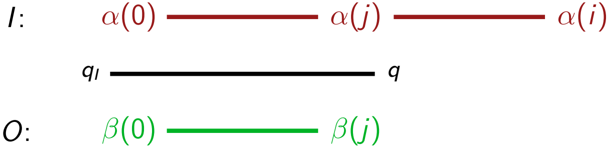
Proof

- Consider a typical situation during a play.



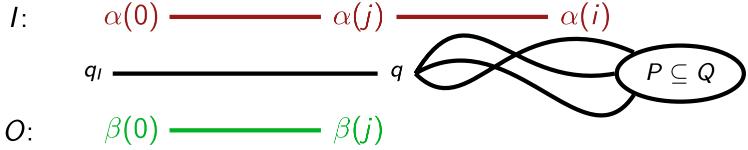
Proof

- Consider a typical situation during a play.



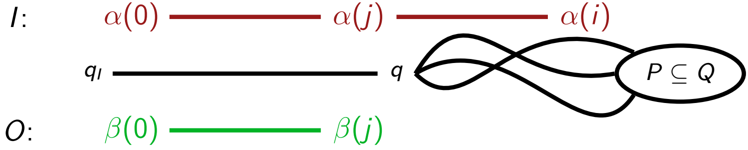
Proof

- Consider a typical situation during a play.



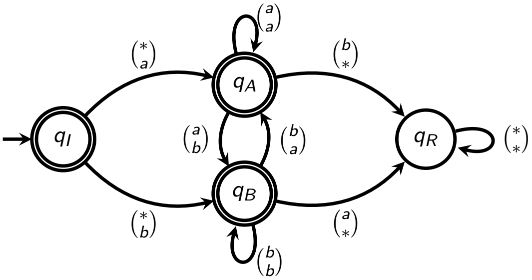
Proof

- Consider a typical situation during a play.

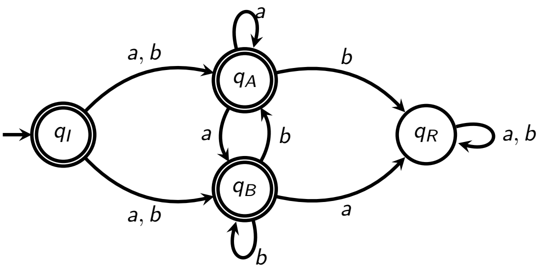


- We abstract moves of I by considering transition profiles of the (non-deterministic) projection automaton $\pi_{\Sigma_I}(\mathcal{A})$.

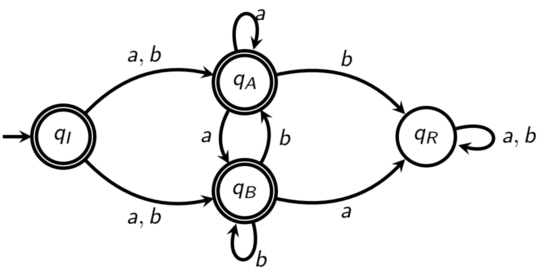
Transition Profiles



Transition Profiles

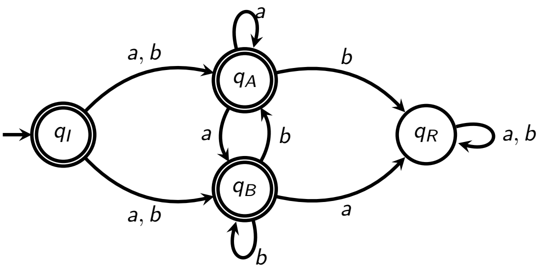


Transition Profiles

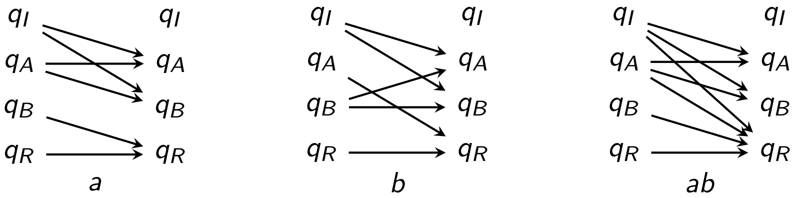


The transition profile of a word w contains for each state q the set of states reachable from q by processing w.

