Completely Reachable Automata

Mikhail Volkov
(joint with Evgenija Bondar and David Fernando Casas Torres)

Ural Federal University, Ekaterinburg, Russia

June 22, 2021

Definitions and Terminology

We consider complete deterministic finite automata (DFAs).

Definitions and Terminology

We consider complete deterministic finite automata (DFAs). $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ where Q stands for the state set, Σ is the input alphabet, and $\delta\colon Q \times \Sigma \to Q$ is a (total) transition function.

To simplify notation we often introduce a DFA as $\langle Q, \Sigma\rangle$

Definitions and Terminology

We consider complete deterministic finite automata (DFAs). $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ where Q stands for the state set, Σ is the input alphabet, and $\delta\colon Q \times \Sigma \to Q$ is a (total) transition function.

To simplify notation we often introduce a DFA as $\langle Q, \Sigma\rangle$ and write $q.w$ for $\delta(q, w)$ and $P.w$ for $\{\delta(q, w) \mid q \in P\}$.

Definitions and Terminology

We consider complete deterministic finite automata (DFAs). $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ where Q stands for the state set, Σ is the input alphabet, and $\delta\colon Q \times \Sigma \to Q$ is a (total) transition function.

To simplify notation we often introduce a DFA as $\langle Q, \Sigma \rangle$ and write $q.w$ for $\delta(q, w)$ and $P.w$ for $\{\delta(q, w) \mid q \in P\}$.

Given a DFA $\mathscr{A} = \langle Q, \Sigma \rangle$, a non-empty subset $P \subseteq Q$ is reachable in $\mathscr{A}$ if $P = Q.w$ for some word $w \in \Sigma^*$.

Definitions and Terminology

We consider complete deterministic finite automata (DFAs). $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ where Q stands for the state set, Σ is the input alphabet, and $\delta\colon Q \times \Sigma \to Q$ is a (total) transition function.

To simplify notation we often introduce a DFA as $\langle Q, \Sigma \rangle$ and write $q.w$ for $\delta(q, w)$ and $P.w$ for $\{\delta(q, w) \mid q \in P\}$.

Given a DFA $\mathscr{A} = \langle Q, \Sigma \rangle$, a non-empty subset $P \subseteq Q$ is reachable in $\mathscr{A}$ if $P = Q.w$ for some word $w \in \Sigma^*$. A DFA is completely reachable if every non-empty set of its states is reachable.

A DFA $\mathscr{A} = \langle Q, \Sigma \rangle$ is synchronizing if there are a word $w \in \Sigma^*$ and a state $f \in Q$ such that the action of w resets $\mathscr{A}$ to f no matter at which state the action started: $q.w = f$ for all $q \in Q$.

Motivation: Synchronizing Automata

A DFA $\mathscr{A} = \langle Q, \Sigma \rangle$ is synchronizing if there are a word $w \in \Sigma^*$ and a state $f \in Q$ such that the action of w resets $\mathscr{A}$ to f no matter at which state the action started: $q.w = f$ for all $q \in Q$. In short, $|Q.w| = 1$; that is, a singleton is reachable in $\mathscr{A}$.

Motivation: Synchronizing Automata

A DFA $\mathscr{A} = \langle Q, \Sigma \rangle$ is synchronizing if there are a word $w \in \Sigma^*$ and a state $f \in Q$ such that the action of w resets $\mathscr{A}$ to f no matter at which state the action started: $q.w = f$ for all $q \in Q$. In short, $|Q.w| = 1$; that is, a singleton is reachable in $\mathscr{A}$.

Hence, a completely reachable automaton is synchronizing.

Motivation: Synchronizing Automata

A DFA $\mathscr{A} = \langle Q, \Sigma \rangle$ is synchronizing if there are a word $w \in \Sigma^*$ and a state $f \in Q$ such that the action of w resets $\mathscr{A}$ to f no matter at which state the action started: $q.w = f$ for all $q \in Q$. In short, $|Q.w| = 1$; that is, a singleton is reachable in $\mathscr{A}$.

Hence, a completely reachable automaton is synchronizing.

Any w with $|Q.w| = 1$ is a reset word for $\mathscr{A}$.

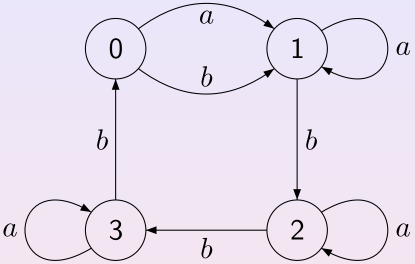
Motivation: Synchronizing Automata

A DFA $\mathscr{A} = \langle Q, \Sigma \rangle$ is synchronizing if there are a word $w \in \Sigma^*$ and a state $f \in Q$ such that the action of w resets $\mathscr{A}$ to f no matter at which state the action started: $q.w = f$ for all $q \in Q$. In short, $|Q.w| = 1$; that is, a singleton is reachable in $\mathscr{A}$.

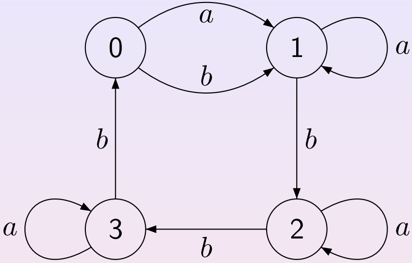
Hence, a completely reachable automaton is synchronizing.

Any w with $|Q.w| = 1$ is a reset word for $\mathscr{A}$. The minimum length of reset words for $\mathscr{A}$ is called the reset threshold of $\mathscr{A}$.

An Example

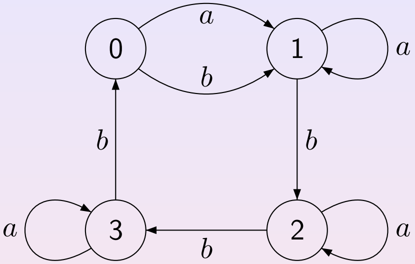


An Example



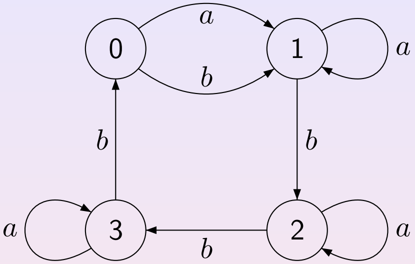
A reset word is $abbbabbba$: applying it at any state brings this automaton to the state 1.

An Example



A reset word is $abbbabbba$: applying it at any state brings this automaton to the state 1. In fact, this is the reset word of minimum length for the automaton whence its reset threshold is 9.

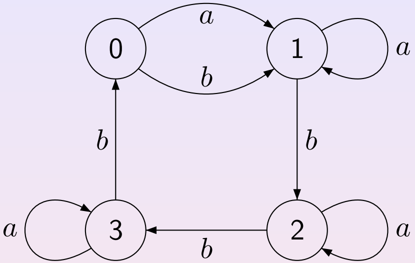
An Example



A reset word is $abbbabbba$: applying it at any state brings this automaton to the state 1. In fact, this is the reset word of minimum length for the automaton whence its reset threshold is 9.

The automaton belongs to the series $\{\mathscr{C}_n\}$ found by Jan Černý in 1964.

An Example



A reset word is $abbbabbba$: applying it at any state brings this automaton to the state 1. In fact, this is the reset word of minimum length for the automaton whence its reset threshold is 9.

The automaton belongs to the series $\{\mathscr{C}_n\}$ found by Jan Černý in 1964. For each $n > 1$, the automaton $\mathscr{C}_n$ has n states, 2 input letters and reset threshold $(n-1)^2$.

The states of $\mathscr{C}_n$ are the residues modulo n, and the input letters a and b act as follows:

$$0.a = 1, \quad m.a = m \text{ for } 0 < m < n, \quad m.b = m + 1 \pmod{n}.$$

The states of $\mathscr{C}_n$ are the residues modulo n, and the input letters a and b act as follows:

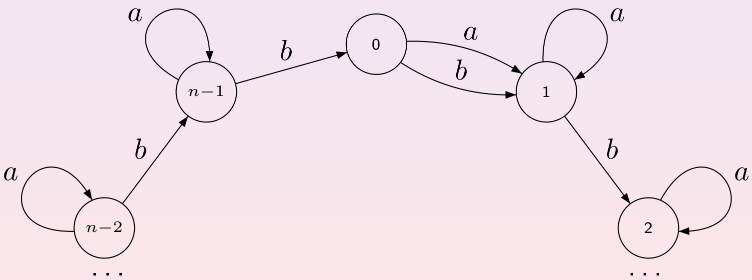
$$0.a = 1, \quad m.a = m \text{ for } 0 < m < n, \quad m.b = m + 1 \pmod{n}.$$

The automaton in the previous slide is $\mathscr{C}_4$.

