Primitive Digraphs, Markov Chains and Synchronizing Automata

Dedicated to Stuart Margolis on the Occasion of His 60th Birthday

Mikhail Volkov

Ural Federal University, Ekaterinburg, Russia

June 13th, 2013

Definitions and Terminology

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

To simplify notation we often write $q \,.\, w$ for $\delta(q, w)$ and $P \,.\, w$ for $\{\delta(q, w) \mid q \in P\}$.

$\mathscr{A}$ is called synchronizing if there is a word $w \in \Sigma^*$ whose action resets $\mathscr{A}$, that is, leaves $\mathscr{A}$ in one particular state no matter at which state in Q it started: $q \,.\, w = q' \,.\, w$ for all $q, q' \in Q$. In short, $|Q \,.\, w| = 1$.

Any w with this property is a reset word for $\mathscr{A}$.

Other names:

- for automata: directable, cofinal, collapsible, etc;
- for words: directing, recurrent, synchronizing, etc.

Definitions and Terminology

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

To simplify notation we often write $q \,.\, w$ for $\delta(q, w)$ and $P \,.\, w$ for $\{\delta(q, w) \mid q \in P\}$.

$\mathscr{A}$ is called synchronizing if there is a word $w \in \Sigma^*$ whose action resets $\mathscr{A}$, that is, leaves $\mathscr{A}$ in one particular state no matter at which state in Q it started: $q \,.\, w = q' \,.\, w$ for all $q, q' \in Q$. In short, $|Q \,.\, w| = 1$.

Any w with this property is a reset word for $\mathscr{A}$.

Other names:

- for automata: directable, cofinal, collapsible, etc;
- for words: directing, recurrent, synchronizing, etc.

Definitions and Terminology

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

To simplify notation we often write $q \,.\, w$ for $\delta(q, w)$ and $P \,.\, w$ for $\{\delta(q, w) \mid q \in P\}$.

$\mathscr{A}$ is called synchronizing if there is a word $w \in \Sigma^*$ whose action resets $\mathscr{A}$, that is, leaves $\mathscr{A}$ in one particular state no matter at which state in Q it started: $q \,.\, w = q' \,.\, w$ for all $q, q' \in Q$.
In short, $|Q \,.\, w| = 1$.

Any w with this property is a reset word for $\mathscr{A}$.

Other names:

- for automata: directable, cofinal, collapsible, etc;
- for words: directing, recurrent, synchronizing, etc.

Definitions and Terminology

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

To simplify notation we often write $q \,.\, w$ for $\delta(q, w)$ and $P \,.\, w$ for $\{\delta(q, w) \mid q \in P\}$.

$\mathscr{A}$ is called synchronizing if there is a word $w \in \Sigma^*$ whose action resets $\mathscr{A}$, that is, leaves $\mathscr{A}$ in one particular state no matter at which state in Q it started: $q \,.\, w = q' \,.\, w$ for all $q, q' \in Q$.

In short, $|Q \,.\, w| = 1$.

Any w with this property is a reset word for $\mathscr{A}$.

Other names:

- for automata: directable, cofinal, collapsible, etc;
- for words: directing, recurrent, synchronizing, etc.

Definitions and Terminology

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

To simplify notation we often write $q \,.\, w$ for $\delta(q, w)$ and $P \,.\, w$ for $\{\delta(q, w) \mid q \in P\}$.
$\mathscr{A}$ is called synchronizing if there is a word $w \in \Sigma^*$ whose action resets $\mathscr{A}$, that is, leaves $\mathscr{A}$ in one particular state no matter at which state in Q it started: $q \,.\, w = q' \,.\, w$ for all $q, q' \in Q$.
In short, $|Q \,.\, w| = 1$.

Any w with this property is a reset word for $\mathscr{A}$.

Other names:

- for automata: directable, cofinal, collapsible, etc;
- for words: directing, recurrent, synchronizing, etc.

Definitions and Terminology

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

To simplify notation we often write $q \,.\, w$ for $\delta(q, w)$ and $P \,.\, w$ for $\{\delta(q, w) \mid q \in P\}$.
$\mathscr{A}$ is called synchronizing if there is a word $w \in \Sigma^*$ whose action resets $\mathscr{A}$, that is, leaves $\mathscr{A}$ in one particular state no matter at which state in Q it started: $q \,.\, w = q' \,.\, w$ for all $q, q' \in Q$.
In short, $|Q \,.\, w| = 1$.

Any w with this property is a reset word for $\mathscr{A}$.

Other names:

- for automata: directable, cofinal, collapsible, etc;
- for words: directing, recurrent, synchronizing, etc.

Definitions and Terminology

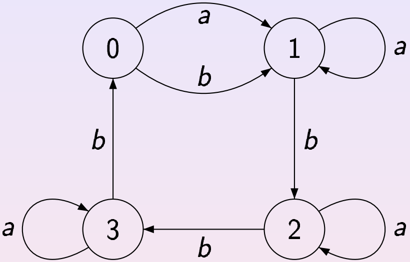
We consider complete deterministic finite automata (DFA) $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ where Q stands for the state set, Σ is the input alphabet, and $\delta : Q \times \Sigma \to Q$ is a (total) transition function.

To simplify notation we often write $q \, . \, w$ for $\delta(q, w)$ and $P \, . \, w$ for $\{\delta(q, w) \mid q \in P\}$.
$\mathscr{A}$ is called synchronizing if there is a word $w \in \Sigma^*$ whose action resets $\mathscr{A}$, that is, leaves $\mathscr{A}$ in one particular state no matter at which state in Q it started: $q \, . \, w = q' \, . \, w$ for all $q, q' \in Q$.
In short, $|Q \, . \, w| = 1$.

Any w with this property is a reset word for $\mathscr{A}$.

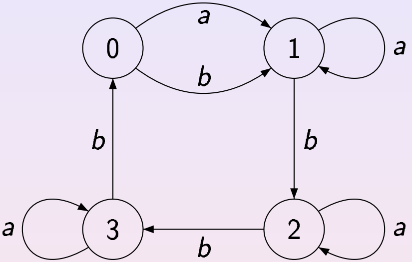
Other names:

- for automata: directable, cofinal, collapsible, etc;
- for words: directing, recurrent, synchronizing, etc.



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

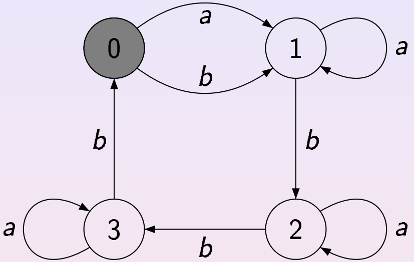
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

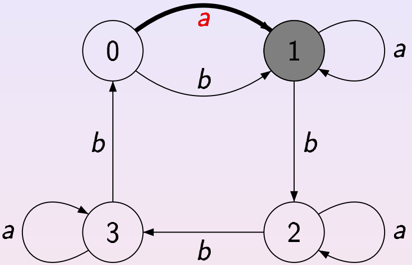
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

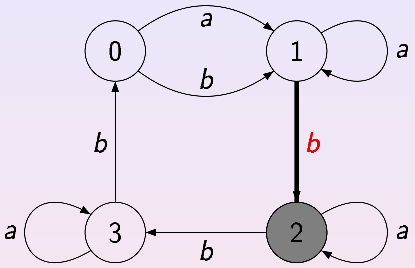
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

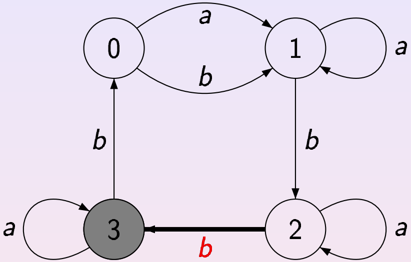
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

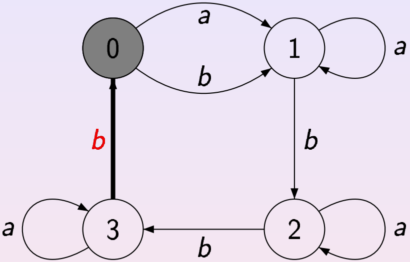
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

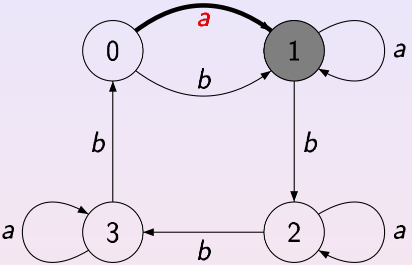
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

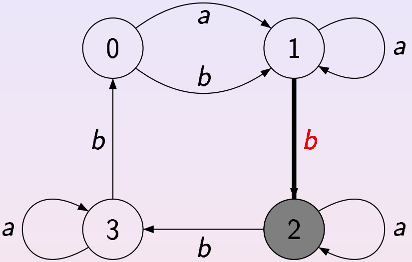
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

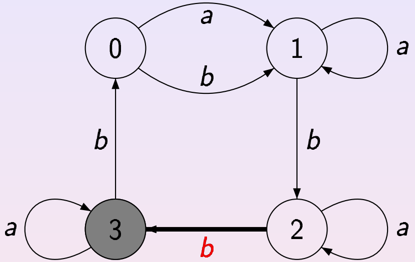
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

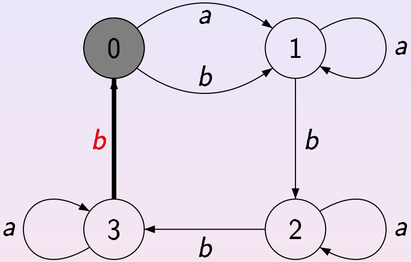
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

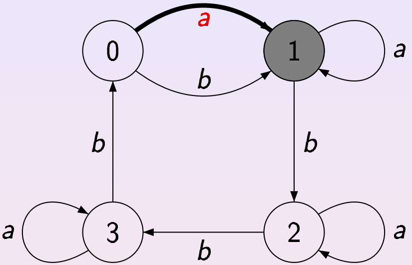
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

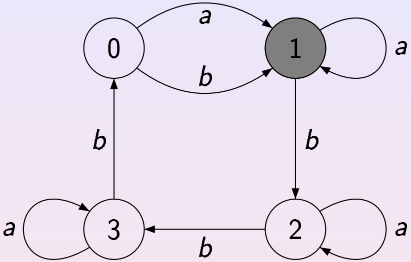
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

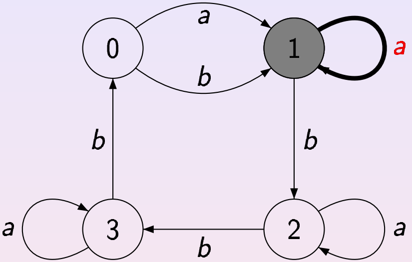
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

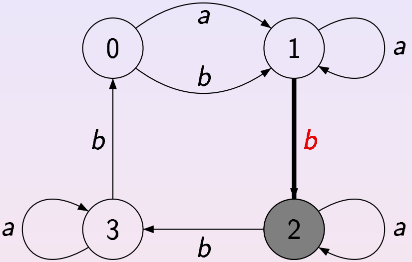
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

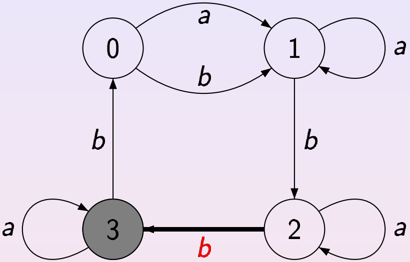
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

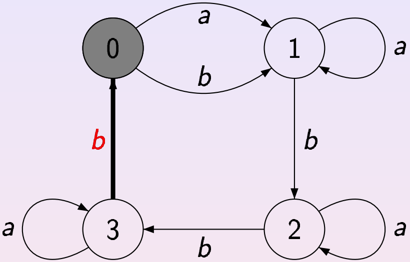
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

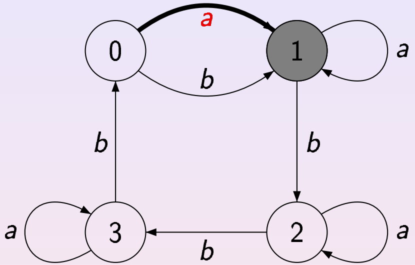
An Example



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

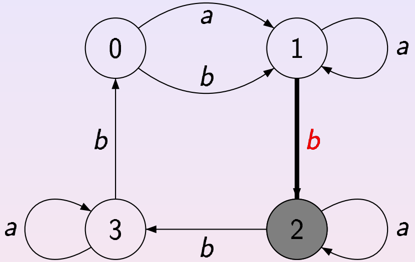
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

An Example



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

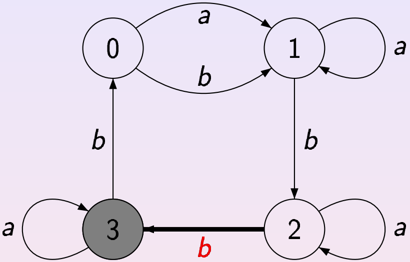
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.