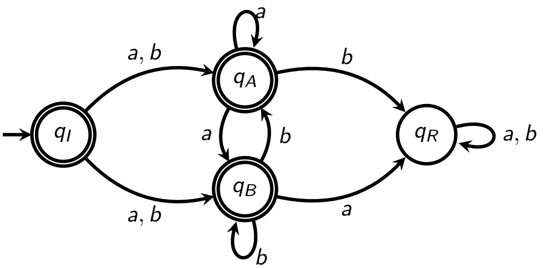
Transition Profiles



The transition profile of a word w contains for each state q the set of states reachable from q by processing w.



Transition Profiles



The transition profile of a word w contains for each state q the set of states reachable from q by processing w.

- There are at most $2^{|Q|^2}$ different transition profiles.
- For each transition profile, there is a DFA with $2^{|Q|^2}$ states recognizing all words of that profile.
- A transition profile is said to be infinite, if there are infinitely many words of that profile.

Removing Delay

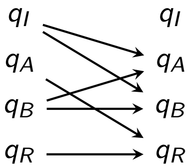
Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \ldots$ as follows:

- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.

Removing Delay

Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \ldots$ as follows:

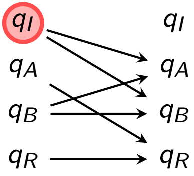
- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.



Removing Delay

Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \dots$ as follows:

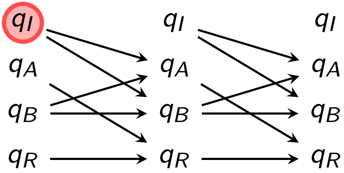
- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.



Removing Delay

Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \ldots$ as follows:

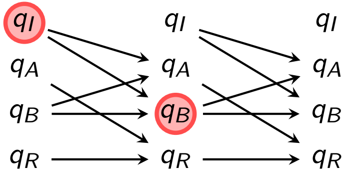
- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.



Removing Delay

Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \dots$ as follows:

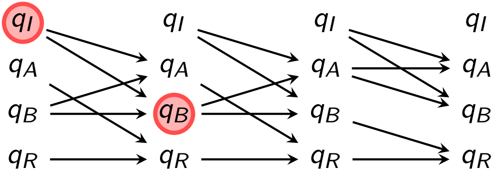
- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.



Removing Delay

Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \ldots$ as follows:

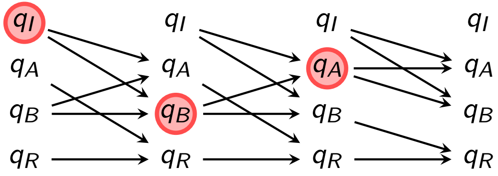
- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.



Removing Delay

Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \ldots$ as follows:

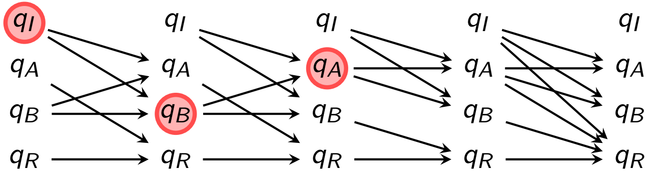
- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.



Removing Delay

Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \ldots$ as follows:

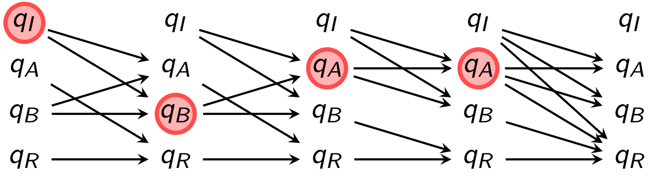
- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.



Removing Delay

Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \ldots$ as follows:

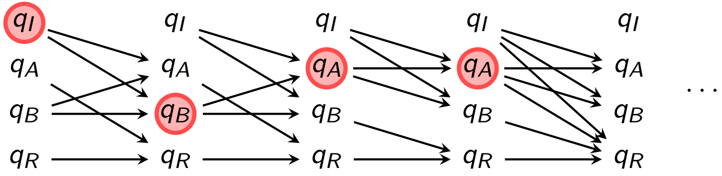
- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.