The states of $\mathscr{C}_n$ are the residues modulo n, and the input letters a and b act as follows:

$$0.a = 1, \quad m.a = m \text{ for } 0 < m < n, \quad m.b = m + 1 \pmod{n}.$$

The automaton in the previous slide is $\mathscr{C}_4$.
Here is a generic automaton from the Černý series:

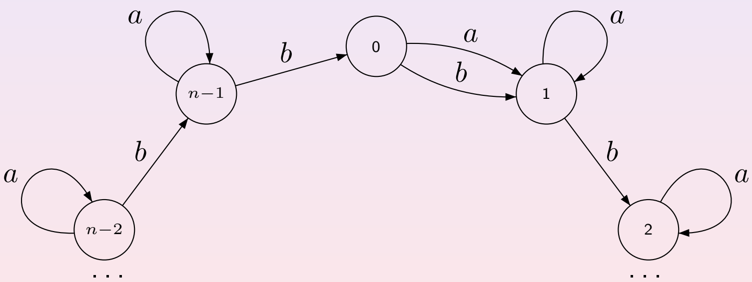


Černý Series

The states of $\mathscr{C}_n$ are the residues modulo n, and the input letters a and b act as follows:

$$0.a = 1, \quad m.a = m \text{ for } 0 < m < n, \quad m.b = m + 1 \pmod{n}.$$

The automaton in the previous slide is $\mathscr{C}_4$.
Here is a generic automaton from the Černý series:



Černý has proved that the shortest reset word for $\mathscr{C}_n$ is $(ab^{n-1})^{n-2}a$ of length $n(n-2)+1 = (n-1)^2$.

Define the Černý function $C(n)$ as the maximum reset threshold of all synchronizing automata with n states. The above property of the series $\{\mathscr{C}_n\}$ yields the inequality $C(n) \geq (n-1)^2$.

Define the Černý function $C(n)$ as the maximum reset threshold of all synchronizing automata with n states. The above property of the series $\{\mathscr{C}_n\}$ yields the inequality $C(n) \geq (n-1)^2$.
The Černý conjecture is the claim that in fact the equality $C(n) = (n-1)^2$ holds true.

Černý Conjecture

Define the Černý function $C(n)$ as the maximum reset threshold of all synchronizing automata with n states. The above property of the series $\{\mathscr{C}_n\}$ yields the inequality $C(n) \geq (n-1)^2$.
The Černý conjecture is the claim that in fact the equality $C(n) = (n-1)^2$ holds true. This simply looking conjecture is arguably the most longstanding open problem in the combinatorial theory of finite automata.

Černý Conjecture

Define the Černý function $C(n)$ as the maximum reset threshold of all synchronizing automata with n states. The above property of the series $\{\mathscr{C}_n\}$ yields the inequality $C(n) \geq (n-1)^2$.
The Černý conjecture is the claim that in fact the equality $C(n) = (n-1)^2$ holds true. Up to recently, everything we knew about the conjecture in general could be summarized in one line:

$$(n-1)^2 \leq C(n) \leq \frac{n^3 - n}{6}.$$

Define the Černý function $C(n)$ as the maximum reset threshold of all synchronizing automata with n states. The above property of the series $\{\mathscr{C}_n\}$ yields the inequality $C(n) \geq (n-1)^2$. The Černý conjecture is the claim that in fact the equality $C(n) = (n-1)^2$ holds true. Up to recently, everything we knew about the conjecture in general could be summarized in one line:

$$(n-1)^2 \leq C(n) \leq \frac{n^3 - n}{6}.$$

A small improvement on this bound has been found by Marek Szykuła (STACS 2018): the new bound is still cubic in n but improves the coefficient $\frac{1}{6} = 0.1666\ldots$ at n^3

Černý Conjecture

Define the Černý function $C(n)$ as the maximum reset threshold of all synchronizing automata with n states. The above property of the series $\{\mathscr{C}_n\}$ yields the inequality $C(n) \geq (n-1)^2$. The Černý conjecture is the claim that in fact the equality $C(n) = (n-1)^2$ holds true. Up to recently, everything we knew about the conjecture in general could be summarized in one line:

$$(n-1)^2 \leq C(n) \leq \frac{n^3 - n}{6}.$$

A small improvement on this bound has been found by Marek Szykuła (STACS 2018): the new bound is still cubic in n but improves the coefficient $\frac{1}{6} = 0.1666\ldots$ at n^3 by $\frac{125}{511104} \approx 0.000245$ so that it becomes ≈ 0.1664.

Černý Conjecture

Define the Černý function $C(n)$ as the maximum reset threshold of all synchronizing automata with n states. The above property of the series $\{\mathscr{C}_n\}$ yields the inequality $C(n) \geq (n-1)^2$.
The Černý conjecture is the claim that in fact the equality $C(n) = (n-1)^2$ holds true. Up to recently, everything we knew about the conjecture in general could be summarized in one line:

$$(n-1)^2 \leq C(n) \leq \frac{n^3 - n}{6}.$$

A small improvement on this bound has been found by Marek Szykuła (STACS 2018): the new bound is still cubic in n but improves the coefficient $\frac{1}{6} = 0.1666\ldots$ at n^3
by $\frac{125}{511104} \approx 0.000245$ so that it becomes ≈ 0.1664.

The new bound is $\dfrac{85059n^3 + 90024n^2 + 196504n - 10648}{511104}$.

Černý Conjecture

Define the Černý function $C(n)$ as the maximum reset threshold of all synchronizing automata with n states. The above property of the series $\{\mathscr{C}_n\}$ yields the inequality $C(n) \geq (n-1)^2$.
The Černý conjecture is the claim that in fact the equality $C(n) = (n-1)^2$ holds true. Up to recently, everything we knew about the conjecture in general could be summarized in one line:

$$(n-1)^2 \leq C(n) \leq \frac{n^3 - n}{6}.$$

A small improvement on this bound has been found by Marek Szykuła (STACS 2018): the new bound is still cubic in n but improves the coefficient $\frac{1}{6} = 0.1666\ldots$ at n^3
by $\frac{125}{511104} \approx 0.000245$ so that it becomes ≈ 0.1664.

The new bound is $\dfrac{85059n^3 + 90024n^2 + 196504n - 10648}{511104}$.
In 2019 Yaroslav Shitov found a further improvement to ≈ 0.1654.

Teaser

Those who got interested in the Černý Conjecture can read more in the forthcoming "Handbook of Automata Theory"
(EMS Publishing House, in print).

Teaser

Those who got interested in the Černý Conjecture can read more in the forthcoming "Handbook of Automata Theory" (EMS Publishing House, in print). See Chapter 15 in Volume I: "Černý's conjecture and the Road Coloring Problem" by Jarkko Kari and MV.

Teaser

Those who got interested in the Černý Conjecture can read more in the forthcoming "Handbook of Automata Theory" (EMS Publishing House, in print). See Chapter 15 in Volume I: "Černý's conjecture and the Road Coloring Problem" by Jarkko Kari and MV.

The latest developments may be found in the special issue of the "Journal of Automata, Languages and Combinatorics" (Volume 24 (2019), no.2–4), freely available under `https://jalc.de/issues/2019/issue_24_2-4/content.html`.

Teaser

Those who got interested in the Černý Conjecture can read more in the forthcoming "Handbook of Automata Theory" (EMS Publishing House, in print). See Chapter 15 in Volume I: "Černý's conjecture and the Road Coloring Problem" by Jarkko Kari and MV.

The latest developments may be found in the special issue of the "Journal of Automata, Languages and Combinatorics" (Volume 24 (2019), no.2–4), freely available under `https://jalc.de/issues/2019/issue_24_2-4/content.html`. In particular, the special issue contains Shitov's paper with the best bound up to date.

Approaching the Černý Conjecture

Since the Černý Conjecture has proved to be hard in general, a natural strategy consists in considering its restrictions to some special classes of DFAs.

Approaching the Černý Conjecture

Since the Černý Conjecture has proved to be hard in general, a natural strategy consists in considering its restrictions to some special classes of DFAs.

The conjecture has been proved for many important special cases.

Since the Černý Conjecture has proved to be hard in general, a natural strategy consists in considering its restrictions to some special classes of DFAs.
The conjecture has been proved for many important special cases. This includes for instance:
• Louis Dubuc's result for automata in which a letter acts on the state set Q as a cyclic permutation of order $|Q|$ (Sur le automates circulaires et la conjecture de Černý, RAIRO Inform. Theor. Appl., 32 (1998) 21–34 [in French]).

Approaching the Černý Conjecture

Since the Černý Conjecture has proved to be hard in general, a natural strategy consists in considering its restrictions to some special classes of DFAs.