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

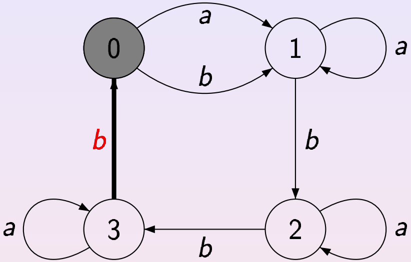
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

An Example



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

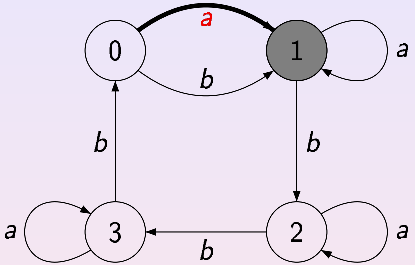
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

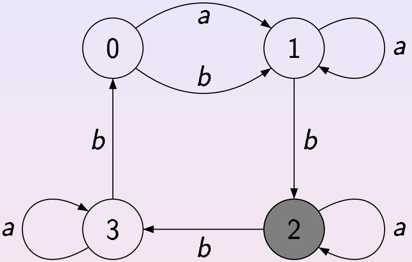
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

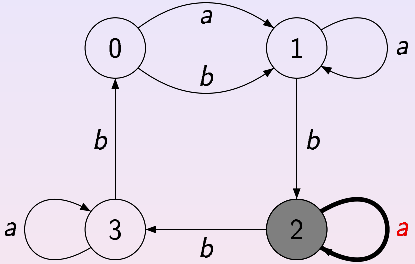
An Example



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

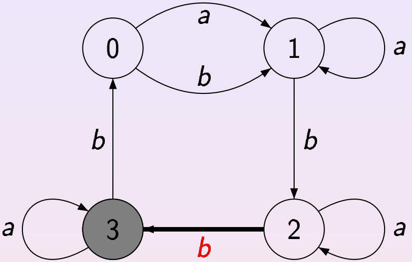
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

An Example



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

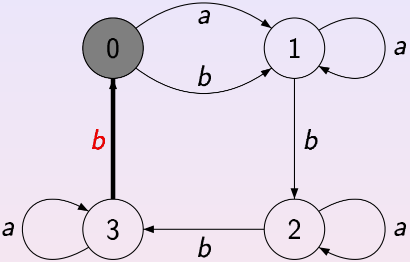
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

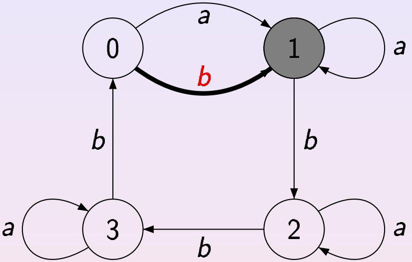
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

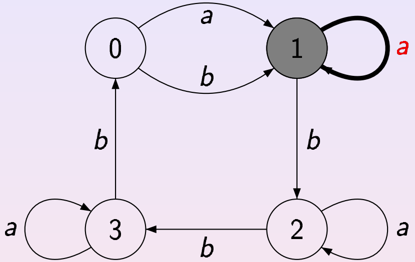
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

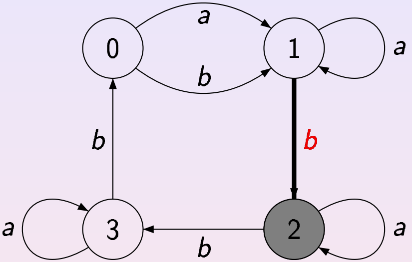
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

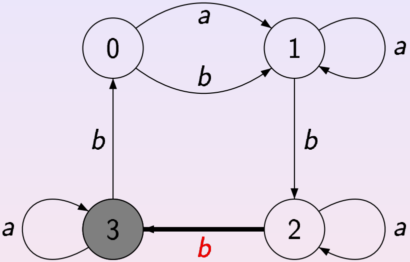
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

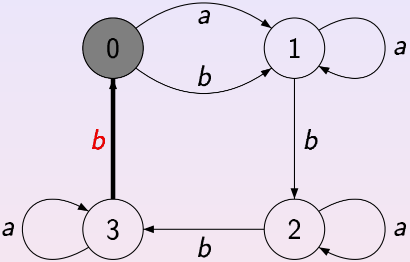
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

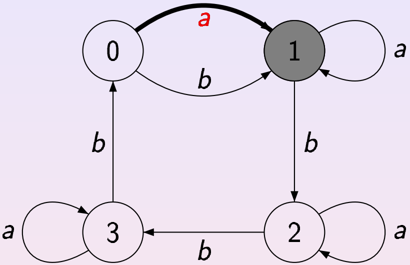
An Example



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

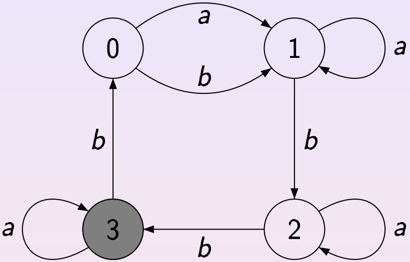
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

An Example



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

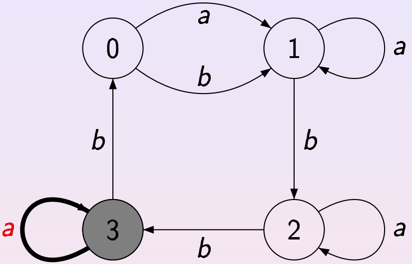
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

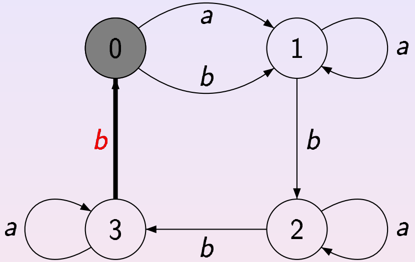
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

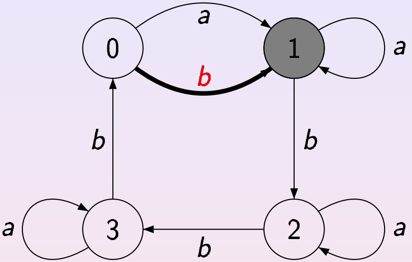
An Example



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

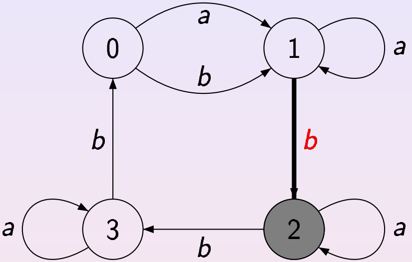
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

An Example



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

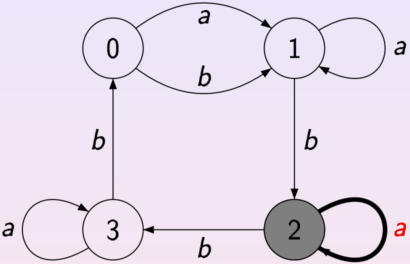
In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

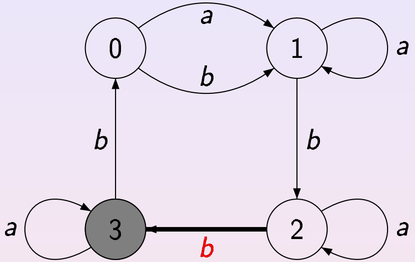
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

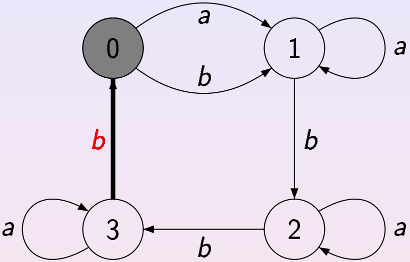
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

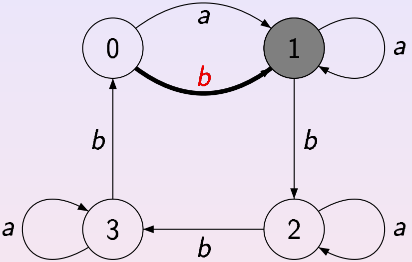
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

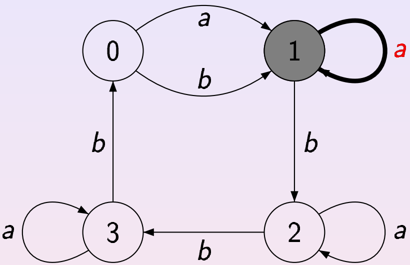
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

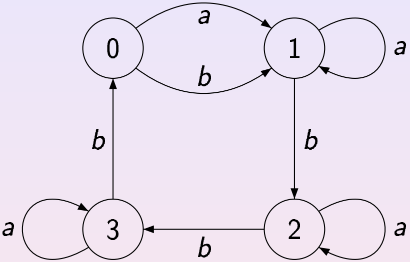
An Example



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

In fact, this is the reset word of minimum length for the automaton whence the reset threshold of the automaton is 9.

An Example



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

Černý's Paper

The notion was formalized in a paper by Jan Černý (Poznámka k homogénnym eksperimentom s konečnými automatami, Matematicko-fyzikalny Časopis Slovensk. Akad. Vied 14, no.3 (1964) 208–216 [in Slovak]) though implicitly it had been around since at least 1956.

The idea of synchronization is pretty natural and of obvious importance: we aim to restore control over a device whose current state is not known.

Think of a satellite which loops around the Moon and cannot be controlled from the Earth while "behind" the Moon (Černý's original motivation).

Independently, the same notion was discovered in coding theory by Shimon Even (Test for synchronizability of finite automata and variable length codes, IEEE Trans. Inform. Theory 10 (1964) 185–189). The name synchronizing seems to have originated from Even's paper.

Černý's Paper

The notion was formalized in a paper by Jan Černý (Poznámka k homogénnym eksperimentom s konečnými automatami, Matematicko-fyzikalny Časopis Slovensk. Akad. Vied 14, no.3 (1964) 208–216 [in Slovak]) though implicitly it had been around since at least 1956.

The idea of synchronization is pretty natural and of obvious importance: we aim to restore control over a device whose current state is not known.

Think of a satellite which loops around the Moon and cannot be controlled from the Earth while "behind" the Moon (Černý's original motivation).

Independently, the same notion was discovered in coding theory by Shimon Even (Test for synchronizability of finite automata and variable length codes, IEEE Trans. Inform. Theory 10 (1964) 185–189). The name synchronizing seems to have originated from Even's paper.

Černý's Paper

The notion was formalized in a paper by Jan Černý (Poznámka k homogénnym eksperimentom s konečnými automatami, Matematicko-fyzikalny Časopis Slovensk. Akad. Vied 14, no.3 (1964) 208–216 [in Slovak]) though implicitly it had been around since at least 1956.

The idea of synchronization is pretty natural and of obvious importance: we aim to restore control over a device whose current state is not known.

Think of a satellite which loops around the Moon and cannot be controlled from the Earth while "behind" the Moon (Černý's original motivation).

Independently, the same notion was discovered in coding theory by Shimon Even (Test for synchronizability of finite automata and variable length codes, IEEE Trans. Inform. Theory 10 (1964) 185–189). The name synchronizing seems to have originated from Even's paper.

Černý's Paper

The notion was formalized in a paper by Jan Černý (Poznámka k homogénnym eksperimentom s konečnými automatami, Matematicko-fyzikalny Časopis Slovensk. Akad. Vied 14, no.3 (1964) 208–216 [in Slovak]) though implicitly it had been around since at least 1956.

The idea of synchronization is pretty natural and of obvious importance: we aim to restore control over a device whose current state is not known.

Think of a satellite which loops around the Moon and cannot be controlled from the Earth while "behind" the Moon (Černý's original motivation).

Independently, the same notion was discovered in coding theory by Shimon Even (Test for synchronizability of finite automata and variable length codes, IEEE Trans. Inform. Theory 10 (1964) 185–189). The name synchronizing seems to have originated from Even's paper.

Černý's Paper

The notion was formalized in a paper by Jan Černý (Poznámka k homogénnym eksperimentom s konečnými automatami, Matematicko-fyzikalny Časopis Slovensk. Akad. Vied 14, no.3 (1964) 208–216 [in Slovak]) though implicitly it had been around since at least 1956.