Removing Delay

Define the game $\mathcal{G}(\mathcal{A})$ played in rounds $i = 0, 1, 2, \ldots$ as follows:

- In round 0,
  - first I picks an infinite transition profile τ_0, then
  - O has to pick $q_0 = q_I$ (the initial state of $\mathcal{A}$).
- In round $i > 0$,
  - first I picks an infinite transition profile τ_i, then
  - O picks a state q_i that is reachable from q_{i-1} in the profile τ_{i-1}.
- O wins if each q_i is safe.



Equivalence

Lemma

The following are equivalent:

- *O wins $\Gamma_f(L(\mathcal{A}))$ for some f.*
- *O wins $\mathcal{G}(\mathcal{A})$.*

Note

$\mathcal{G}(\mathcal{A})$ can be modeled as a safety game of exponential size, which yields the desired exponential-time algorithm.

From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

$\mathcal{G}$ I:

O:

Γ I:

O:

From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

$\mathcal{G}$ I: τ_0

O:

Γ I:

O:

From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

$\mathcal{G}$

I: τ_0

O: $q_0 = q_I$

Γ

I:

O:

From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

$\mathcal{G}$

I: τ_0 τ_1

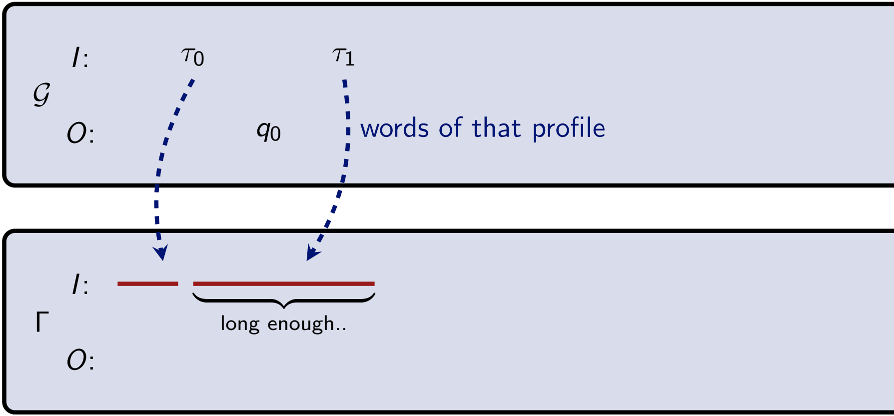
O: q_0

Γ

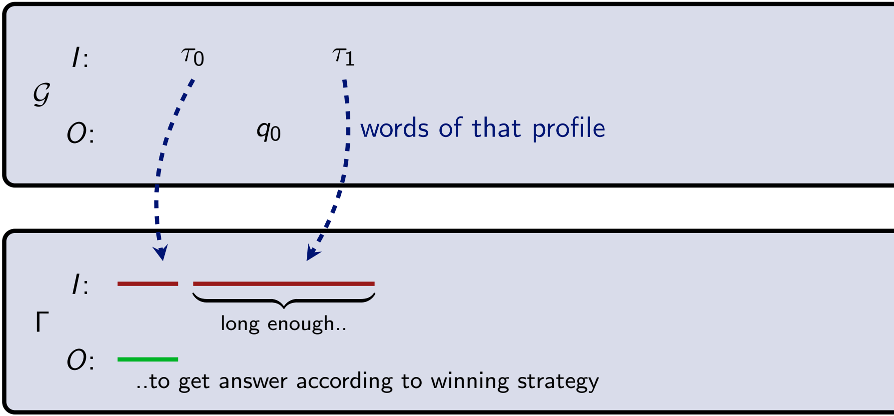
I:

O:

From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

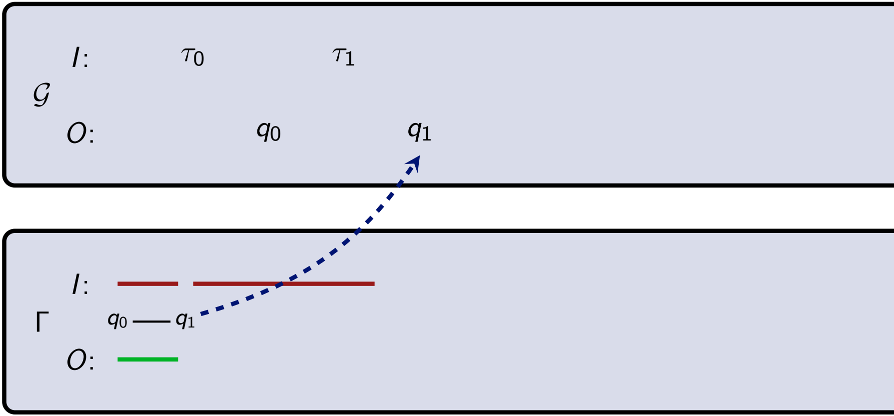


From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

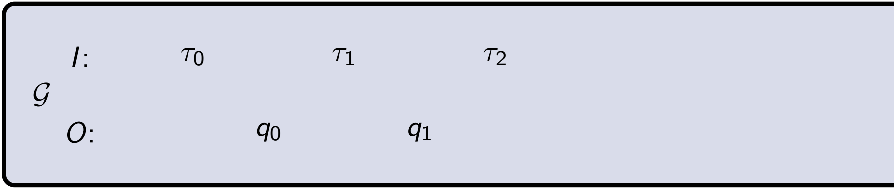


From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

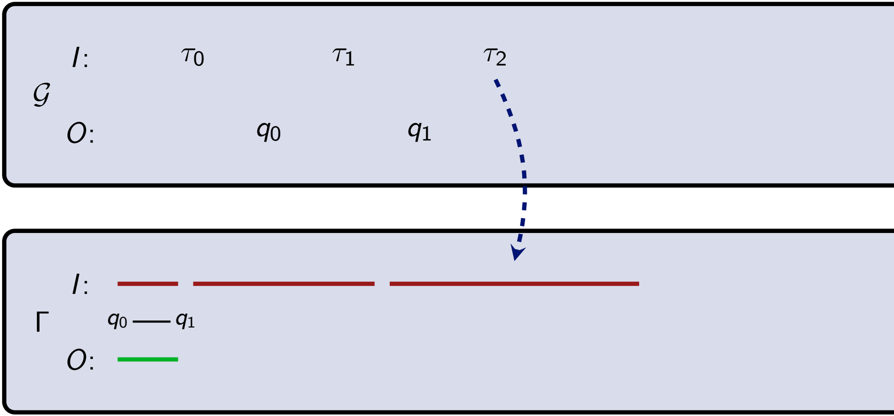
From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$



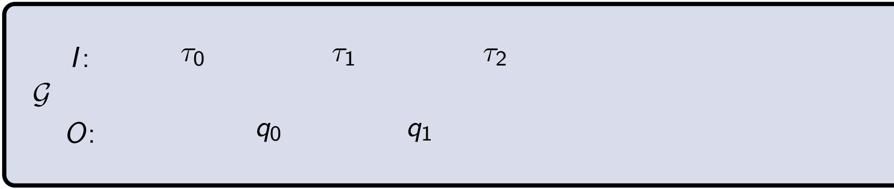
From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$



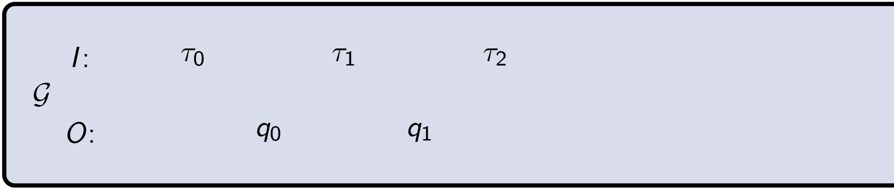
From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