The conjecture has been proved for many important special cases. This includes for instance:

• Louis Dubuc's result for automata in which a letter acts on the state set Q as a cyclic permutation of order $|Q|$ (Sur le automates circulaires et la conjecture de Černý, RAIRO Inform. Theor. Appl., 32 (1998) 21–34 [in French]).

• Jarkko Kari's result for automata with Eulerian digraphs (Synchronizing finite automata on Eulerian digraphs, Theoret. Comput. Sci., 295 (2003) 223–232).

Approaching the Černý Conjecture

Since the Černý Conjecture has proved to be hard in general, a natural strategy consists in considering its restrictions to some special classes of DFAs.
The conjecture has been proved for many important special cases. This includes for instance:

• Louis Dubuc's result for automata in which a letter acts on the state set Q as a cyclic permutation of order $|Q|$ (Sur le automates circulaires et la conjecture de Černý, RAIRO Inform. Theor. Appl., 32 (1998) 21–34 [in French]).

• Jarkko Kari's result for automata with Eulerian digraphs (Synchronizing finite automata on Eulerian digraphs, Theoret. Comput. Sci., 295 (2003) 223–232).

• Avraam Trahtman result for automata whose transition monoid contains no non-trivial subgroups (The Černý conjecture for aperiodic automata, Discrete Math. Theoret. Comp. Sci., 9, no.2 (2007), 3–10).

An Observation

Observation (Marina Maslennikova, arXiv:1404.2816 (2014); Henk Don, Electronic J. Combinatorics 23 (2016) #P3.12)

The Černý automata $\mathscr{C}_n$ are completely reachable.

An Observation

Observation (Marina Maslennikova, arXiv:1404.2816 (2014); Henk Don, Electronic J. Combinatorics 23 (2016) #P3.12)

The Černý automata $\mathscr{C}_n$ are completely reachable.

In an implicit form, this observation is contained in a result due to Donald B. McAlister, Comm. Algebra **26**(2) (1998) 515–547, who provided a comprehensive analysis of the submonoid generated by the transformations from the definition of the Černý automata in the transformation monoid on the set $\{0, 1, \ldots, n-1\}$.

An Observation

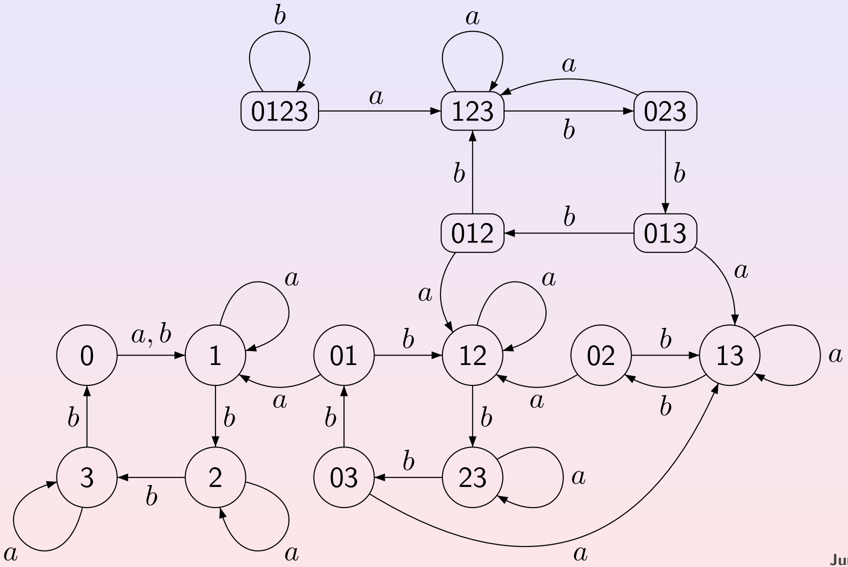
Observation (Marina Maslennikova, arXiv:1404.2816 (2014); Henk Don, Electronic J. Combinatorics 23 (2016) #P3.12)

The Černý automata $\mathscr{C}_n$ are completely reachable.

In an implicit form, this observation is contained in a result due to Donald B. McAlister, Comm. Algebra **26**(2) (1998) 515–547, who provided a comprehensive analysis of the submonoid generated by the transformations from the definition of the Černý automata in the transformation monoid on the set $\{0, 1, \ldots, n-1\}$.

For an illustration, consider the power-set automaton of the Černý automaton $\mathscr{C}_4$.

An Observation



Restricting to Completely Reachable Automata

Recall that every completely reachable automaton is synchronizing.

Restricting to Completely Reachable Automata

Recall that every completely reachable automaton is synchronizing. On the other hand, the above observation ensures that the lower bound $(n-1)^2$ for the Černý function $C(n)$ is attained by a family of completely reachable automata.

Restricting to Completely Reachable Automata

Recall that every completely reachable automaton is synchronizing. On the other hand, the above observation ensures that the lower bound $(n-1)^2$ for the Černý function $C(n)$ is attained by a family of completely reachable automata.

Therefore completely reachable automata form quite a natural class to study from the viewpoint of the Černý conjecture.

Restricting to Completely Reachable Automata

Recall that every completely reachable automaton is synchronizing. On the other hand, the above observation ensures that the lower bound $(n-1)^2$ for the Černý function $C(n)$ is attained by a family of completely reachable automata.

Therefore completely reachable automata form quite a natural class to study from the viewpoint of the Černý conjecture.

There are further reasons to study complete reachability

Restricting to Completely Reachable Automata

Recall that every completely reachable automaton is synchronizing. On the other hand, the above observation ensures that the lower bound $(n-1)^2$ for the Černý function $C(n)$ is attained by a family of completely reachable automata.

Therefore completely reachable automata form quite a natural class to study from the viewpoint of the Černý conjecture.

There are further reasons to study complete reachability . . . which we skip since we need time to introduce certain notions that are used for our main results.

Let $\mathscr{A} = \langle Q, \Sigma \rangle$ be a DFA. The defect of a word $w \in \Sigma^*$ w.r.t. $\mathscr{A}$ is $\mathrm{df}(w) := |Q \backslash Q.w|$.

Let $\mathscr{A} = \langle Q, \Sigma\rangle$ be a DFA. The defect of a word $w \in \Sigma^*$ w.r.t. $\mathscr{A}$ is $\mathrm{df}(w) := |Q\backslash Q.w|$.

If $\mathrm{df}(w) = 1$, the set $Q\backslash Q.w$ consists of a unique state $\mathrm{excl}(w)$, the excluded state.

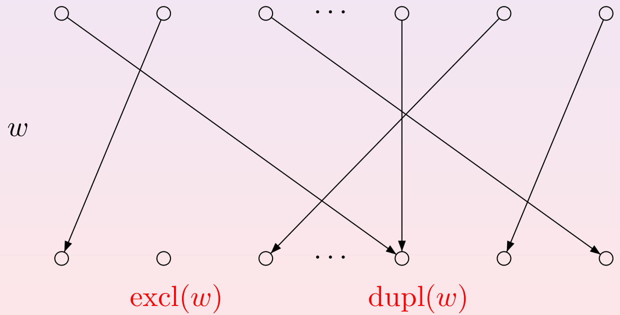
Let $\mathscr{A} = \langle Q, \Sigma \rangle$ be a DFA. The defect of a word $w \in \Sigma^*$ w.r.t. $\mathscr{A}$ is $\mathrm{df}(w) := |Q \backslash Q.w|$.

If $\mathrm{df}(w) = 1$, the set $Q \backslash Q.w$ consists of a unique state $\mathrm{excl}(w)$, the excluded state. The set $Q.w$ contains a unique duplicate state p such that $p = q_1.w = q_2.w$ for some $q_1 \neq q_2$; this state p is denoted by $\mathrm{dupl}(w)$.

Graph $\Gamma_1(\mathscr{A})$

Let $\mathscr{A} = \langle Q, \Sigma \rangle$ be a DFA. The defect of a word $w \in \Sigma^*$ w.r.t. $\mathscr{A}$ is $\mathrm{df}(w) := |Q \backslash Q.w|$.

If $\mathrm{df}(w) = 1$, the set $Q \backslash Q.w$ consists of a unique state $\mathrm{excl}(w)$, the excluded state. The set $Q.w$ contains a unique duplicate state p such that $p = q_1.w = q_2.w$ for some $q_1 \neq q_2$; this state p is denoted by $\mathrm{dupl}(w)$.



Graph $\Gamma_1(\mathscr{A})$

Let $\mathscr{A} = \langle Q, \Sigma \rangle$ be a DFA. The defect of a word $w \in \Sigma^*$ w.r.t. $\mathscr{A}$ is $\mathrm{df}(w) := |Q \backslash Q.w|$.