The idea of synchronization is pretty natural and of obvious importance: we aim to restore control over a device whose current state is not known.

Think of a satellite which loops around the Moon and cannot be controlled from the Earth while "behind" the Moon (Černý's original motivation).

Independently, the same notion was discovered in coding theory by Shimon Even (Test for synchronizability of finite automata and variable length codes, IEEE Trans. Inform. Theory 10 (1964) 185–189). The name synchronizing seems to have originated from Even's paper.

Černý Conjecture

The Černý conjecture is the claim that every synchronizing automaton with n states possesses a reset word of length $(n-1)^2$. The validity of the conjecture is main open problem of the area and arguably one of the most long-standing open problems in combinatorial theory of finite automata.

Define the Černý function $C(n)$ as the maximum reset threshold for synchronizing automata with n states. In terms of this function, our current knowledge can be summarized in one line:

The Černý conjecture thus claims that in fact $C(n) = (n-1)^2$.

Černý Conjecture

The Černý conjecture is the claim that every synchronizing automaton with n states possesses a reset word of length $(n-1)^2$. The validity of the conjecture is main open problem of the area and arguably one of the most long-standing open problems in combinatorial theory of finite automata.

Define the Černý function $C(n)$ as the maximum reset threshold for synchronizing automata with n states. In terms of this function, our current knowledge can be summarized in one line:

The Černý conjecture thus claims that in fact $C(n) = (n-1)^2$.

The Černý conjecture is the claim that every synchronizing automaton with n states possesses a reset word of length $(n-1)^2$. The validity of the conjecture is main open problem of the area and arguably one of the most long-standing open problems in combinatorial theory of finite automata.

Define the Černý function $C(n)$ as the maximum reset threshold for synchronizing automata with n states. In terms of this function, our current knowledge can be summarized in one line:

The Černý conjecture thus claims that in fact $C(n) = (n-1)^2$.

Černý Conjecture

The Černý conjecture is the claim that every synchronizing automaton with n states possesses a reset word of length $(n-1)^2$. The validity of the conjecture is main open problem of the area and arguably one of the most long-standing open problems in combinatorial theory of finite automata.

Define the Černý function $C(n)$ as the maximum reset threshold for synchronizing automata with n states. In terms of this function, our current knowledge can be summarized in one line:

The Černý conjecture thus claims that in fact $C(n) = (n-1)^2$.

Černý Conjecture

The Černý conjecture is the claim that every synchronizing automaton with n states possesses a reset word of length $(n-1)^2$. The validity of the conjecture is main open problem of the area and arguably one of the most long-standing open problems in combinatorial theory of finite automata.

Define the Černý function $C(n)$ as the maximum reset threshold for synchronizing automata with n states. In terms of this function, our current knowledge can be summarized in one line:

$$(\text{Černý, } 1964) \quad (n-1)^2 \leq C(n)$$

The Černý conjecture thus claims that in fact $C(n) = (n-1)^2$.

Černý Conjecture

The Černý conjecture is the claim that every synchronizing automaton with n states possesses a reset word of length $(n-1)^2$. The validity of the conjecture is main open problem of the area and arguably one of the most long-standing open problems in combinatorial theory of finite automata.

Define the Černý function $C(n)$ as the maximum reset threshold for synchronizing automata with n states. In terms of this function, our current knowledge can be summarized in one line:

$$\text{(Černý, 1964)}\quad (n-1)^2 \leq C(n) \leq \frac{n^3-n}{6}\quad \text{(Pin–Frankl, 1983)}.$$

The Černý conjecture thus claims that in fact $C(n) = (n-1)^2$.

Černý Conjecture

The Černý conjecture is the claim that every synchronizing automaton with n states possesses a reset word of length $(n-1)^2$. The validity of the conjecture is main open problem of the area and arguably one of the most long-standing open problems in combinatorial theory of finite automata.

Define the Černý function $C(n)$ as the maximum reset threshold for synchronizing automata with n states. In terms of this function, our current knowledge can be summarized in one line:

$$\text{(Černý, 1964)} \quad (n-1)^2 \le C(n) \le \frac{n^3-n}{6} \quad \text{(Pin–Frankl, 1983)}.$$

The Černý conjecture thus claims that in fact $C(n) = (n-1)^2$.

Why so Difficult?

Why is the problem so surprisingly difficult?

One of the reasons: "slowly" synchronizing automata turn out to be extremely rare. Only one infinite series of n-state synchronizing automata with reset threshold $(n-1)^2$ is known (due to Černý, 1964), with a few (actually, 8) sporadic examples for $n \leq 6$.

Why so Difficult?

Why is the problem so surprisingly difficult?

One of the reasons: "slowly" synchronizing automata turn out to be extremely rare. Only one infinite series of n-state synchronizing automata with reset threshold $(n-1)^2$ is known (due to Černý, 1964), with a few (actually, 8) sporadic examples for $n \leq 6$.

Why so Difficult?

Why is the problem so surprisingly difficult?

One of the reasons: “slowly” synchronizing automata turn out to be extremely rare. Only one infinite series of n-state synchronizing automata with reset threshold $(n-1)^2$ is known (due to Černý, 1964), with a few (actually, 8) sporadic examples for $n \leq 6$.

Episode IV: A New Hope

In 2009/10, Vladimir Gusev, at that time a PhD student of mine, has performed a massive series of experiments searching exhaustively through automata with a modest number of states in order to find new examples of “slowly” synchronizing automata. The next tables present the distribution of non-isomorphic synchronizing automata with 8 and 9 states and 2 letters with respect to their reset thresholds.

Episode IV: A New Hope

In 2009/10, Vladimir Gusev, at that time a PhD student of mine, has performed a massive series of experiments searching exhaustively through automata with a modest number of states in order to find new examples of “slowly” synchronizing automata. The next tables present the distribution of non-isomorphic synchronizing automata with 8 and 9 states and 2 letters with respect to their reset thresholds.

Episode IV: A New Hope

In 2009/10, Vladimir Gusev, at that time a PhD student of mine, has performed a massive series of experiments searching exhaustively through automata with a modest number of states in order to find new examples of "slowly" synchronizing automata. The next tables present the distribution of non-isomorphic synchronizing automata with 8 and 9 states and 2 letters with respect to their reset thresholds.

8 states:

Reset threshold	49	48	47	46	45	44	43	42	41	40
# of automata	1	0	0	0	0	1	1	3	1	5

Episode IV: A New Hope

In 2009/10, Vladimir Gusev, at that time a PhD student of mine, has performed a massive series of experiments searching exhaustively through automata with a modest number of states in order to find new examples of “slowly” synchronizing automata. The next tables present the distribution of non-isomorphic synchronizing automata with 8 and 9 states and 2 letters with respect to their reset thresholds.

8 states:

Reset threshold	49	48	47	46	45	44	43	42	41	40
# of automata	1	0	0	0	0	1	1	3	1	5

Episode IV: A New Hope

In 2009/10, Vladimir Gusev, at that time a PhD student of mine, has performed a massive series of experiments searching exhaustively through automata with a modest number of states in order to find new examples of "slowly" synchronizing automata. The next tables present the distribution of non-isomorphic synchronizing automata with 8 and 9 states and 2 letters with respect to their reset thresholds.

8 states:

Reset threshold	49	48	47	46	45	44	43	42	41	40
# of automata	1	0	0	0	0	1	1	3	1	5

9 states:

Reset threshold	64	63	62	61	60	59	58	57	56	55
# of automata	1	0	0	0	0	0	1	2	3	0

Reset threshold	54	53	52	51
# of automata	0	0	4	4

Episode IV: A New Hope

In 2009/10, Vladimir Gusev, at that time a PhD student of mine, has performed a massive series of experiments searching exhaustively through automata with a modest number of states in order to find new examples of "slowly" synchronizing automata. The next tables present the distribution of non-isomorphic synchronizing automata with 8 and 9 states and 2 letters with respect to their reset thresholds.

8 states:

Reset threshold	49	48	47	46	45	44	43	42	41	40
# of automata	1	0	0	0	0	1	1	3	1	5

9 states:

Reset threshold	64	63	62	61	60	59	58	57	56	55
# of automata	1	0	0	0	0	0	1	2	3	0

Reset threshold	54	53	52	51
# of automata	0	0	4	4

Advantage of Being Old

Thus, the pattern is:

$(n-1)^2$ the first gap the "island" the second gap

The second gap first appears at 9 states and grows rather regularly with the number of states. The size of the island depends only on the parity of the number of states.

The very same pattern appears in the distribution of exponents of non-negative matrices.

Advantage of Being Old

Thus, the pattern is:

$(n-1)^2$ the first gap the "island" the second gap

The second gap first appears at 9 states and grows rather regularly with the number of states. The size of the island depends only on the parity of the number of states.

The very same pattern appears in the distribution of exponents of non-negative matrices.

Advantage of Being Old

Thus, the pattern is:

$(n-1)^2$ the first gap the "island" the second gap

The second gap first appears at 9 states and grows rather regularly with the number of states. The size of the island depends only on the parity of the number of states.

The very same pattern appears in the distribution of exponents of non-negative matrices.

Exponents of Non-negative Matrices

A non-negative matrix A is said to be primitive if some power A^k is positive. The minimum k with this property is called the exponent of A, denoted $\exp A$.

Helmut Wielandt proved in 1950 that for any primitive $n \times n$-matrix A, one has $\exp A \leq n^2 - 2n + 2 = (n-1)^2 + 1$, and this bound is tight. Possible exponents of $n \times n$-matrices were intensively studied in the 1960s, and it was discovered that two extreme values are each attained by a unique matrix, then there is a gap followed by an island followed by another gap. The sizes of the gaps and of the island perfectly match the sizes of the gaps and of the islands in possible reset thresholds of synchronizing automata with n states — basically one has the same picture shifted by 1. Clearly, this cannot be a mere coincidence.

Exponents of Non-negative Matrices

A non-negative matrix A is said to be primitive if some power A^k is positive. The minimum k with this property is called the exponent of A, denoted $\exp A$.

Helmut Wielandt proved in 1950 that for any primitive $n \times n$-matrix A, one has $\exp A \leq n^2 - 2n + 2 = (n-1)^2 + 1$, and this bound is tight. Possible exponents of $n \times n$-matrices were intensively studied in the 1960s, and it was discovered that two extreme values are each attained by a unique matrix, then there is a gap followed by an island followed by another gap. The sizes of the gaps and of the island perfectly match the sizes of the gaps and of the islands in possible reset thresholds of synchronizing automata with n states — basically one has the same picture shifted by 1. Clearly, this cannot be a mere coincidence.

Exponents of Non-negative Matrices

A non-negative matrix A is said to be primitive if some power A^k is positive. The minimum k with this property is called the exponent of A, denoted $\exp A$.

Helmut Wielandt proved in 1950 that for any primitive $n \times n$-matrix A, one has $\exp A \leq n^2 - 2n + 2 = (n-1)^2 + 1$, and this bound is tight. Possible exponents of $n \times n$-matrices were intensively studied in the 1960s, and it was discovered that two extreme values are each attained by a unique matrix, then there is a gap followed by an island followed by another gap. The sizes of the gaps and of the island perfectly match the sizes of the gaps and of the islands in possible reset thresholds of synchronizing automata with n states — basically one has the same picture shifted by 1. Clearly, this cannot be a mere coincidence.

Exponents of Non-negative Matrices

A non-negative matrix A is said to be primitive if some power A^k is positive. The minimum k with this property is called the exponent of A, denoted $\exp A$.

Helmut Wielandt proved in 1950 that for any primitive $n \times n$-matrix A, one has $\exp A \leq n^2 - 2n + 2 = (n-1)^2 + 1$, and this bound is tight. Possible exponents of $n \times n$-matrices were intensively studied in the 1960s, and it was discovered that two extreme values are each attained by a unique matrix, then there is a gap followed by an island followed by another gap. The sizes of the gaps and of the island perfectly match the sizes of the gaps and of the islands in possible reset thresholds of synchronizing automata with n states — basically one has the same picture shifted by 1. Clearly, this cannot be a mere coincidence.

Exponents of Non-negative Matrices

A non-negative matrix A is said to be primitive if some power A^k is positive. The minimum k with this property is called the exponent of A, denoted $\exp A$.