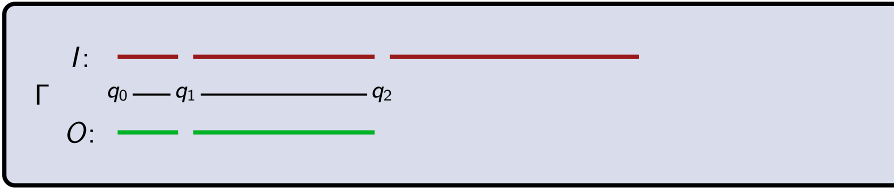


From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

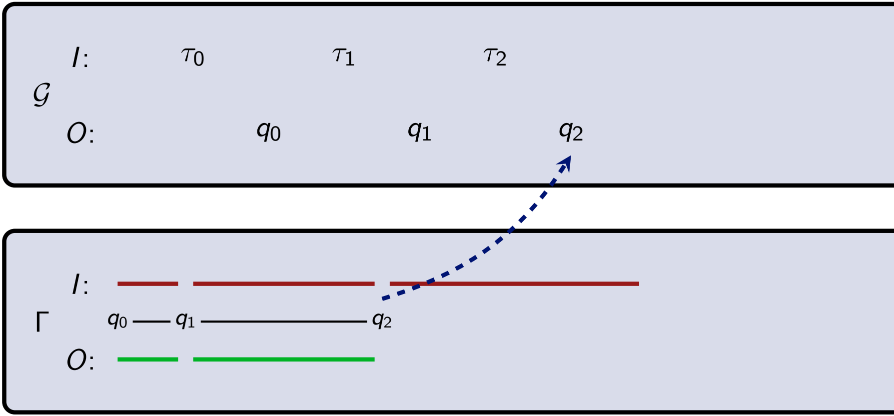


From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$



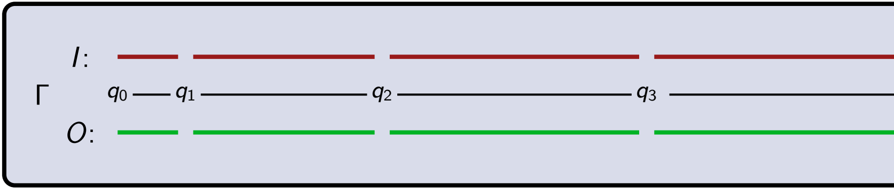


From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

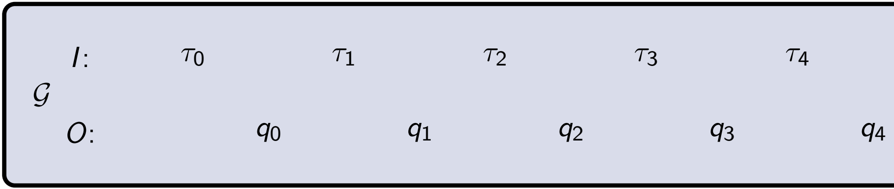


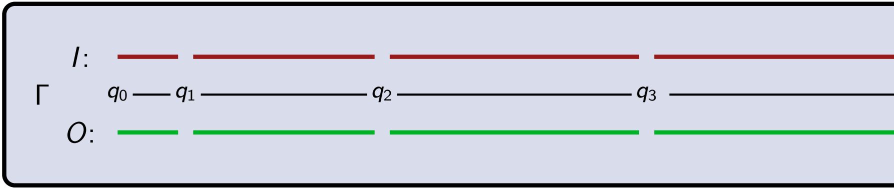
From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$

$\mathcal{G}$	I:	τ_0		τ_1		τ_2		τ_3		τ_4	
	O:		q_0		q_1		q_2		q_3		q_4



From $\Gamma_f(L(\mathcal{A}))$ to $\mathcal{G}(\mathcal{A})$





- The play in $\Gamma_f(L(\mathcal{A}))$ is consistent with a winning strategy for O.
- Hence, all q_i are safe, i.e., O wins the play in $\mathcal{G}(\mathcal{A})$.

From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.

I:

Γ

O:

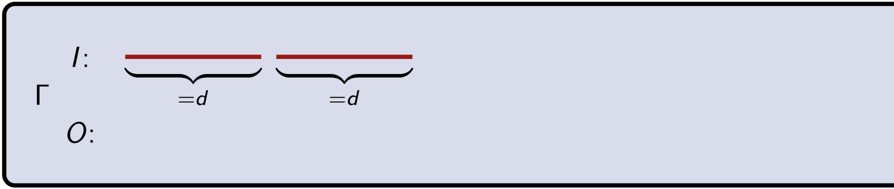
I:

$\mathcal{G}$

O:

From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.