If $\mathrm{df}(w) = 1$, the set $Q \backslash Q.w$ consists of a unique state $\mathrm{excl}(w)$, the excluded state. The set $Q.w$ contains a unique duplicate state p such that $p = q_1.w = q_2.w$ for some $q_1 \neq q_2$; this state p is denoted by $\mathrm{dupl}(w)$.
Let $\Gamma_1(\mathscr{A})$ denote the graph with the vertex set Q and the edge set

$$E_1 := \{(\mathrm{excl}(w), \mathrm{dupl}(w)) \mid \mathrm{df}(w) = 1)\}.$$

Graph $\Gamma_1(\mathscr{A})$

Let $\mathscr{A} = \langle Q, \Sigma \rangle$ be a DFA. The defect of a word $w \in \Sigma^*$ w.r.t. $\mathscr{A}$ is $\mathrm{df}(w) := |Q \backslash Q.w|$.

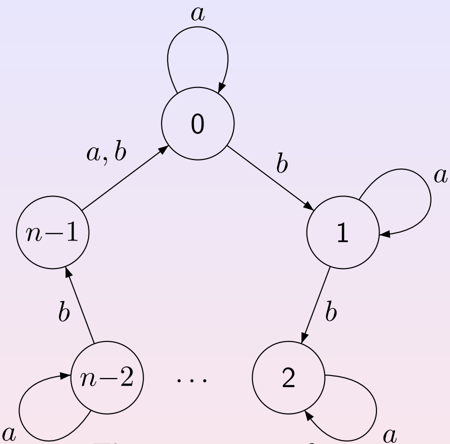
If $\mathrm{df}(w) = 1$, the set $Q \backslash Q.w$ consists of a unique state $\mathrm{excl}(w)$, the excluded state. The set $Q.w$ contains a unique duplicate state p such that $p = q_1.w = q_2.w$ for some $q_1 \neq q_2$; this state p is denoted by $\mathrm{dupl}(w)$.
Let $\Gamma_1(\mathscr{A})$ denote the graph with the vertex set Q and the edge set

$$E_1 := \{(\mathrm{excl}(w), \mathrm{dupl}(w)) \mid \mathrm{df}(w) = 1)\}.$$

Theorem (Bondar and MV, DCFS 2016)

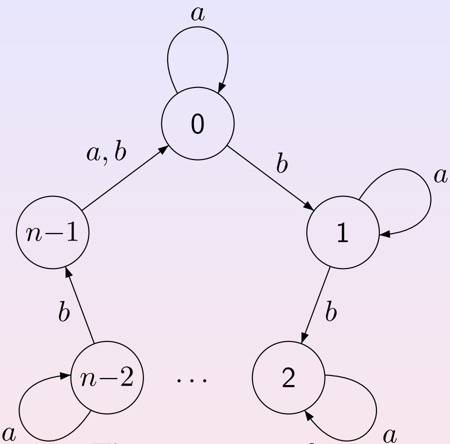
If a DFA $\mathscr{A} = \langle Q, \Sigma \rangle$ is such that the graph $\Gamma_1(\mathscr{A})$ is strongly connected, then $\mathscr{A}$ is completely reachable; more precisely, for every non-empty subset $P \subseteq Q$, there is a product w of words of defect 1 such that $P = Q.w$.

Example

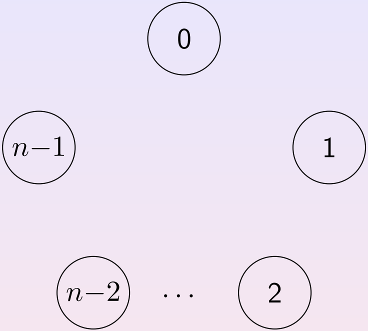


The automaton $\mathscr{C}_n$

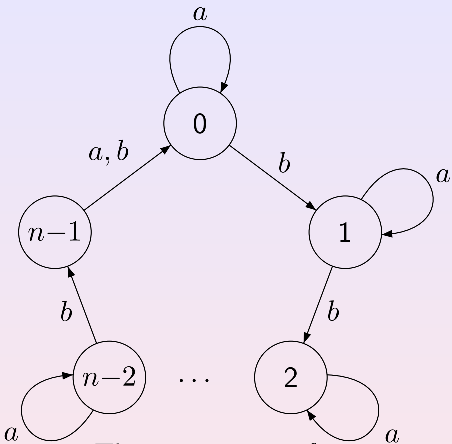
Example



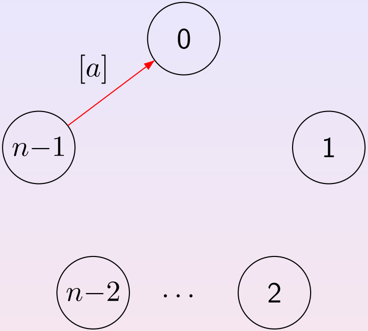
The automaton $\mathscr{C}_n$



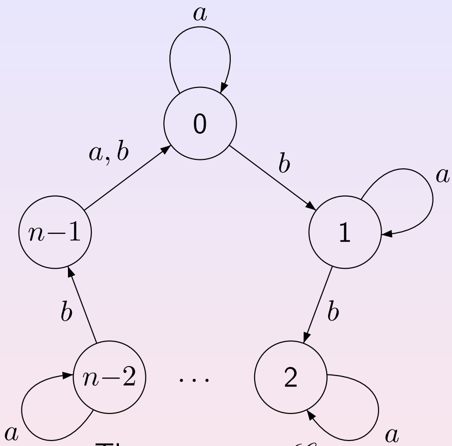
Example



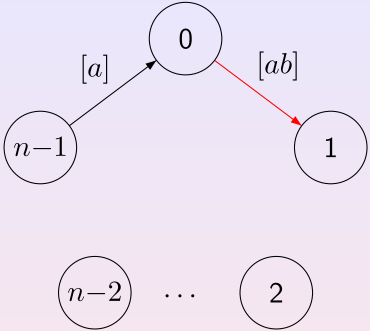
The automaton $\mathscr{C}_n$

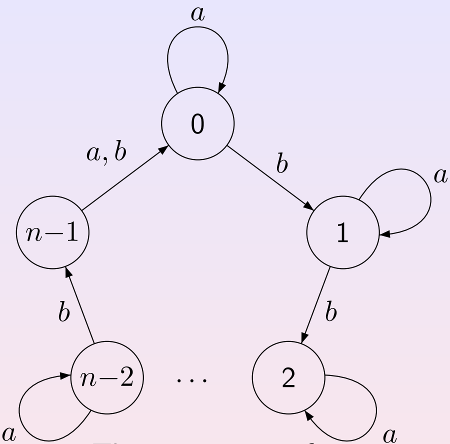


Example

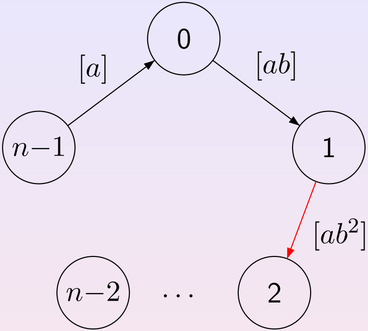


The automaton $\mathscr{C}_n$

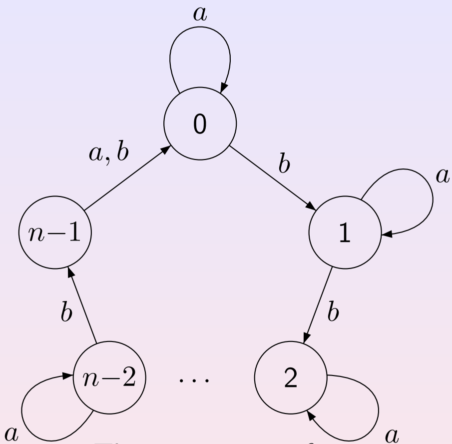




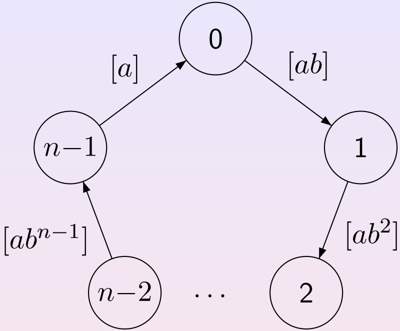
The automaton $\mathscr{C}_n$



Example



The automaton $\mathscr{C}_n$



$\Gamma_1(\mathscr{C}_n)$ is strongly connected

The converse of this theorem does not hold: if $\mathscr{A}$ is a completely reachable automaton, and even if for every non-empty subset $P \subseteq Q$, there is a product w of words of defect 1 such that $P = Q.w$, the graph $\Gamma_1(\mathscr{A})$ need not be strongly connected.

Graphs $\Gamma_k(\mathscr{A})$

The converse of this theorem does not hold: if $\mathscr{A}$ is a completely reachable automaton, and even if for every non-empty subset $P \subseteq Q$, there is a product w of words of defect 1 such that $P = Q.w$, the graph $\Gamma_1(\mathscr{A})$ need not be strongly connected. An example was in our DCFS 2016 paper, and a stronger example was found by François Gonze and Raphaël Jungers, DLT 2018.