Helmut Wielandt proved in 1950 that for any primitive $n \times n$-matrix A, one has $\exp A \leq n^2 - 2n + 2 = (n-1)^2 + 1$, and this bound is tight. Possible exponents of $n \times n$-matrices were intensively studied in the 1960s, and it was discovered that two extreme values are each attained by a unique matrix, then there is a gap followed by an island followed by another gap. The sizes of the gaps and of the island perfectly match the sizes of the gaps and of the islands in possible reset thresholds of synchronizing automata with n states — basically one has the same picture shifted by 1. Clearly, this cannot be a mere coincidence.

A directed graph (digraph) is a pair $D = \langle V, E\rangle$.

- V set of vertices
- $E \subseteq V \times V$ set of edges

This definition allows loops but excludes multiple edges.

The matrix of a digraph $D = \langle V, E\rangle$ is just the incidence matrix of the edge relation, that is, a $V \times V$-matrix whose entry in the row v and the column v' is 1 if $(v, v') \in E$ and 0 otherwise.

Digraphs and Matrices

A directed graph (digraph) is a pair $D = \langle V, E \rangle$.

- V set of vertices
- $E \subseteq V \times V$ set of edges

This definition allows loops but excludes multiple edges.

The matrix of a digraph $D = \langle V, E \rangle$ is just the incidence matrix of the edge relation, that is, a $V \times V$-matrix whose entry in the row v and the column v' is 1 if $(v, v') \in E$ and 0 otherwise.

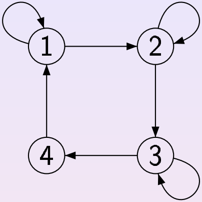
A directed graph (digraph) is a pair $D = \langle V, E \rangle$.

- V set of vertices
- $E \subseteq V \times V$ set of edges

This definition allows loops but excludes multiple edges.

The matrix of a digraph $D = \langle V, E \rangle$ is just the incidence matrix of the edge relation, that is, a $V \times V$-matrix whose entry in the row v and the column v' is 1 if $(v, v') \in E$ and 0 otherwise.

For instance, the matrix of the digraph



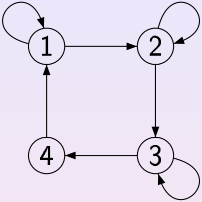
(with respect to the chosen numbering of its vertices) is $\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$.

Conversely, given an $n \times n$-matrix $P = (p_{ij})$ with non-negative real entries, we assign to it a digraph $D(P)$ on the set $\{1, 2, \ldots, n\}$ as follows: (i, j) is an edge of $D(P)$ if and only if $p_{ij} > 0$.

This 'two-way' correspondence allows us to reformulate in terms of digraphs several important notions and results which originated in the classical Perron–Frobenius theory of non-negative matrices.

Digraphs and Matrices

For instance, the matrix of the digraph

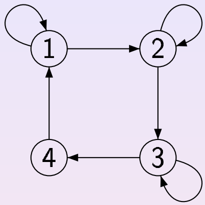


(with respect to the chosen numbering of its vertices) is $\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$.

Conversely, given an $n \times n$-matrix $P = (p_{ij})$ with non-negative real entries, we assign to it a digraph $D(P)$ on the set $\{1, 2, \dots, n\}$ as follows: (i, j) is an edge of $D(P)$ if and only if $p_{ij} > 0$.

This 'two-way' correspondence allows us to reformulate in terms of digraphs several important notions and results which originated in the classical Perron–Frobenius theory of non-negative matrices.

For instance, the matrix of the digraph



(with respect to the chosen numbering of its vertices) is $\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$.

Conversely, given an $n \times n$-matrix $P = (p_{ij})$ with non-negative real entries, we assign to it a digraph $D(P)$ on the set $\{1, 2, \ldots, n\}$ as follows: (i, j) is an edge of $D(P)$ if and only if $p_{ij} > 0$.

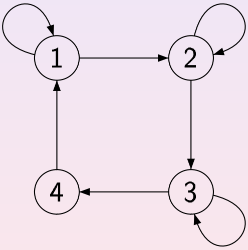
This 'two-way' correspondence allows us to reformulate in terms of digraphs several important notions and results which originated in the classical Perron–Frobenius theory of non-negative matrices.

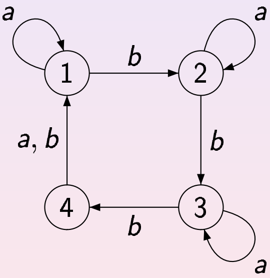
Digraphs and Colorings

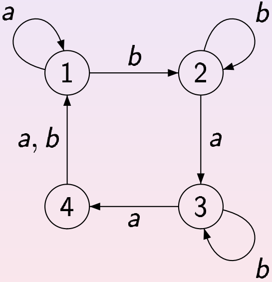
By a coloring of a digraph we mean assigning labels from an alphabet Σ to edges such that the digraph labeled this way becomes a DFA.

Digraphs and Colorings

By a coloring of a digraph we mean assigning labels from an alphabet Σ to edges such that the digraph labeled this way becomes a DFA.







Primitive Digraphs

A digraph D is primitive if D is strongly connected and the greatest common divisor of the lengths of all cycles in D is equal to 1.

Adler, Goodwyn, Weiss (Equivalence of topological Markov shifts, Israel J. Math. 27 (1977) 49–63):
Underlying digraphs of strongly connected synchronizing automata are primitive.

The Road Coloring Conjecture: Every primitive digraph admits a synchronizing coloring.

This was confirmed by Avraham Trahtman (The Road Coloring Problem, Israel J. Math. 172 (2009) 51–60).

Primitive Digraphs

A digraph D is primitive if D is strongly connected and the greatest common divisor of the lengths of all cycles in D is equal to 1.

Adler, Goodwyn, Weiss (Equivalence of topological Markov shifts, Israel J. Math. 27 (1977) 49–63):
Underlying digraphs of strongly connected synchronizing automata are primitive.

The Road Coloring Conjecture: Every primitive digraph admits a synchronizing coloring.

This was confirmed by Avraham Trahtman (The Road Coloring Problem, Israel J. Math. 172 (2009) 51–60).

Primitive Digraphs

A digraph D is primitive if D is strongly connected and the greatest common divisor of the lengths of all cycles in D is equal to 1.

Adler, Goodwyn, Weiss (Equivalence of topological Markov shifts, Israel J. Math. 27 (1977) 49–63):
Underlying digraphs of strongly connected synchronizing automata are primitive.

The Road Coloring Conjecture: Every primitive digraph admits a synchronizing coloring.

This was confirmed by Avraham Trahtman (The Road Coloring Problem, Israel J. Math. 172 (2009) 51–60).

Primitive Digraphs

A digraph D is primitive if D is strongly connected and the greatest common divisor of the lengths of all cycles in D is equal to 1.

Adler, Goodwyn, Weiss (Equivalence of topological Markov shifts, Israel J. Math. 27 (1977) 49–63):
Underlying digraphs of strongly connected synchronizing automata are primitive.

The Road Coloring Conjecture: Every primitive digraph admits a synchronizing coloring.

This was confirmed by Avraham Trahtman (The Road Coloring Problem, Israel J. Math. 172 (2009) 51–60).

A digraph D is primitive iff there exists t such that for each pair of vertices there exists a path between them of length exactly t. (This is equivalent to saying that the t-th power of the matrix of D is positive.) The least t with this property is called the exponent of the digraph D and is denoted by $\gamma(D)$.

1950, Wielandt: The exponent of every primitive digraph on n vertices is not greater than $(n-1)^2+1$ and this bound is tight.

1964, Dulmage–Mendelsohn: There is exactly one primitive digraph on n vertices with exponent $(n-1)^2+1$ and exactly one primitive digraph on n vertices with exponent $(n-1)^2$.
If $n > 4$ is even, then there is no primitive digraph D on n vertices such that $n^2-4n+6 < \gamma(D) < (n-1)^2$.
If $n > 3$ is odd, then there is no primitive digraph D on n vertices such that $n^2-3n+4 < \gamma(D) < (n-1)^2$, or $n^2-4n+6 < \gamma(D) < n^2-3n+2$.

Exponents

A digraph D is primitive iff there exists t such that for each pair of vertices there exists a path between them of length exactly t. (This is equivalent to saying that the t-th power of the matrix of D is positive.) The least t with this property is called the exponent of the digraph D and is denoted by $\gamma(D)$.

1950, Wielandt: The exponent of every primitive digraph on n vertices is not greater than $(n-1)^2+1$ and this bound is tight.

1964, Dulmage–Mendelsohn: There is exactly one primitive digraph on n vertices with exponent $(n-1)^2+1$ and exactly one primitive digraph on n vertices with exponent $(n-1)^2$.
If $n > 4$ is even, then there is no primitive digraph D on n vertices such that $n^2-4n+6 < \gamma(D) < (n-1)^2$.
If $n > 3$ is odd, then there is no primitive digraph D on n vertices such that $n^2-3n+4 < \gamma(D) < (n-1)^2$, or $n^2-4n+6 < \gamma(D) < n^2-3n+2$.

Exponents

A digraph D is primitive iff there exists t such that for each pair of vertices there exists a path between them of length exactly t. (This is equivalent to saying that the t-th power of the matrix of D is positive.) The least t with this property is called the exponent of the digraph D and is denoted by $\gamma(D)$.

1950, Wielandt: The exponent of every primitive digraph on n vertices is not greater than $(n-1)^2+1$ and this bound is tight.

1964, Dulmage–Mendelsohn: There is exactly one primitive digraph on n vertices with exponent $(n-1)^2+1$ and exactly one primitive digraph on n vertices with exponent $(n-1)^2$.
If $n > 4$ is even, then there is no primitive digraph D on n vertices such that $n^2-4n+6 < \gamma(D) < (n-1)^2$.
If $n > 3$ is odd, then there is no primitive digraph D on n vertices such that $n^2-3n+4 < \gamma(D) < (n-1)^2$, or
$n^2-4n+6 < \gamma(D) < n^2-3n+2$.

Exponents

A digraph D is primitive iff there exists t such that for each pair of vertices there exists a path between them of length exactly t. (This is equivalent to saying that the t-th power of the matrix of D is positive.) The least t with this property is called the exponent of the digraph D and is denoted by $\gamma(D)$.

1950, Wielandt: The exponent of every primitive digraph on n vertices is not greater than $(n-1)^2+1$ and this bound is tight.

1964, Dulmage–Mendelsohn: There is exactly one primitive digraph on n vertices with exponent $(n-1)^2+1$ and exactly one primitive digraph on n vertices with exponent $(n-1)^2$.
If $n > 4$ is even, then there is no primitive digraph D on n vertices such that $n^2-4n+6 < \gamma(D) < (n-1)^2$.
If $n > 3$ is odd, then there is no primitive digraph D on n vertices such that $n^2-3n+4 < \gamma(D) < (n-1)^2$, or
$n^2-4n+6 < \gamma(D) < n^2-3n+2$.

Exponents

A digraph D is primitive iff there exists t such that for each pair of vertices there exists a path between them of length exactly t. (This is equivalent to saying that the t-th power of the matrix of D is positive.) The least t with this property is called the exponent of the digraph D and is denoted by $\gamma(D)$.

1950, Wielandt: The exponent of every primitive digraph on n vertices is not greater than $(n-1)^2+1$ and this bound is tight.

1964, Dulmage–Mendelsohn: There is exactly one primitive digraph on n vertices with exponent $(n-1)^2+1$ and exactly one primitive digraph on n vertices with exponent $(n-1)^2$.

If $n>4$ is even, then there is no primitive digraph D on n vertices such that $n^2-4n+6<\gamma(D)<(n-1)^2$.

If $n>3$ is odd, then there is no primitive digraph D on n vertices such that $n^2-3n+4<\gamma(D)<(n-1)^2$, or $n^2-4n+6<\gamma(D)<n^2-3n+2$.

A digraph D is primitive iff there exists t such that for each pair of vertices there exists a path between them of length exactly t. (This is equivalent to saying that the t-th power of the matrix of D is positive.) The least t with this property is called the exponent of the digraph D and is denoted by $\gamma(D)$.

1950, Wielandt: The exponent of every primitive digraph on n vertices is not greater than $(n-1)^2+1$ and this bound is tight.

1964, Dulmage–Mendelsohn: There is exactly one primitive digraph on n vertices with exponent $(n-1)^2+1$ and exactly one primitive digraph on n vertices with exponent $(n-1)^2$.
If $n>4$ is even, then there is no primitive digraph D on n vertices such that $n^2-4n+6<\gamma(D)<(n-1)^2$.
If $n>3$ is odd, then there is no primitive digraph D on n vertices such that $n^2-3n+4<\gamma(D)<(n-1)^2$, or
$n^2-4n+6<\gamma(D)<n^2-3n+2$.