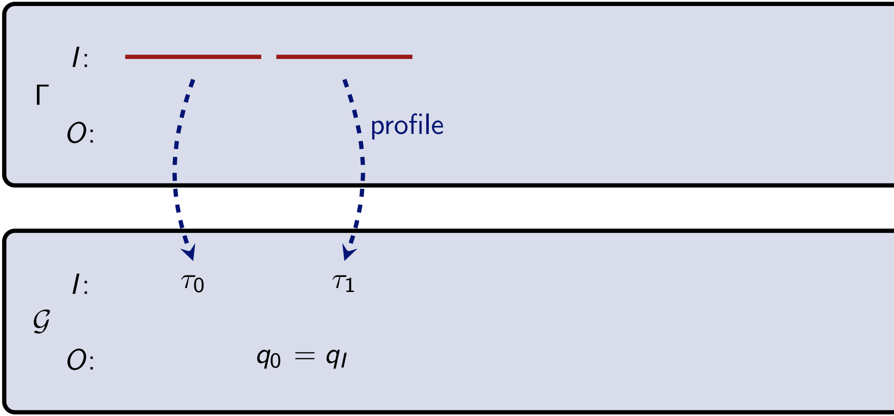
$\mathcal{G}$

I:

O:

From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.



From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.

Γ

I:

O:

$\mathcal{G}$

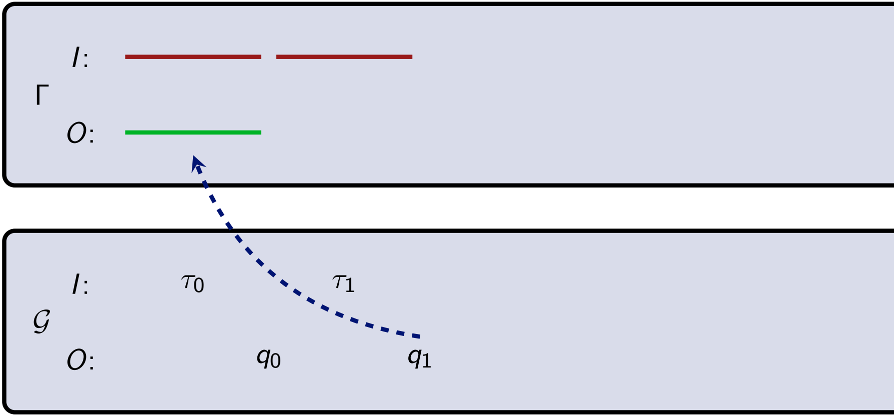
I: τ_0 τ_1

O: q_0 q_1

According to w.s.

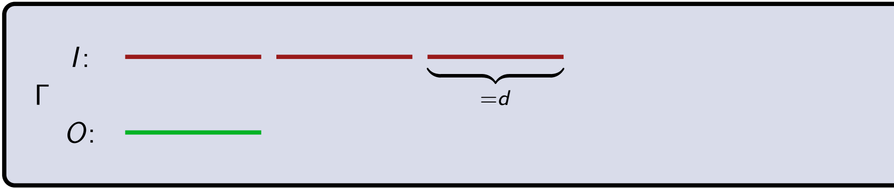
From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

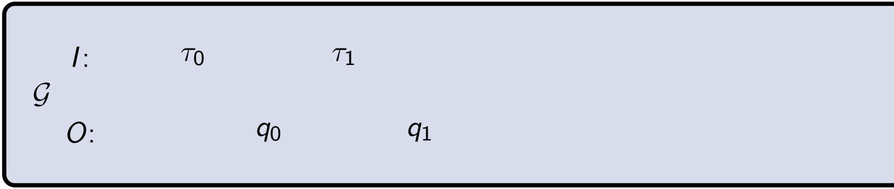
Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.