The converse of this theorem does not hold: if $\mathscr{A}$ is a completely reachable automaton, and even if for every non-empty subset $P \subseteq Q$, there is a product w of words of defect 1 such that $P = Q.w$, the graph $\Gamma_1(\mathscr{A})$ need not be strongly connected. An example was in our DCFS 2016 paper, and a stronger example was found by François Gonze and Raphaël Jungers, DLT 2018.

Now we describe an iterative process for which the graph $\Gamma_1(\mathscr{A})$ serves as the starting point.

The converse of this theorem does not hold: if $\mathscr{A}$ is a completely reachable automaton, and even if for every non-empty subset $P \subseteq Q$, there is a product w of words of defect 1 such that $P = Q.w$, the graph $\Gamma_1(\mathscr{A})$ need not be strongly connected. An example was in our DCFS 2016 paper, and a stronger example was found by François Gonze and Raphaël Jungers, DLT 2018.

Now we describe an iterative process for which the graph $\Gamma_1(\mathscr{A})$ serves as the starting point. The process produces a sequence of graphs $\Gamma_1(\mathscr{A}) \subset \Gamma_2(\mathscr{A}) \subset \cdots \subset \Gamma_k(\mathscr{A})$, where $k < n$.

We add both new states and new edges when passing from $\Gamma_{k-1}(\mathscr{A})$ to $\Gamma_k(\mathscr{A})$.

Graphs $\Gamma_k(\mathscr{A})$

The converse of this theorem does not hold: if $\mathscr{A}$ is a completely reachable automaton, and even if for every non-empty subset $P \subseteq Q$, there is a product w of words of defect 1 such that $P = Q.w$, the graph $\Gamma_1(\mathscr{A})$ need not be strongly connected. An example was in our DCFS 2016 paper, and a stronger example was found by François Gonze and Raphaël Jungers, DLT 2018.

Now we describe an iterative process for which the graph $\Gamma_1(\mathscr{A})$ serves as the starting point. The process produces a sequence of graphs $\Gamma_1(\mathscr{A}) \subset \Gamma_2(\mathscr{A}) \subset \cdots \subset \Gamma_k(\mathscr{A})$, where $k < n$.

We add both new states and new edges when passing from $\Gamma_{k-1}(\mathscr{A})$ to $\Gamma_k(\mathscr{A})$.

For this, we extend the operators $\mathrm{excl}(\_)$ and $\mathrm{dupl}(\_)$ to words with defect > 1: if $\mathscr{A} = \langle Q, \Sigma \rangle$ is a DFA and $w \in \Sigma^*$, we define $\mathrm{excl}(w)$ as the set $Q \backslash Q.w$

The converse of this theorem does not hold: if $\mathscr{A}$ is a completely reachable automaton, and even if for every non-empty subset $P \subseteq Q$, there is a product w of words of defect 1 such that $P = Q.w$, the graph $\Gamma_1(\mathscr{A})$ need not be strongly connected. An example was in our DCFS 2016 paper, and a stronger example was found by François Gonze and Raphaël Jungers, DLT 2018.

Now we describe an iterative process for which the graph $\Gamma_1(\mathscr{A})$ serves as the starting point. The process produces a sequence of graphs $\Gamma_1(\mathscr{A}) \subset \Gamma_2(\mathscr{A}) \subset \cdots \subset \Gamma_k(\mathscr{A})$, where $k < n$.

We add both new states and new edges when passing from $\Gamma_{k-1}(\mathscr{A})$ to $\Gamma_k(\mathscr{A})$.

For this, we extend the operators $\mathrm{excl}(\_)$ and $\mathrm{dupl}(\_)$ to words with defect > 1: if $\mathscr{A} = \langle Q, \Sigma \rangle$ is a DFA and $w \in \Sigma^*$, we define $\mathrm{excl}(w)$ as the set $Q \backslash Q.w$ and $\mathrm{dupl}(w)$ as the set $\{p \in Q \mid p = q_1.w = q_2.w \text{ for some } q_1 \neq q_2\}$.

Constructing $\Gamma_k(\mathscr{A})$

Let $R_1 := Q$ and $J_1 := E_1$.

Let $R_1 := Q$ and $J_1 := E_1$. So, $\Gamma_1(\mathscr{A}) = (R_1, J_1)$.

Let $R_1 := Q$ and $J_1 := E_1$. So, $\Gamma_1(\mathscr{A}) = (R_1, J_1)$.

Suppose that $k > 1$ and the graph $\Gamma_{k-1}(\mathscr{A})$ with the vertex set R_{k-1} and the edge set J_{k-1} has already been defined.

Constructing $\Gamma_k(\mathscr{A})$

Let $R_1 := Q$ and $J_1 := E_1$. So, $\Gamma_1(\mathscr{A}) = (R_1, J_1)$.

Suppose that $k > 1$ and the graph $\Gamma_{k-1}(\mathscr{A})$ with the vertex set R_{k-1} and the edge set J_{k-1} has already been defined.

If $\Gamma_{k-1}(\mathscr{A})$ is strongly connected, we stop with SUCCESS.

Constructing $\Gamma_k(\mathscr{A})$

Let $R_1 := Q$ and $J_1 := E_1$. So, $\Gamma_1(\mathscr{A}) = (R_1, J_1)$.

Suppose that $k > 1$ and the graph $\Gamma_{k-1}(\mathscr{A})$ with the vertex set R_{k-1} and the edge set J_{k-1} has already been defined.

If $\Gamma_{k-1}(\mathscr{A})$ is strongly connected, we stop with SUCCESS.
If $\Gamma_{k-1}(\mathscr{A})$ is not strongly connected, and $|\Delta \cap Q| < k$ for each SCC Δ of $\Gamma_{k-1}(\mathscr{A})$, we stop with FAILURE.

Constructing $\Gamma_k(\mathscr{A})$

Let $R_1 := Q$ and $J_1 := E_1$. So, $\Gamma_1(\mathscr{A}) = (R_1, J_1)$.

Suppose that $k > 1$ and the graph $\Gamma_{k-1}(\mathscr{A})$ with the vertex set R_{k-1} and the edge set J_{k-1} has already been defined.

If $\Gamma_{k-1}(\mathscr{A})$ is strongly connected, we stop with SUCCESS.
If $\Gamma_{k-1}(\mathscr{A})$ is not strongly connected, and $|\Delta \cap Q| < k$ for each SCC Δ of $\Gamma_{k-1}(\mathscr{A})$, we stop with FAILURE.

Otherwise, we let $R_k := R_{k-1} \cup Q_k$ where
$Q_k := \{\Delta \mid \Delta \text{ is a SCC of } \Gamma_{k-1}(\mathscr{A}) \text{ such that } |\Delta \cap Q| \geq k\}$

Constructing $\Gamma_k(\mathscr{A})$

Let $R_1 := Q$ and $J_1 := E_1$. So, $\Gamma_1(\mathscr{A}) = (R_1, J_1)$.

Suppose that $k > 1$ and the graph $\Gamma_{k-1}(\mathscr{A})$ with the vertex set R_{k-1} and the edge set J_{k-1} has already been defined.

If $\Gamma_{k-1}(\mathscr{A})$ is strongly connected, we stop with SUCCESS.
If $\Gamma_{k-1}(\mathscr{A})$ is not strongly connected, and $|\Delta \cap Q| < k$ for each SCC Δ of $\Gamma_{k-1}(\mathscr{A})$, we stop with FAILURE.

Otherwise, we let $R_k := R_{k-1} \cup Q_k$ where
$Q_k := \{\Delta \mid \Delta \text{ is a SCC of } \Gamma_{k-1}(\mathscr{A}) \text{ such that } |\Delta \cap Q| \geq k\}$
and let $J_k := J_{k-1} \cup I_k \cup E_k$ where
$I_k := \{(D, C) \in R_{k-1} \times Q_k \mid D \subset C\}$ (inclusion edges), and
$E_k := \{(C, p) \in Q_k \times Q \mid C \supseteq \mathrm{excl}(w),\ p \in \mathrm{dupl}(w),\ \mathrm{df}(w) = k\}$
(exclusion edges).

Let $R_1 := Q$ and $J_1 := E_1$. So, $\Gamma_1(\mathscr{A}) = (R_1, J_1)$.

Suppose that $k > 1$ and the graph $\Gamma_{k-1}(\mathscr{A})$ with the vertex set R_{k-1} and the edge set J_{k-1} has already been defined.