Exponents of primitive digraphs with 9 vertices vs reset thresholds of 2-letter strongly connected synchronizing automata with 9 states

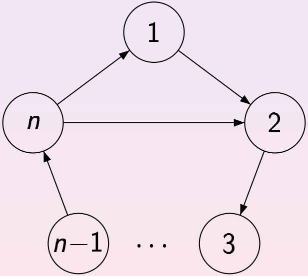
N	65	64	63	62	61	60	59	58	57	56	55	54	53	52	51
# of primitive digraphs with exponent N	1	1	0	0	0	0	0	1	1	2	0	0	0	0	3
# of 2-letter synchronizing automata with reset threshold N	0	1	0	0	0	0	0	1	2	3	0	0	0	4	4

The Wielandt Automaton

The Wielandt automaton $\mathscr{W}_n$ is a (unique) coloring of the Wielandt digraph W_n with $\gamma(W_n) = (n-1)^2 + 1$. The Wielandt digraph has n vertices $1, 2, \dots, n$, say, and the following $n+1$ edges: $(i, i+1)$ for $i = 1, \dots, n-1$, $(n, 1)$, and $(n, 2)$.

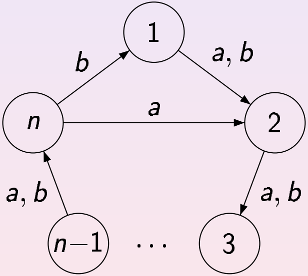
The Wielandt Automaton

The Wielandt automaton $\mathscr{W}_n$ is a (unique) coloring of the Wielandt digraph W_n with $\gamma(W_n) = (n-1)^2 + 1$. The Wielandt digraph has n vertices $1, 2, \dots, n$, say, and the following $n+1$ edges: $(i, i+1)$ for $i = 1, \dots, n-1$, $(n, 1)$, and $(n, 2)$.



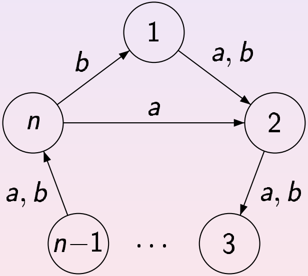
The Wielandt Automaton

The Wielandt automaton $\mathscr{W}_n$ is a (unique) coloring of the Wielandt digraph W_n with $\gamma(W_n) = (n-1)^2 + 1$. The Wielandt digraph has n vertices $1, 2, \dots, n$, say, and the following $n+1$ edges: $(i, i+1)$ for $i = 1, \dots, n-1$, $(n, 1)$, and $(n, 2)$.



The Wielandt Automaton

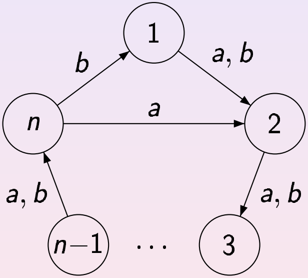
The Wielandt automaton $\mathscr{W}_n$ is a (unique) coloring of the Wielandt digraph W_n with $\gamma(W_n) = (n-1)^2 + 1$. The Wielandt digraph has n vertices $1, 2, \ldots, n$, say, and the following $n+1$ edges: $(i, i+1)$ for $i = 1, \ldots, n-1$, $(n, 1)$, and $(n, 2)$.



It is easy to show that the reset threshold of $\mathscr{W}_n$ is $n^2 - 3n + 3$.

The Wielandt Automaton

The Wielandt automaton $\mathscr{W}_n$ is a (unique) coloring of the Wielandt digraph W_n with $\gamma(W_n) = (n-1)^2 + 1$. The Wielandt digraph has n vertices $1, 2, \ldots, n$, say, and the following $n+1$ edges: $(i, i+1)$ for $i = 1, \ldots, n-1$, $(n, 1)$, and $(n, 2)$.



It is easy to show that the reset threshold of $\mathscr{W}_n$ is $n^2 - 3n + 3$.

In a similar way, each digraph with large exponent generates slowly synchronizing automata.

Observation

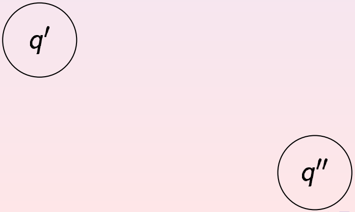
Let a strongly connected synchronizing automaton with n states and reset threshold t be a coloring of a digraph D. Then

$$\gamma(D) \leq t + n - 1.$$

Observation

Let a strongly connected synchronizing automaton with n states and reset threshold t be a coloring of a digraph D. Then

$$\gamma(D) \leq t + n - 1.$$



Observation

Let a strongly connected synchronizing automaton with n states and reset threshold t be a coloring of a digraph D. Then

$$\gamma(D) \leq t + n - 1.$$

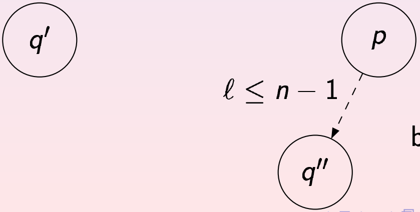
the state to which our automaton is reset by a word of length t

Colorings of Digraphs with Large Exponents

Observation

Let a strongly connected synchronizing automaton with n states and reset threshold t be a coloring of a digraph D. Then

$$\gamma(D) \leq t + n - 1.$$

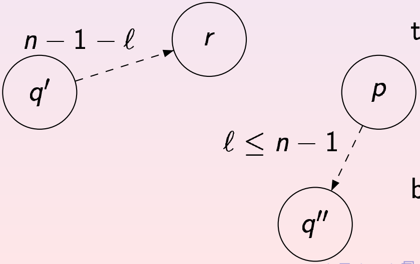


the state to which our automaton is reset by a word of length t

Observation

Let a strongly connected synchronizing automaton with n states and reset threshold t be a coloring of a digraph D. Then

$$\gamma(D) \le t + n - 1.$$

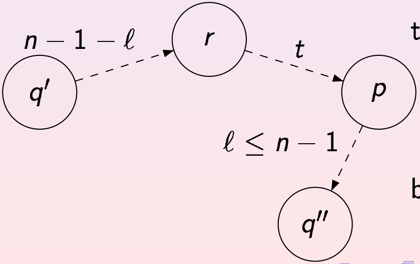


the state to which our automaton is reset by a word of length t

Observation

Let a strongly connected synchronizing automaton with n states and reset threshold t be a coloring of a digraph D. Then

$$\gamma(D) \leq t + n - 1.$$



the state to which our automaton is reset by a word of length t

Colorings of Digraphs with Large Exponents

Observation

Let a strongly connected synchronizing automaton with n states and reset threshold t be a coloring of a digraph D. Then

$$\gamma(D) \leq t + n - 1.$$

For instance, the reset threshold t of the Wielandt automaton $\mathscr{W}_n$ must satisfy

$$t \geq \gamma(W_n) - n + 1 = (n-1)^2 + 1 - n + 1 = n^2 - 3n + 3,$$

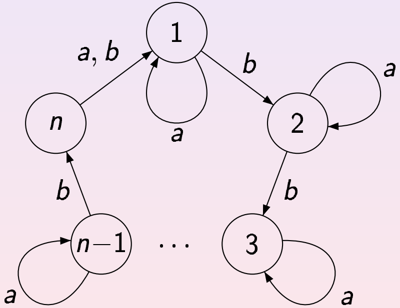
and it is easy to find a reset word of length $n^2 - 3n + 3$.

The Černý Automaton

There are slowly synchronizing automata that cannot be obtained as colorings of a digraph with large exponent. For instance, the Černý automaton $\mathscr{C}_n$ has reset threshold $(n-1)^2$ while its underlying digraph has exponent $n-1$.

The Černý Automaton

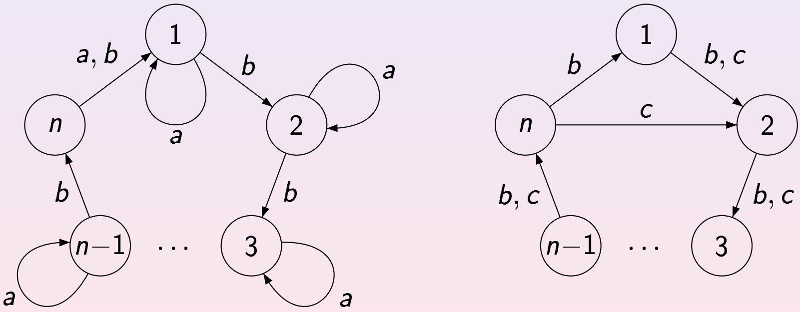
There are slowly synchronizing automata that cannot be obtained as colorings of a digraph with large exponent. For instance, the Černý automaton $\mathscr{C}_n$ has reset threshold $(n-1)^2$ while its underlying digraph has exponent $n-1$.



June 13th, 2013

The Černý Automaton

There are slowly synchronizing automata that cannot be obtained as colorings of a digraph with large exponent. For instance, the Černý automaton $\mathscr{C}_n$ has reset threshold $(n-1)^2$ while its underlying digraph has exponent $n-1$.



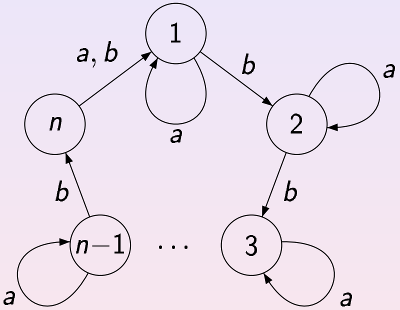
However, $\mathscr{C}_n$ becomes $\mathscr{W}_n$ under the action of b and $c = ab$.

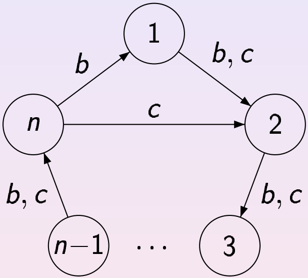
The Černý Automaton

Let w be a shortest reset word for $\mathscr{C}_n$. It must end with a and every other occurrence of a in w is followed by an occurrence of b. Thus, $w = w'a$ where w' can be rewritten into a word v over the alphabet $\{b, c\}$. Since w' and v act in the same way, the word vc is a reset word for $\mathscr{W}_n$. Hence $|v| \geq n^2 - 3n + 2$.

The Černý Automaton

Let w be a shortest reset word for $\mathscr{C}_n$. It must end with a and every other occurrence of a in w is followed by an occurrence of b.

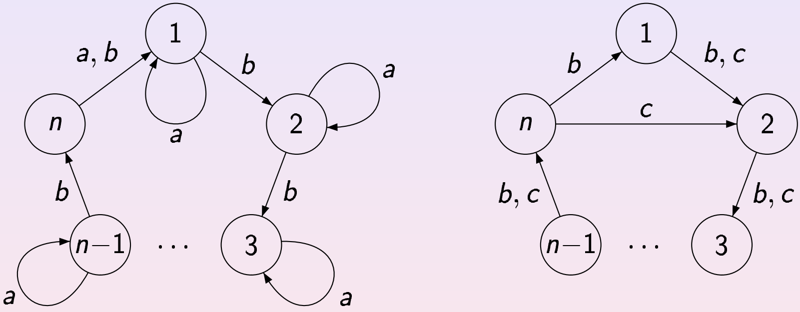




Thus, $w = w'a$ where w' can be rewritten into a word v over the alphabet $\{b, c\}$. Since w' and v act in the same way, the word vc is a reset word for $\mathscr{W}_n$. Hence $|v| \geq n^2 - 3n + 2$.

The Černý Automaton

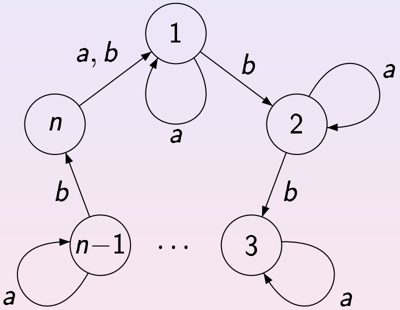
Let w be a shortest reset word for $\mathscr{C}_n$. It must end with a and every other occurrence of a in w is followed by an occurrence of b.

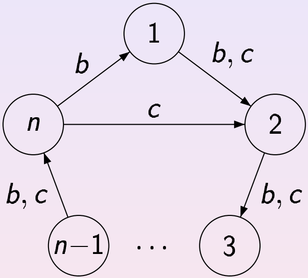


Thus, $w = w'a$ where w' can be rewritten into a word v over the alphabet $\{b, c\}$. Since w' and v act in the same way, the word vc is a reset word for $\mathscr{W}_n$. Hence $|v| \geq n^2 - 3n + 2$.