From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

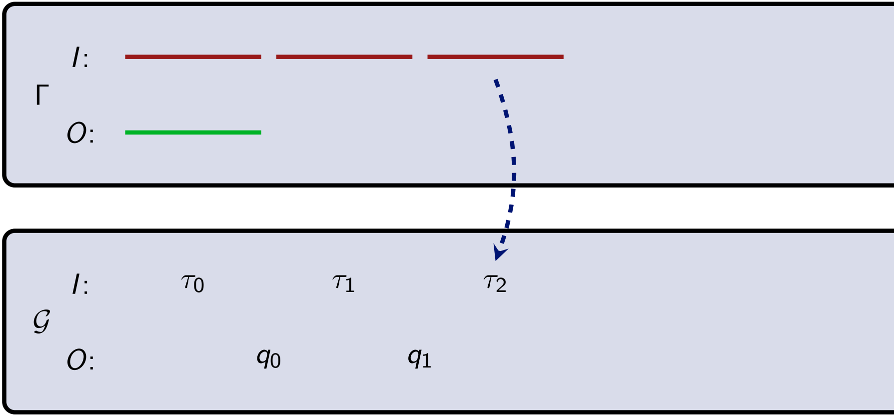
Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.





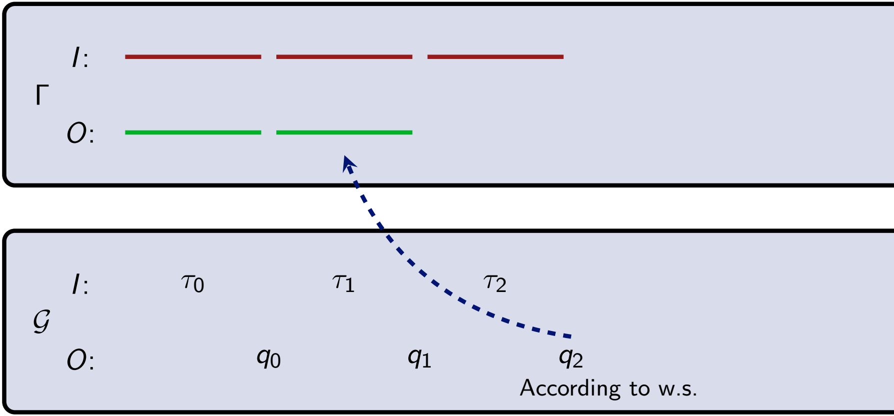
From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.



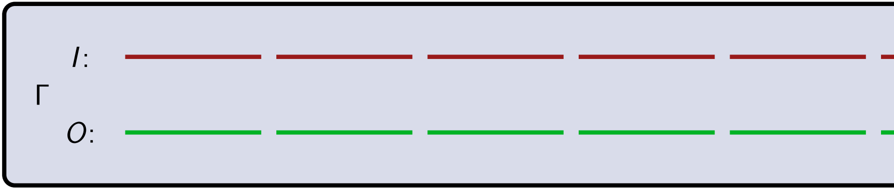
From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.



From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

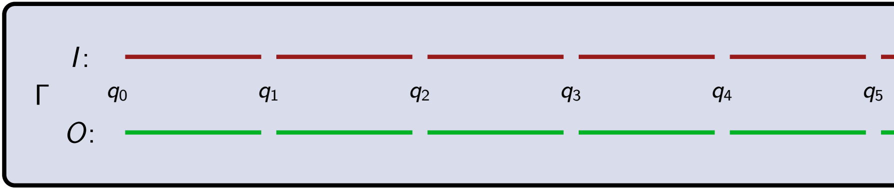
Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.

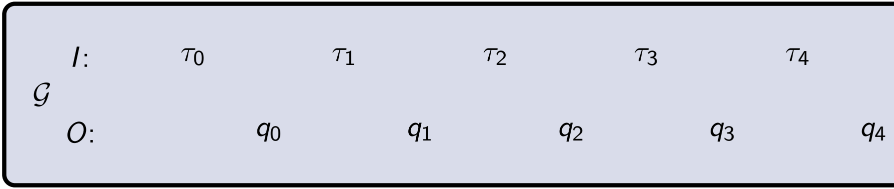


$\mathcal{G}$	I:	τ_0		τ_1		τ_2		τ_3		τ_4	
	O:		q_0		q_1		q_2		q_3		q_4

From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

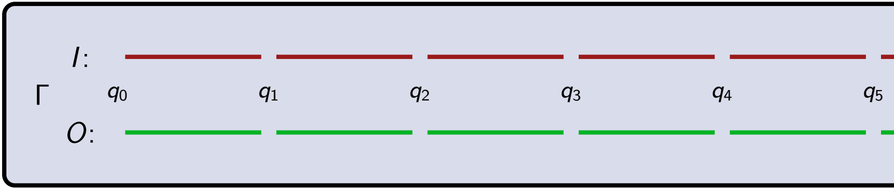
Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.