If $\Gamma_{k-1}(\mathscr{A})$ is strongly connected, we stop with SUCCESS.
If $\Gamma_{k-1}(\mathscr{A})$ is not strongly connected, and $|\Delta \cap Q| < k$ for each SCC Δ of $\Gamma_{k-1}(\mathscr{A})$, we stop with FAILURE.

Otherwise, we let $R_k := R_{k-1} \cup Q_k$ where
$Q_k := \{\Delta \mid \Delta \text{ is a SCC of } \Gamma_{k-1}(\mathscr{A}) \text{ such that } |\Delta \cap Q| \geq k\}$
and let $J_k := J_{k-1} \cup I_k \cup E_k$ where
$I_k := \{(D, C) \in R_{k-1} \times Q_k \mid D \subset C\}$ (inclusion edges), and
$E_k := \{(C, p) \in Q_k \times Q \mid C \supseteq \mathrm{excl}(w),\ p \in \mathrm{dupl}(w),\ \mathrm{df}(w) = k\}$
(exclusion edges).

Now $\Gamma_k(\mathscr{A}) := \big(R_k, J_k\big)$.

Example

Consider the DFA $\mathscr{E}_5$ with 5 states $1, 2, 3, 4, 5$ and 8 input letters $a_{[1]}, a_{[2]}, a_{[3]}, a_{[4]}, a_{[5]}, a_{[1,2]}, a_{[4,5]}, a_{[1,3]}$ whose actions are shown in the following table:

	$a_{[1]}$	$a_{[2]}$	$a_{[3]}$	$a_{[4]}$	$a_{[5]}$	$a_{[1,2]}$	$a_{[4,5]}$	$a_{[1,3]}$
1	2	1	1	1	1	3	1	4
2	2	1	1	2	2	3	1	4
3	3	3	2	3	3	3	2	4
4	4	4	4	5	4	4	3	5
5	5	4	5	5	4	5	3	5

.

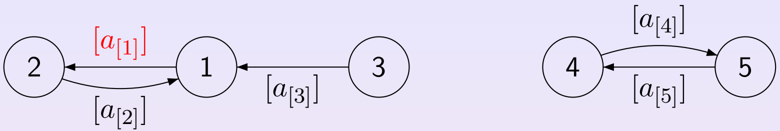
Example

Consider the DFA $\mathscr{E}_5$ with 5 states $1, 2, 3, 4, 5$ and 8 input letters $a_{[1]}, a_{[2]}, a_{[3]}, a_{[4]}, a_{[5]}, a_{[1,2]}, a_{[4,5]}, a_{[1,3]}$ whose actions are shown in the following table:

	$a_{[1]}$	$a_{[2]}$	$a_{[3]}$	$a_{[4]}$	$a_{[5]}$	$a_{[1,2]}$	$a_{[4,5]}$	$a_{[1,3]}$
1	2	1	1	1	1	3	1	4
2	2	1	1	2	2	3	1	4
3	3	3	2	3	3	3	2	4
4	4	4	4	5	4	4	3	5
5	5	4	5	5	4	5	3	5
defect	1	1	1	1	1	2	2	3

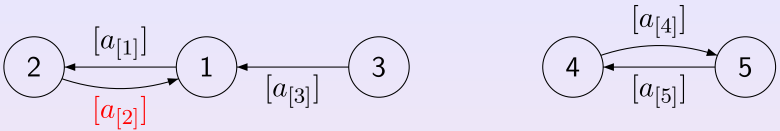
.

Graph $\Gamma_1(\mathscr{E}_5)$



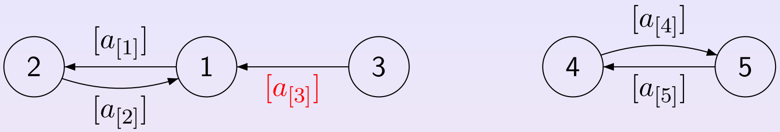
	$a_{[1]}$	$a_{[2]}$	$a_{[3]}$	$a_{[4]}$	$a_{[5]}$	$a_{[1,2]}$	$a_{[4,5]}$	$a_{[1,3]}$
1	2	1	1	1	1	3	1	4
2	2	1	1	2	2	3	1	4
3	3	3	2	3	3	3	2	4
4	4	4	4	5	4	4	3	5
5	5	4	5	5	4	5	3	5

Graph $\Gamma_1(\mathscr{E}_5)$



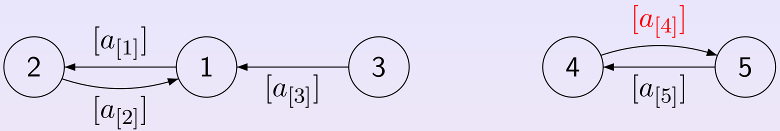
	$a_{[1]}$	$a_{[2]}$	$a_{[3]}$	$a_{[4]}$	$a_{[5]}$	$a_{[1,2]}$	$a_{[4,5]}$	$a_{[1,3]}$
1	2	1	1	1	1	3	1	4
2	2	1	1	2	2	3	1	4
3	3	3	2	3	3	3	2	4
4	4	4	4	5	4	4	3	5
5	5	4	5	5	4	5	3	5

Graph $\Gamma_1(\mathscr{E}_5)$



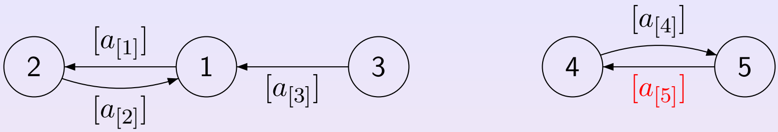
	$a_{[1]}$	$a_{[2]}$	$a_{[3]}$	$a_{[4]}$	$a_{[5]}$	$a_{[1,2]}$	$a_{[4,5]}$	$a_{[1,3]}$
1	2	1	1	1	1	3	1	4
2	2	1	1	2	2	3	1	4
3	3	3	2	3	3	3	2	4
4	4	4	4	5	4	4	3	5
5	5	4	5	5	4	5	3	5

Graph $\Gamma_1(\mathscr{E}_5)$



	$a_{[1]}$	$a_{[2]}$	$a_{[3]}$	$a_{[4]}$	$a_{[5]}$	$a_{[1,2]}$	$a_{[4,5]}$	$a_{[1,3]}$
1	2	1	1	1	1	3	1	4
2	2	1	1	2	2	3	1	4
3	3	3	2	3	3	3	2	4
4	4	4	4	5	4	4	3	5
5	5	4	5	5	4	5	3	5

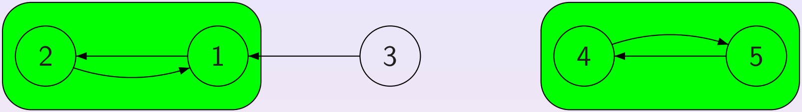
Graph $\Gamma_1(\mathscr{E}_5)$



	$a_{[1]}$	$a_{[2]}$	$a_{[3]}$	$a_{[4]}$	$a_{[5]}$	$a_{[1,2]}$	$a_{[4,5]}$	$a_{[1,3]}$
1	2	1	1	1	1	3	1	4
2	2	1	1	2	2	3	1	4
3	3	3	2	3	3	3	2	4
4	4	4	4	5	4	4	3	5
5	5	4	5	5	4	5	3	5