The Černý Automaton

Let w be a shortest reset word for $\mathscr{C}_n$. It must end with a and every other occurrence of a in w is followed by an occurrence of b.

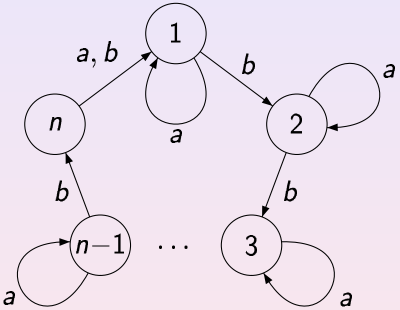


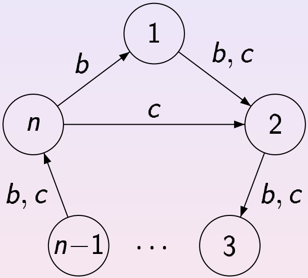


Thus, $w = w'a$ where w' can be rewritten into a word v over the alphabet $\{b, c\}$. Since w' and v act in the same way, the word vc is a reset word for $\mathscr{W}_n$. Hence $|v| \geq n^2 - 3n + 2$.

The Černý Automaton

Let w be a shortest reset word for $\mathscr{C}_n$. It must end with a and every other occurrence of a in w is followed by an occurrence of b.





Thus, $w = w'a$ where w' can be rewritten into a word v over the alphabet $\{b, c\}$. Since w' and v act in the same way, the word vc is a reset word for $\mathscr{W}_n$. Hence $|v| \geq n^2 - 3n + 2$.

The Černý Automaton

Let w be a shortest reset word for $\mathscr{C}_n$. It must end with a and every other occurrence of a in w is followed by an occurrence of b. Thus, $w = w'a$ where w' can be rewritten into a word v over the alphabet $\{b, c\}$. Since w' and v act in the same way, the word vc is a reset word for $\mathscr{W}_n$. Hence $|v| \geq n^2 - 3n + 2$.
Further, v contains at least $n - 2$ occurrences of c. Since each occurrence of c in v corresponds to an occurrence of ab in w', we conclude that $|w'| \geq n^2 - 3n + 2 + n - 2 = n^2 - 2n$.

Let w be a shortest reset word for $\mathscr{C}_n$. It must end with a and every other occurrence of a in w is followed by an occurrence of b. Thus, $w = w'a$ where w' can be rewritten into a word v over the alphabet $\{b, c\}$. Since w' and v act in the same way, the word vc is a reset word for $\mathscr{W}_n$. Hence $|v| \geq n^2 - 3n + 2$.
Further, v contains at least $n - 2$ occurrences of c. Since each occurrence of c in v corresponds to an occurrence of ab in w', we conclude that $|w'| \geq n^2 - 3n + 2 + n - 2 = n^2 - 2n$. Thus, $|w| = |w'a| \geq n^2 - 2n + 1 = (n-1)^2$.

The Černý Automaton

Let w be a shortest reset word for $\mathscr{C}_n$. It must end with a and every other occurrence of a in w is followed by an occurrence of b. Thus, $w = w'a$ where w' can be rewritten into a word v over the alphabet $\{b, c\}$. Since w' and v act in the same way, the word vc is a reset word for $\mathscr{W}_n$. Hence $|v| \geq n^2 - 3n + 2$.
Further, v contains at least $n - 2$ occurrences of c. Since each occurrence of c in v corresponds to an occurrence of ab in w', we conclude that $|w'| \geq n^2 - 3n + 2 + n - 2 = n^2 - 2n$. Thus, $|w| = |w'a| \geq n^2 - 2n + 1 = (n - 1)^2$.

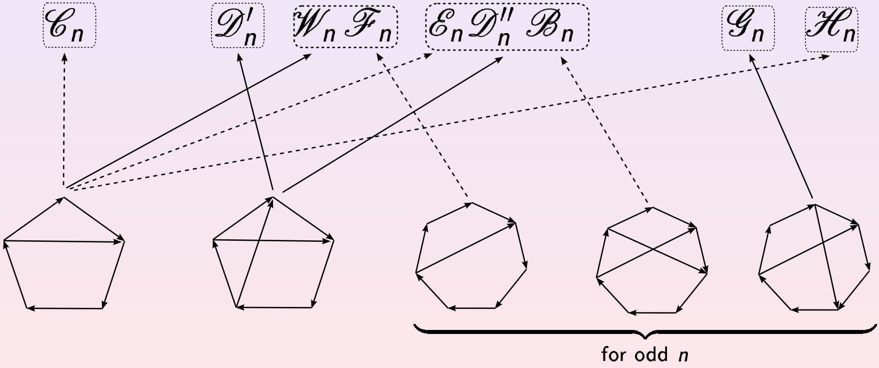
Thus, it is the Wielandt digraph that stays behind the Černý automaton!

Digraphs vs Automata

In a similar manner it is easy to recover every known slowly synchronizing automaton from a suitable digraph with large exponent.

Digraphs vs Automata

In a similar manner it is easy to recover every known slowly synchronizing automaton from a suitable digraph with large exponent.



How to get upper bounds for reset threshold?

For $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$, a subset $P \subset Q$ is extensible if $P \supseteq R \,.\, w$ for some $w \in \Sigma^*$ of length at most $n = |Q|$ and some $R \subseteq Q$ with $|R| > |P|$. It was conjectured for some time that in synchronizing automata every proper non-singleton subset is extensible. Observe that this would imply the Černý conjecture.

Indeed, some letter a should sent two states q, q' to the same state p. Let $P_0 = \{q, q'\}$ and, for $i > 0$, let P_i be such that $|P_i| > |P_{i-1}|$ and $P_{i-1} \supseteq P_i \,.\, w_i$ for some word w_i of length $\leq n$. Then in at most $n - 2$ steps the sequence $P_0, P_1, P_2, \dots$ reaches Q and

$$Q \,.\, w_{n-2} w_{n-3} \cdots w_1 a = \{p\},$$

that is, $w_{n-2} w_{n-3} \cdots w_1 a$ is a reset word. The length of this reset word is at most $n(n-2) + 1 = (n-1)^2$.

Extensibility

How to get upper bounds for reset threshold?

For $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$, a subset $P \subset Q$ is extensible if $P \supseteq R\,.\,w$ for some $w \in \Sigma^*$ of length at most $n = |Q|$ and some $R \subseteq Q$ with $|R| > |P|$. It was conjectured for some time that in synchronizing automata every proper non-singleton subset is extensible. Observe that this would imply the Černý conjecture.

Indeed, some letter a should sent two states q, q' to the same state p. Let $P_0 = \{q, q'\}$ and, for $i > 0$, let P_i be such that $|P_i| > |P_{i-1}|$ and $P_{i-1} \supseteq P_i\,.\,w_i$ for some word w_i of length $\leq n$. Then in at most $n-2$ steps the sequence $P_0, P_1, P_2, \dots$ reaches Q and

$$Q\,.\,w_{n-2}w_{n-3}\cdots w_1 a = \{p\},$$

that is, $w_{n-2}w_{n-3}\cdots w_1 a$ is a reset word. The length of this reset word is at most $n(n-2)+1 = (n-1)^2$.

Extensibility

How to get upper bounds for reset threshold?

For $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$, a subset $P \subset Q$ is extensible if $P \supseteq R \, . \, w$ for some $w \in \Sigma^*$ of length at most $n = |Q|$ and some $R \subseteq Q$ with $|R| > |P|$. It was conjectured for some time that in synchronizing automata every proper non-singleton subset is extensible. Observe that this would imply the Černý conjecture.

Indeed, some letter a should sent two states q, q' to the same state p. Let $P_0 = \{q, q'\}$ and, for $i > 0$, let P_i be such that $|P_i| > |P_{i-1}|$ and $P_{i-1} \supseteq P_i \, . \, w_i$ for some word w_i of length $\leq n$. Then in at most $n-2$ steps the sequence $P_0, P_1, P_2, \ldots$ reaches Q and

$$Q \, . \, w_{n-2} w_{n-3} \cdots w_1 a = \{p\},$$

that is, $w_{n-2} w_{n-3} \cdots w_1 a$ is a reset word. The length of this reset word is at most $n(n-2) + 1 = (n-1)^2$.

Extensibility

How to get upper bounds for reset threshold?

For $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$, a subset $P \subset Q$ is extensible if $P \supseteq R \,.\, w$ for some $w \in \Sigma^*$ of length at most $n = |Q|$ and some $R \subseteq Q$ with $|R| > |P|$. It was conjectured for some time that in synchronizing automata every proper non-singleton subset is extensible. Observe that this would imply the Černý conjecture.

Indeed, some letter a should sent two states q, q' to the same state p. Let $P_0 = \{q, q'\}$ and, for $i > 0$, let P_i be such that $|P_i| > |P_{i-1}|$ and $P_{i-1} \supseteq P_i \,.\, w_i$ for some word w_i of length $\leq n$. Then in at most $n - 2$ steps the sequence $P_0, P_1, P_2, \dots$ reaches Q and

$$Q \,.\, w_{n-2} w_{n-3} \cdots w_1 a = \{p\},$$

that is, $w_{n-2} w_{n-3} \cdots w_1 a$ is a reset word. The length of this reset word is at most $n(n-2) + 1 = (n-1)^2$.

Extensibility

How to get upper bounds for reset threshold?

For $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$, a subset $P \subset Q$ is extensible if $P \supseteq R\,.\,w$ for some $w \in \Sigma^*$ of length at most $n = |Q|$ and some $R \subseteq Q$ with $|R| > |P|$. It was conjectured for some time that in synchronizing automata every proper non-singleton subset is extensible. Observe that this would imply the Černý conjecture.

Indeed, some letter a should sent two states q, q' to the same state p. Let $P_0 = \{q, q'\}$ and, for $i > 0$, let P_i be such that $|P_i| > |P_{i-1}|$ and $P_{i-1} \supseteq P_i\,.\,w_i$ for some word w_i of length $\leq n$. Then in at most $n - 2$ steps the sequence $P_0, P_1, P_2, \dots$ reaches Q and

$$Q\,.\,w_{n-2}w_{n-3}\cdots w_1 a = \{p\},$$

that is, $w_{n-2}w_{n-3}\cdots w_1 a$ is a reset word. The length of this reset word is at most $n(n-2) + 1 = (n-1)^2$.

Extensibility

How to get upper bounds for reset threshold?

For $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$, a subset $P \subset Q$ is extensible if $P \supseteq R\,.\,w$ for some $w \in \Sigma^*$ of length at most $n = |Q|$ and some $R \subseteq Q$ with $|R| > |P|$. It was conjectured for some time that in synchronizing automata every proper non-singleton subset is extensible. Observe that this would imply the Černý conjecture.

Indeed, some letter a should sent two states q, q' to the same state p. Let $P_0 = \{q, q'\}$ and, for $i > 0$, let P_i be such that $|P_i| > |P_{i-1}|$ and $P_{i-1} \supseteq P_i\,.\,w_i$ for some word w_i of length $\leq n$. Then in at most $n-2$ steps the sequence $P_0, P_1, P_2, \ldots$ reaches Q and

$$Q\,.\,w_{n-2}w_{n-3}\cdots w_1 a = \{p\},$$

that is, $w_{n-2}w_{n-3}\cdots w_1 a$ is a reset word. The length of this reset word is at most $n(n-2)+1 = (n-1)^2$.

Extensibility

How to get upper bounds for reset threshold?

For $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$, a subset $P \subset Q$ is extensible if $P \supseteq R\,.\,w$ for some $w \in \Sigma^*$ of length at most $n = |Q|$ and some $R \subseteq Q$ with $|R| > |P|$. It was conjectured for some time that in synchronizing automata every proper non-singleton subset is extensible. Observe that this would imply the Černý conjecture.

Indeed, some letter a should sent two states q, q' to the same state p. Let $P_0 = \{q, q'\}$ and, for $i > 0$, let P_i be such that $|P_i| > |P_{i-1}|$ and $P_{i-1} \supseteq P_i\,.\,w_i$ for some word w_i of length $\leq n$. Then in at most $n-2$ steps the sequence $P_0, P_1, P_2, \ldots$ reaches Q and

$$Q\,.\,w_{n-2}w_{n-3}\cdots w_1 a = \{p\},$$

that is, $w_{n-2}w_{n-3}\cdots w_1 a$ is a reset word. The length of this reset word is at most $n(n-2)+1 = (n-1)^2$.

Extensibility

How to get upper bounds for reset threshold?