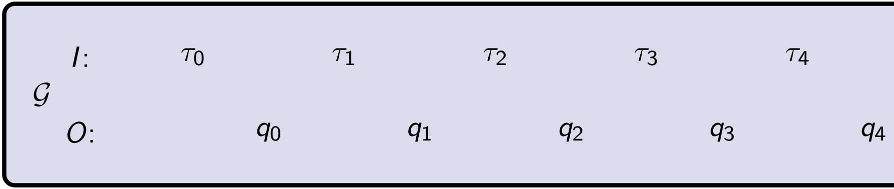




From $\mathcal{G}(\mathcal{A})$ to $\Gamma_f(L(\mathcal{A}))$

Let $d = 2^{|Q|^2}$ and $f(0) = 2d$.





- The play in $\mathcal{G}(\mathcal{A})$ is consistent with a winning strategy for O.
- Hence, all q_i are safe, i.e., O wins the play in $\Gamma_f(L(\mathcal{A}))$.

Remarks

Applying both directions yields an upper bound on the lookahead.

Corollary

The following are equivalent.

- *O wins $\Gamma_f(L(\mathcal{A}))$ for some f.*
- *O wins $\Gamma_f(L(\mathcal{A}))$ for the constant delay function f with $f(0) = 2^{|Q|^2+1}$.*

Remarks

Applying both directions yields an upper bound on the lookahead.

Corollary

The following are equivalent.

- *O wins $\Gamma_f(L(\mathcal{A}))$ for some f.*
- *O wins $\Gamma_f(L(\mathcal{A}))$ for the constant delay function f with $f(0) = 2^{|Q|^2+1}$.*

More results:

- By aggregating colors occurring during a run, the same technique is applicable to parity automata.

Remarks

Applying both directions yields an upper bound on the lookahead.

Corollary

The following are equivalent.

- *O wins $\Gamma_f(L(\mathcal{A}))$ for some f.*
- *O wins $\Gamma_f(L(\mathcal{A}))$ for the constant delay function f with $f(0) = 2^{|Q|^2+1}$.*

More results:

- By aggregating colors occurring during a run, the same technique is applicable to parity automata.
- In fact, it is applicable to every winning condition that can be aggregated in a certain sense, e.g., Muller, Rabin, Streett, and parity with costs.

Remarks

Applying both directions yields an upper bound on the lookahead.

Corollary

The following are equivalent.

- *O wins $\Gamma_f(L(\mathcal{A}))$ for some f.*
- *O wins $\Gamma_f(L(\mathcal{A}))$ for the constant delay function f with $f(0) = 2^{|Q|^2+1}$.*

More results:

- By aggregating colors occurring during a run, the same technique is applicable to parity automata.
- In fact, it is applicable to every winning condition that can be aggregated in a certain sense, e.g., Muller, Rabin, Streett, and parity with costs.
- ExpTime-hardness for safety delay games.

Outline

1. Introduction

2. Lower Bounds on the Necessary Lookahead

3. Solving Delay Games

4. **Conclusion**

Outlook

- Many aspects of the classical theory of infinite games have been transferred to the setting with delay.
- New interesting phenomena appear in this setting: bounds on the lookahead, tradeoffs, etc.

Outlook

- Many aspects of the classical theory of infinite games have been transferred to the setting with delay.
- New interesting phenomena appear in this setting: bounds on the lookahead, tradeoffs, etc.
- Many challenging problem are still open:
  - Delay games with succinct acceptance conditions, e.g., Muller, Rabin, Streett.
  - Lower bounds on necessary memory for finite-state strategies in delay games.
  - Solving delay games without reductions to delay-free games.
  - Delay games as optimization problem.