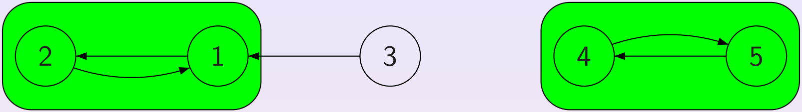
Graph $\Gamma_2(\mathscr{E}_5)$

SCCs of $\Gamma_1(\mathscr{E}_5)$

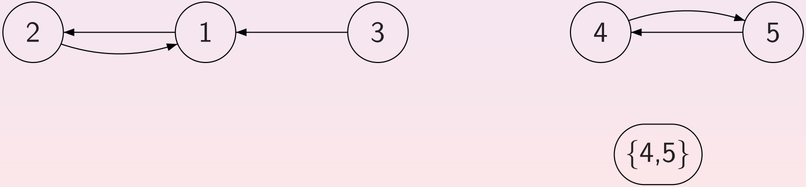


Graph $\Gamma_2(\mathscr{E}_5)$

SCCs of $\Gamma_1(\mathscr{E}_5)$

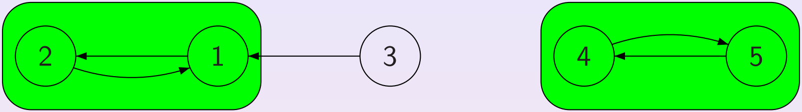


$\Gamma_2(\mathscr{E}_5)$

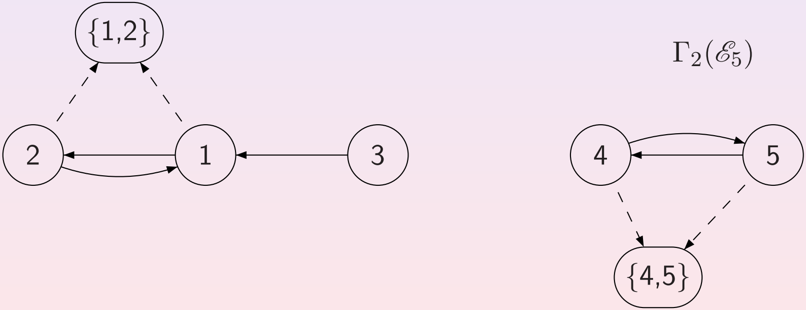


Graph $\Gamma_2(\mathscr{E}_5)$

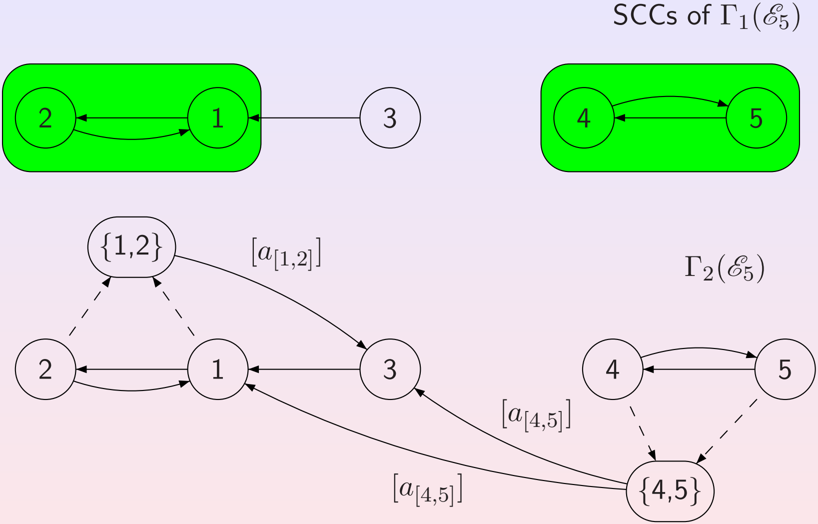
SCCs of $\Gamma_1(\mathscr{E}_5)$



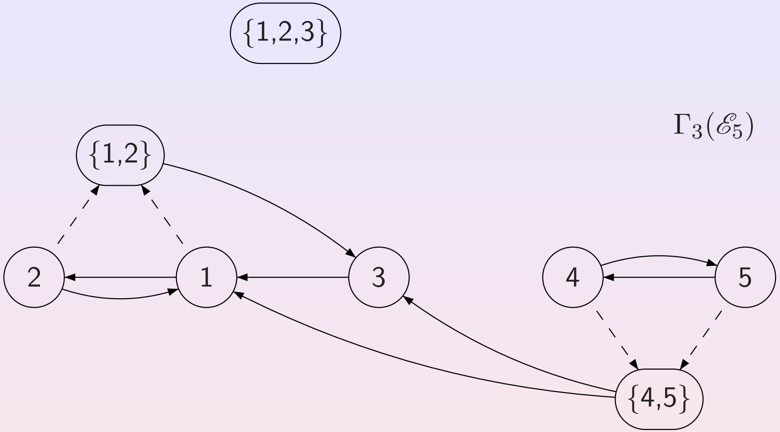
$\Gamma_2(\mathscr{E}_5)$



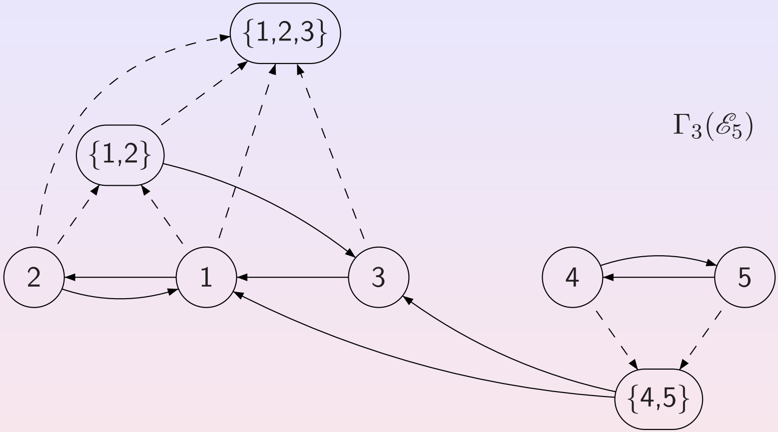
Graph $\Gamma_2(\mathscr{E}_5)$



Graph $\Gamma_3(\mathscr{E}_5)$

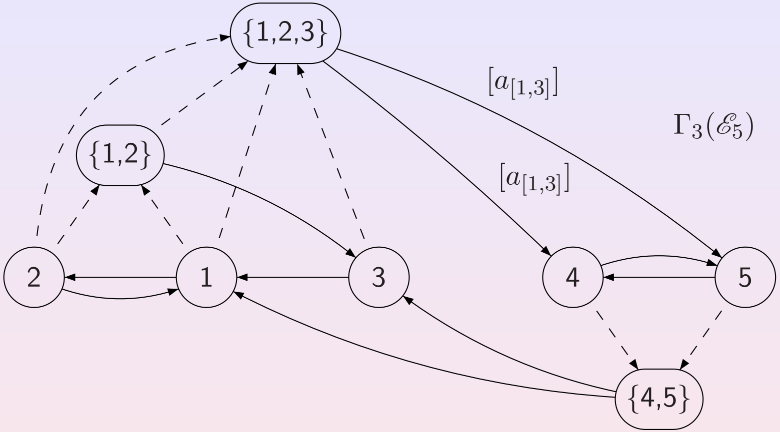


Graph $\Gamma_3(\mathscr{E}_5)$

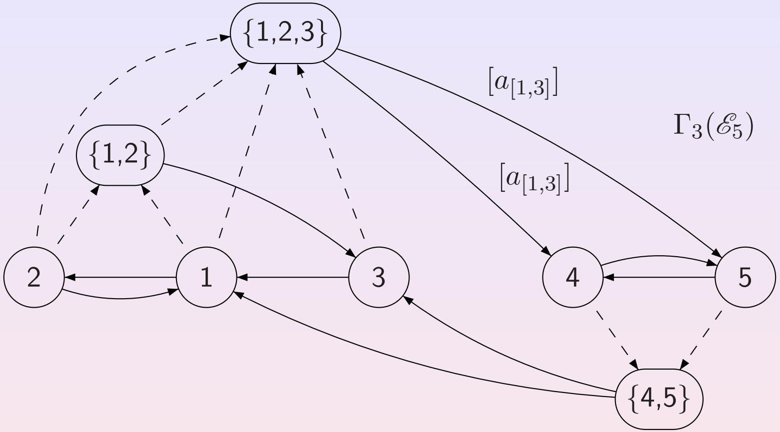


$\Gamma_3(\mathscr{E}_5)$

Graph $\Gamma_3(\mathscr{E}_5)$

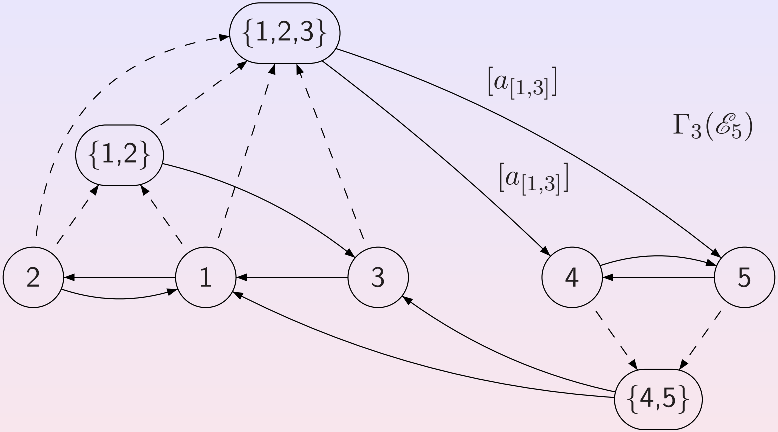


Graph $\Gamma_3(\mathscr{E}_5)$



We see that the graph $\Gamma_3(\mathscr{E}_5)$ is strongly connected

Graph $\Gamma_3(\mathscr{E}_5)$



We see that the graph $\Gamma_3(\mathscr{E}_5)$ is strongly connected, whence our process applied to $\mathscr{E}_5$ stops with SUCCESS.

Clearly, for a DFA $\mathscr{A}$ with n states, constructing the sequence of graphs $\Gamma_1(\mathscr{A})$, $\Gamma_2(\mathscr{A})$, ... must stop after at most $n-1$ steps.