For $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$, a subset $P \subset Q$ is extensible if $P \supseteq R\,.\,w$ for some $w \in \Sigma^*$ of length at most $n = |Q|$ and some $R \subseteq Q$ with $|R| > |P|$. It was conjectured for some time that in synchronizing automata every proper non-singleton subset is extensible. Observe that this would imply the Černý conjecture.

Indeed, some letter a should sent two states q, q' to the same state p. Let $P_0 = \{q, q'\}$ and, for $i > 0$, let P_i be such that $|P_i| > |P_{i-1}|$ and $P_{i-1} \supseteq P_i\,.\,w_i$ for some word w_i of length $\leq n$. Then in at most $n - 2$ steps the sequence $P_0, P_1, P_2, \dots$ reaches Q and

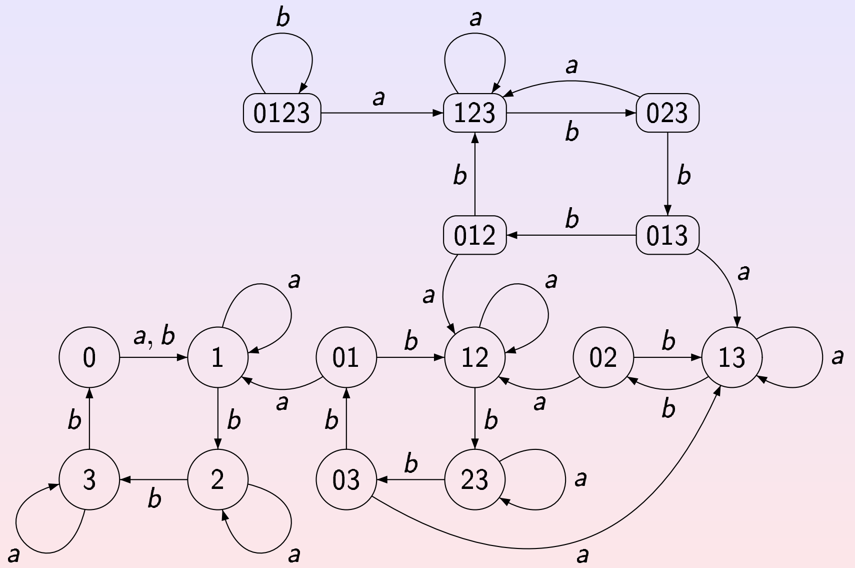
$$Q\,.\,w_{n-2}w_{n-3}\cdots w_1 a = \{p\},$$

that is, $w_{n-2}w_{n-3}\cdots w_1 a$ is a reset word. The length of this reset word is at most $n(n-2)+1 = (n-1)^2$.

Example

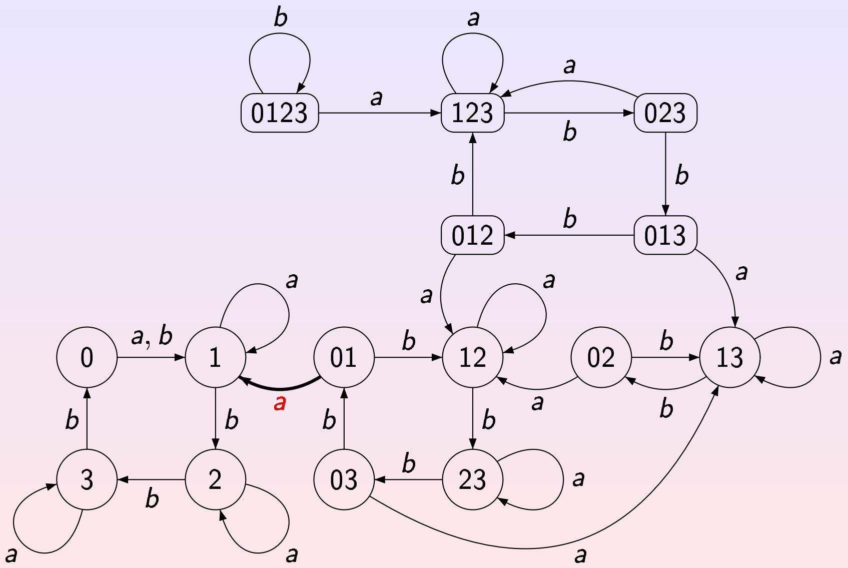
For an illustration, consider the subset automaton of the Černý automaton $\mathscr{C}_4$.

Example



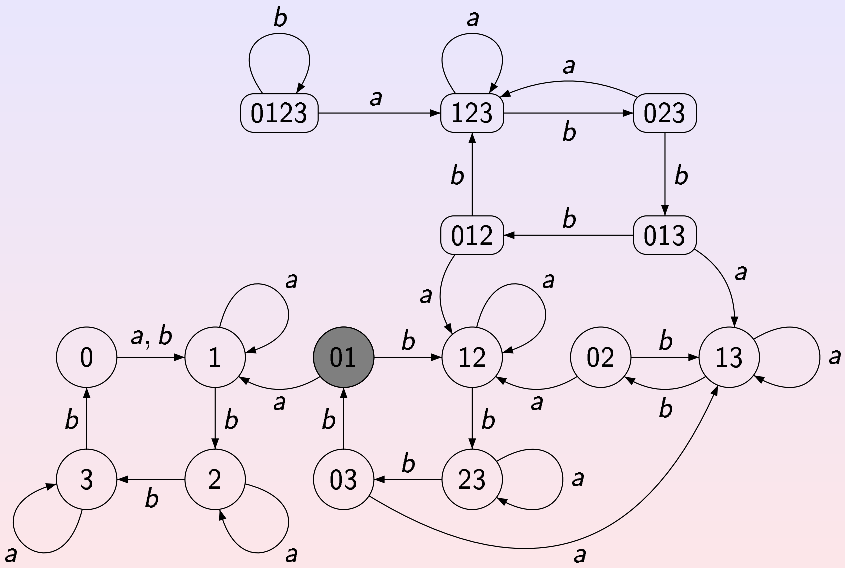
June 13th, 2013

Example

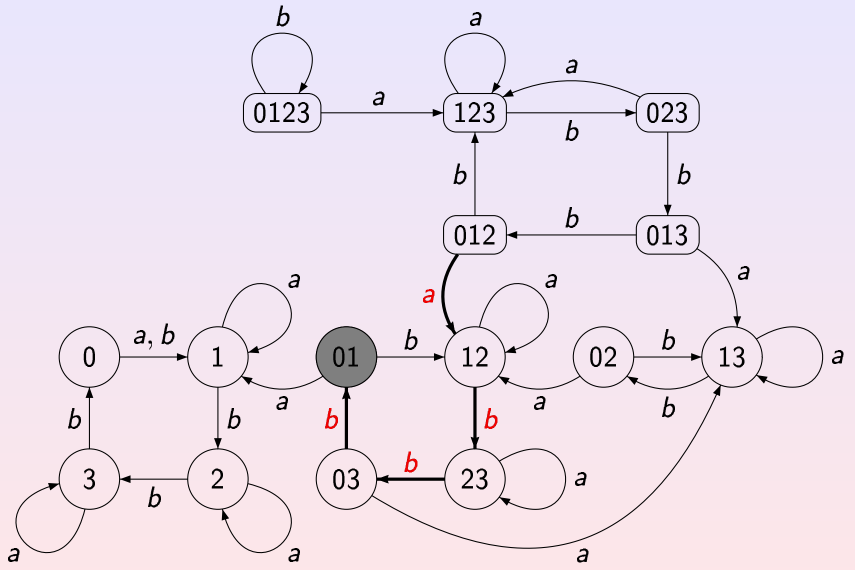


June 13th, 2013

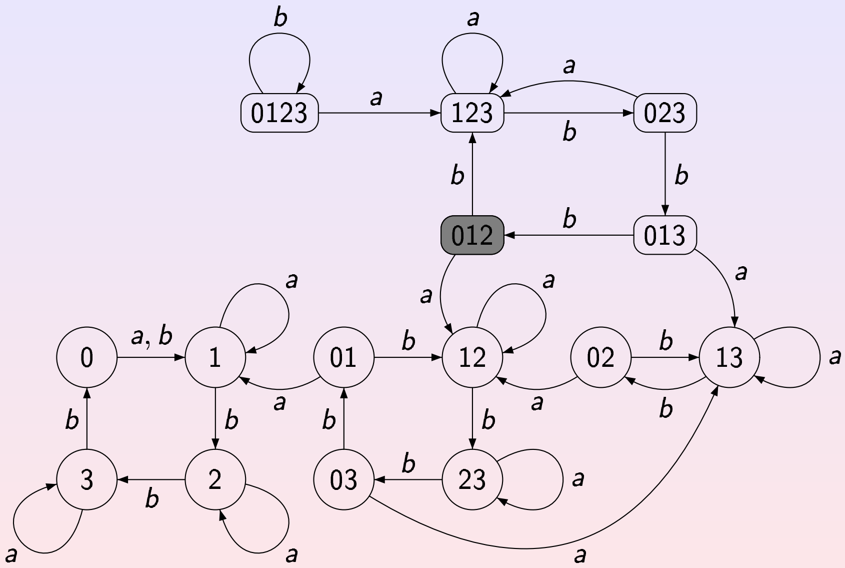
Example



Example

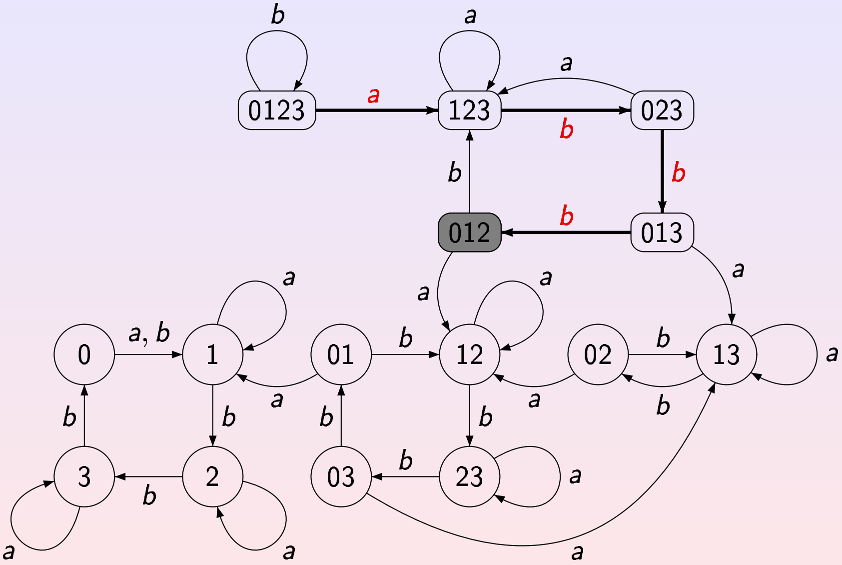


Example



b
a
a
0123
a
123
023
b
b
b
b
012
013
a
a
a
a
0
a, b
1
01
b
12
02
b
13
a
a
a
b
b
b
b
b
b
3
2
03
b
23
a
b
a
a
a

Example



June 13th, 2013

Applications of Extensibility

Several important results confirming the Černý conjecture for various partial cases have been proved by verifying the extensibility conjecture for the corresponding automata. This includes:

- 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).
- Benjamin Steinberg's result for automata in which a letter labels only one cycle (one-cluster automata) and this cycle is of prime length (The Černý conjecture for one-cluster automata with prime length cycle. Theoret. Comput. Sci. 412 (2011) 5487–5491).

Applications of Extensibility

Several important results confirming the Černý conjecture for various partial cases have been proved by verifying the extensibility conjecture for the corresponding automata. This includes:

- 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).
- Benjamin Steinberg's result for automata in which a letter labels only one cycle (one-cluster automata) and this cycle is of prime length (The Černý conjecture for one-cluster automata with prime length cycle. Theoret. Comput. Sci. 412 (2011) 5487–5491).

Applications of Extensibility

Several important results confirming the Černý conjecture for various partial cases have been proved by verifying the extensibility conjecture for the corresponding automata. This includes:

- 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).
- Benjamin Steinberg's result for automata in which a letter labels only one cycle (one-cluster automata) and this cycle is of prime length (The Černý conjecture for one-cluster automata with prime length cycle. Theoret. Comput. Sci. 412 (2011) 5487–5491).

Applications of Extensibility

Several important results confirming the Černý conjecture for various partial cases have been proved by verifying the extensibility conjecture for the corresponding automata. This includes:

- 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).
- Benjamin Steinberg's result for automata in which a letter labels only one cycle (one-cluster automata) and this cycle is of prime length (The Černý conjecture for one-cluster automata with prime length cycle. Theoret. Comput. Sci. 412 (2011) 5487–5491).