Main Results

Clearly, for a DFA $\mathscr{A}$ with n states, constructing the sequence of graphs $\Gamma_1(\mathscr{A})$, $\Gamma_2(\mathscr{A})$, ... must stop after at most $n-1$ steps.

Theorem 1

If for DFA $\mathscr{A} = (Q, \Sigma)$, the described process stops at step k with SUCCESS (i.e., the graph $\Gamma_k(\mathscr{A})$ is strongly connected), then $\mathscr{A}$ is completely reachable; more precisely, for every non-empty subset $P \subseteq Q$, there is a product w of words of defect at most k such that $P = Q.w$.

Clearly, for a DFA $\mathscr{A}$ with n states, constructing the sequence of graphs $\Gamma_1(\mathscr{A})$, $\Gamma_2(\mathscr{A})$, $\ldots$ must stop after at most $n-1$ steps.

Theorem 1

If for DFA $\mathscr{A} = (Q, \Sigma)$, the described process stops at step k with SUCCESS (i.e., the graph $\Gamma_k(\mathscr{A})$ is strongly connected), then $\mathscr{A}$ is completely reachable; more precisely, for every non-empty subset $P \subseteq Q$, there is a product w of words of defect at most k such that $P = Q.w$.

Theorem 2

If for DFA $\mathscr{A} = (Q, \Sigma)$, the described process stops at step k with FAILURE, then $\mathscr{A}$ is not completely reachable; more precisely, some subset in Q with at least $|Q| - k$ states is not reachable in $\mathscr{A}$.

Characterization

Let $\Gamma(\mathscr{A})$ stand for the graph $\Gamma_k(\mathscr{A})$ at which our process stops (with either of the two possible outcomes).

Characterization

Let $\Gamma(\mathscr{A})$ stand for the graph $\Gamma_k(\mathscr{A})$ at which our process stops (with either of the two possible outcomes).

Combining Theorems 1 and 2, we arrive at the following characterization of completely reachable automata:

Let $\Gamma(\mathscr{A})$ stand for the graph $\Gamma_k(\mathscr{A})$ at which our process stops (with either of the two possible outcomes).

Combining Theorems 1 and 2, we arrive at the following characterization of completely reachable automata:

Theorem 3

A DFA $\mathscr{A}$ is completely reachable if and only if the graph $\Gamma(\mathscr{A})$ is strongly connected.

Complexity

It is still open whether or not complete reachability of a DFA can be recognized in polynomial time.

Complexity

It is still open whether or not complete reachability of a DFA can be recognized in polynomial time.

The size of the graph $\Gamma(\mathscr{A})$ is polynomial in the size of $\mathscr{A}$, but it is far from being obvious that $\Gamma(\mathscr{A})$ can always be constructed in polynomial time.

It is still open whether or not complete reachability of a DFA can be recognized in polynomial time.

The size of the graph $\Gamma(\mathscr{A})$ is polynomial in the size of $\mathscr{A}$, but it is far from being obvious that $\Gamma(\mathscr{A})$ can always be constructed in polynomial time.

Indeed, to construct the graph $\Gamma_1(\mathscr{A})$ one must (in principle) analyse all transformations caused by words of defect 1 w.r.t. $\mathscr{A}$, and the number of such transformations may reach $\binom{n}{2}n!$.

Complexity

It is still open whether or not complete reachability of a DFA can be recognized in polynomial time.

The size of the graph $\Gamma(\mathscr{A})$ is polynomial in the size of $\mathscr{A}$, but it is far from being obvious that $\Gamma(\mathscr{A})$ can always be constructed in polynomial time.

Indeed, to construct the graph $\Gamma_1(\mathscr{A})$ one must (in principle) analyse all transformations caused by words of defect 1 w.r.t. $\mathscr{A}$, and the number of such transformations may reach $\binom{n}{2}n!$.

Similarly, to construct the graph $\Gamma_2(\mathscr{A})$ one must (in principle) analyse all transformations caused by words of defect 2, etc.

It is still open whether or not complete reachability of a DFA can be recognized in polynomial time.

The size of the graph $\Gamma(\mathscr{A})$ is polynomial in the size of $\mathscr{A}$, but it is far from being obvious that $\Gamma(\mathscr{A})$ can always be constructed in polynomial time.

Indeed, to construct the graph $\Gamma_1(\mathscr{A})$ one must (in principle) analyse all transformations caused by words of defect 1 w.r.t. $\mathscr{A}$, and the number of such transformations may reach $\binom{n}{2}n!$.

Similarly, to construct the graph $\Gamma_2(\mathscr{A})$ one must (in principle) analyse all transformations caused by words of defect 2, etc.

François Gonze and Raphaël Jungers (DLT 2018) developed a polynomial algorithm for constructing the graph $\Gamma_1(\mathscr{A})$

Complexity, continued

David Casas has managed to extend the approach of Gonze and Jungers to master a polynomial algorithm for constructing the graph $\Gamma(\mathscr{A})$ provided that all letters of $\mathscr{A}$ have defect at most 1.

Complexity, continued

David Casas has managed to extend the approach of Gonze and Jungers to master a polynomial algorithm for constructing the graph $\Gamma(\mathscr{A})$ provided that all letters of $\mathscr{A}$ have defect at most 1.

Corollary

Complete reachability of binary DFAs is polynomially decidable.

Complexity, continued

David Casas has managed to extend the approach of Gonze and Jungers to master a polynomial algorithm for constructing the graph $\Gamma(\mathscr{A})$ provided that all letters of $\mathscr{A}$ have defect at most 1.

Corollary

Complete reachability of binary DFAs is polynomially decidable.

The following polynomiality conjecture is a weaker version of a conjecture suggested by Henk Don: there exists a constant c such that in every completely reachable automaton with n states, each non-empty subset can be reached by a word of length n^c.

Complexity, continued

David Casas has managed to extend the approach of Gonze and Jungers to master a polynomial algorithm for constructing the graph $\Gamma(\mathscr{A})$ provided that all letters of $\mathscr{A}$ have defect at most 1.

Corollary

Complete reachability of binary DFAs is polynomially decidable.

The following polynomiality conjecture is a weaker version of a conjecture suggested by Henk Don: there exists a constant c such that in every completely reachable automaton with n states, each non-empty subset can be reached by a word of length n^c. If this conjecture holds, then Theorem 3 implies that the problem of whether a given DFA is completely reachable lies in NP.

Complexity, continued

David Casas has managed to extend the approach of Gonze and Jungers to master a polynomial algorithm for constructing the graph $\Gamma(\mathscr{A})$ provided that all letters of $\mathscr{A}$ have defect at most 1.

Corollary

Complete reachability of binary DFAs is polynomially decidable.

The following polynomiality conjecture is a weaker version of a conjecture suggested by Henk Don: there exists a constant c such that in every completely reachable automaton with n states, each non-empty subset can be reached by a word of length n^c. If this conjecture holds, then Theorem 3 implies that the problem of whether a given DFA is completely reachable lies in NP.

It is known that there is no constant C such that in every DFA with n states (not necessarily completely reachable!), each reachable subset can be reached by a word of length n^C.

Synchronization

Recall that we motivated our interest in completely reachable automata via the Černý conjecture, viewing complete reachability as a stronger form of synchronization.

Synchronization

Recall that we motivated our interest in completely reachable automata via the Černý conjecture, viewing complete reachability as a stronger form of synchronization.

What is the reset threshold of a completely reachable automaton?

Synchronization

Recall that we motivated our interest in completely reachable automata via the Černý conjecture, viewing complete reachability as a stronger form of synchronization.

What is the reset threshold of a completely reachable automaton? We have only a partial result:

Proposition

A completely reachable automaton with n states has a reset word of length at most $\frac{7}{48}n^3 + O(n^2)$.

Synchronization

Recall that we motivated our interest in completely reachable automata via the Černý conjecture, viewing complete reachability as a stronger form of synchronization.

What is the reset threshold of a completely reachable automaton? We have only a partial result:

Proposition

A completely reachable automaton with n states has a reset word of length at most $\frac{7}{48}n^3 + O(n^2)$.

$\frac{7}{48} \approx 0.1458333$ improves on the best bound known for general synchronizing automata (with the leading coefficient ≈ 0.1654).

Recall that we motivated our interest in completely reachable automata via the Černý conjecture, viewing complete reachability as a stronger form of synchronization.

What is the reset threshold of a completely reachable automaton? We have only a partial result:

Proposition

A completely reachable automaton with n states has a reset word of length at most $\frac{7}{48}n^3 + O(n^2)$.

$\frac{7}{48} \approx 0.1458333$ improves on the best bound known for general synchronizing automata (with the leading coefficient ≈ 0.1654). Still, we fell short to get a quadratic upper bound so far.

Happy Birthday Werner!!

Wir gratulieren herzlich zum Geburtstag und wünschen alles Gute!