Limits of Extensibility

In general, the extensibility conjecture fails. The first (implicit) example was a 6-state automaton found by Jarkko Kari in 2001, and Mikhail Berlinkov has constructed for every $\alpha < 2$ an infinite series of synchronizing automata $\mathscr{B}_\alpha = \langle Q, \Sigma, \delta \rangle$ such that there is a proper non-singleton subset $P \subset Q$ that cannot be extended by any word of length $< \alpha|Q|$ (On a conjecture by Carpi and D'Alessandro, Int. J. Found. Comput. Sci. 22 (2011) 1565–1576).

It is not excluded that in every synchronizing automaton each proper non-singleton subset can be extended by a word of length $2 \times \#$ of states. On the other hand, we don't know even a quadratic (in the number of states) upper bound for the length of extending words.

Limits of Extensibility

In general, the extensibility conjecture fails. The first (implicit) example was a 6-state automaton found by Jarkko Kari in 2001, and Mikhail Berlinkov has constructed for every $\alpha < 2$ an infinite series of synchronizing automata $\mathscr{B}_\alpha = \langle Q, \Sigma, \delta \rangle$ such that there is a proper non-singleton subset $P \subset Q$ that cannot be extended by any word of length $< \alpha|Q|$ (On a conjecture by Carpi and D'Alessandro, Int. J. Found. Comput. Sci. 22 (2011) 1565–1576).

It is not excluded that in every synchronizing automaton each proper non-singleton subset can be extended by a word of length $2 \times \#$ of states. On the other hand, we don't know even a quadratic (in the number of states) upper bound for the length of extending words.

Limits of Extensibility

In general, the extensibility conjecture fails. The first (implicit) example was a 6-state automaton found by Jarkko Kari in 2001, and Mikhail Berlinkov has constructed for every $\alpha < 2$ an infinite series of synchronizing automata $\mathscr{B}_\alpha = \langle Q, \Sigma, \delta \rangle$ such that there is a proper non-singleton subset $P \subset Q$ that cannot be extended by any word of length $< \alpha|Q|$ (On a conjecture by Carpi and D'Alessandro, Int. J. Found. Comput. Sci. 22 (2011) 1565–1576).

It is not excluded that in every synchronizing automaton each proper non-singleton subset can be extended by a word of length $2 \times \#$ of states. On the other hand, we don't know even a quadratic (in the number of states) upper bound for the length of extending words.

Limits of Extensibility

In general, the extensibility conjecture fails. The first (implicit) example was a 6-state automaton found by Jarkko Kari in 2001, and Mikhail Berlinkov has constructed for every $\alpha < 2$ an infinite series of synchronizing automata $\mathscr{B}_\alpha = \langle Q, \Sigma, \delta \rangle$ such that there is a proper non-singleton subset $P \subset Q$ that cannot be extended by any word of length $< \alpha|Q|$ (On a conjecture by Carpi and D'Alessandro, Int. J. Found. Comput. Sci. 22 (2011) 1565–1576).

It is not excluded that in every synchronizing automaton each proper non-singleton subset can be extended by a word of length $2 \times \#$ of states. On the other hand, we don't know even a quadratic (in the number of states) upper bound for the length of extending words.

Limits of Extensibility

In general, the extensibility conjecture fails. The first (implicit) example was a 6-state automaton found by Jarkko Kari in 2001, and Mikhail Berlinkov has constructed for every $\alpha < 2$ an infinite series of synchronizing automata $\mathscr{B}_\alpha = \langle Q, \Sigma, \delta \rangle$ such that there is a proper non-singleton subset $P \subset Q$ that cannot be extended by any word of length $< \alpha|Q|$ (On a conjecture by Carpi and D'Alessandro, Int. J. Found. Comput. Sci. 22 (2011) 1565–1576).

It is not excluded that in every synchronizing automaton each proper non-singleton subset can be extended by a word of length $2 \times \#$ of states. On the other hand, we don't know even a quadratic (in the number of states) upper bound for the length of extending words.

Linearization

We associate a natural linear structure with each automaton $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$. Assume that $Q = \{1, 2, \dots, n\}$ and assign to each subset $K \subseteq Q$ its characteristic vector $[K] \in \mathbb{R}^n$ (the space of n-dimensional column vectors): the i-th entry of $[K]$ is 1 if $i \in K$, otherwise the entry is 0.

For each word $w \in \Sigma^*$, its action on Q gives rise to a linear transformation of $\mathbb{R}^n$; we denote by $[w]$ the matrix of this transformation in the standard basis $[1], \dots, [n]$ of $\mathbb{R}^n$. Clearly, the matrix $[w]$ has exactly one non-zero entry in each column and this entry is equal to 1.

For $K \subseteq Q$ and $v \in \Sigma^*$, let $K \,.\, v^{-1} = \{q \mid q.v \in K\}$. Then $[K \,.\, v^{-1}] = [v]^T[K]$, where $[v]^T$ stands for the usual transpose of the matrix $[v]$. A word w is a reset word for $\mathscr{A}$ iff $q \,.\, w^{-1} = Q$ for some state q. Now we can rewrite this as $[w]^T[q] = [Q]$.

Linearization

We associate a natural linear structure with each automaton $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$. Assume that $Q = \{1, 2, \dots, n\}$ and assign to each subset $K \subseteq Q$ its characteristic vector $[K] \in \mathbb{R}^n$ (the space of n-dimensional column vectors): the i-th entry of $[K]$ is 1 if $i \in K$, otherwise the entry is 0.

For each word $w \in \Sigma^*$, its action on Q gives rise to a linear transformation of $\mathbb{R}^n$; we denote by $[w]$ the matrix of this transformation in the standard basis $[1], \dots, [n]$ of $\mathbb{R}^n$. Clearly, the matrix $[w]$ has exactly one non-zero entry in each column and this entry is equal to 1.

For $K \subseteq Q$ and $v \in \Sigma^*$, let $K \,.\, v^{-1} = \{q \mid q.v \in K\}$. Then $[K \,.\, v^{-1}] = [v]^T[K]$, where $[v]^T$ stands for the usual transpose of the matrix $[v]$. A word w is a reset word for $\mathscr{A}$ iff $q \,.\, w^{-1} = Q$ for some state q. Now we can rewrite this as $[w]^T[q] = [Q]$.

Linearization

We associate a natural linear structure with each automaton $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$. Assume that $Q = \{1, 2, \ldots, n\}$ and assign to each subset $K \subseteq Q$ its characteristic vector $[K] \in \mathbb{R}^n$ (the space of n-dimensional column vectors): the i-th entry of $[K]$ is 1 if $i \in K$, otherwise the entry is 0.

For each word $w \in \Sigma^*$, its action on Q gives rise to a linear transformation of $\mathbb{R}^n$; we denote by $[w]$ the matrix of this transformation in the standard basis $[1], \ldots, [n]$ of $\mathbb{R}^n$. Clearly, the matrix $[w]$ has exactly one non-zero entry in each column and this entry is equal to 1.

For $K \subseteq Q$ and $v \in \Sigma^*$, let $K \, . \, v^{-1} = \{q \mid q.v \in K\}$. Then $[K \, . \, v^{-1}] = [v]^T [K]$, where $[v]^T$ stands for the usual transpose of the matrix $[v]$. A word w is a reset word for $\mathscr{A}$ iff $q \, . \, w^{-1} = Q$ for some state q. Now we can rewrite this as $[w]^T [q] = [Q]$.

Linearization

We associate a natural linear structure with each automaton $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$. Assume that $Q = \{1, 2, \ldots, n\}$ and assign to each subset $K \subseteq Q$ its characteristic vector $[K] \in \mathbb{R}^n$ (the space of n-dimensional column vectors): the i-th entry of $[K]$ is 1 if $i \in K$, otherwise the entry is 0.

For each word $w \in \Sigma^*$, its action on Q gives rise to a linear transformation of $\mathbb{R}^n$; we denote by $[w]$ the matrix of this transformation in the standard basis $[1], \ldots, [n]$ of $\mathbb{R}^n$. Clearly, the matrix $[w]$ has exactly one non-zero entry in each column and this entry is equal to 1.

For $K \subseteq Q$ and $v \in \Sigma^*$, let $K \,.\, v^{-1} = \{q \mid q.v \in K\}$. Then $[K \,.\, v^{-1}] = [v]^T[K]$, where $[v]^T$ stands for the usual transpose of the matrix $[v]$. A word w is a reset word for $\mathscr{A}$ iff $q \,.\, w^{-1} = Q$ for some state q. Now we can rewrite this as $[w]^T[q] = [Q]$.

Linearization

We associate a natural linear structure with each automaton $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$. Assume that $Q = \{1, 2, \dots, n\}$ and assign to each subset $K \subseteq Q$ its characteristic vector $[K] \in \mathbb{R}^n$ (the space of n-dimensional column vectors): the i-th entry of $[K]$ is 1 if $i \in K$, otherwise the entry is 0.

For each word $w \in \Sigma^*$, its action on Q gives rise to a linear transformation of $\mathbb{R}^n$; we denote by $[w]$ the matrix of this transformation in the standard basis $[1], \dots, [n]$ of $\mathbb{R}^n$. Clearly, the matrix $[w]$ has exactly one non-zero entry in each column and this entry is equal to 1.

For $K \subseteq Q$ and $v \in \Sigma^*$, let $K\,.\,v^{-1} = \{q \mid q.v \in K\}$. Then $[K\,.\,v^{-1}] = [v]^T[K]$, where $[v]^T$ stands for the usual transpose of the matrix $[v]$. A word w is a reset word for $\mathscr{A}$ iff $q\,.\,w^{-1} = Q$ for some state q. Now we can rewrite this as $[w]^T[q] = [Q]$.

We associate a natural linear structure with each automaton $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$. Assume that $Q = \{1, 2, \ldots, n\}$ and assign to each subset $K \subseteq Q$ its characteristic vector $[K] \in \mathbb{R}^n$ (the space of n-dimensional column vectors): the i-th entry of $[K]$ is 1 if $i \in K$, otherwise the entry is 0.

For each word $w \in \Sigma^*$, its action on Q gives rise to a linear transformation of $\mathbb{R}^n$; we denote by $[w]$ the matrix of this transformation in the standard basis $[1], \ldots, [n]$ of $\mathbb{R}^n$. Clearly, the matrix $[w]$ has exactly one non-zero entry in each column and this entry is equal to 1.

For $K \subseteq Q$ and $v \in \Sigma^*$, let $K \,.\, v^{-1} = \{q \mid q.v \in K\}$. Then $[K \,.\, v^{-1}] = [v]^T[K]$, where $[v]^T$ stands for the usual transpose of the matrix $[v]$. A word w is a reset word for $\mathscr{A}$ iff $q \,.\, w^{-1} = Q$ for some state q. Now we can rewrite this as $[w]^T[q] = [Q]$.

Linearization

We associate a natural linear structure with each automaton $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$. Assume that $Q = \{1, 2, \ldots, n\}$ and assign to each subset $K \subseteq Q$ its characteristic vector $[K] \in \mathbb{R}^n$ (the space of n-dimensional column vectors): the i-th entry of $[K]$ is 1 if $i \in K$, otherwise the entry is 0.

For each word $w \in \Sigma^*$, its action on Q gives rise to a linear transformation of $\mathbb{R}^n$; we denote by $[w]$ the matrix of this transformation in the standard basis $[1], \ldots, [n]$ of $\mathbb{R}^n$. Clearly, the matrix $[w]$ has exactly one non-zero entry in each column and this entry is equal to 1.

For $K \subseteq Q$ and $v \in \Sigma^*$, let $K \,.\, v^{-1} = \{q \mid q.v \in K\}$. Then $[K \,.\, v^{-1}] = [v]^T[K]$, where $[v]^T$ stands for the usual transpose of the matrix $[v]$. A word w is a reset word for $\mathscr{A}$ iff $q \,.\, w^{-1} = Q$ for some state q. Now we can rewrite this as $[w]^T[q] = [Q]$.

Extensibility in Linear Terms

For vectors $g_1, g_2 \in \mathbb{R}^n$, we denote their usual inner product by (g_1, g_2). Then for any $K, L \subset Q$, we have $([K],[L]) = |K \cap L|$.

Denote by $\mathbf{1}_n$ the uniform stochastic vector in $\mathbb{R}^n$, that is, the vector with all entries equal to $\frac{1}{n}$. Then the fact that a word w extends a subset $K \subset Q$ (that is, the inequality $|K| < |K \, . \, w^{-1}|$) can be rewritten as $([K], \mathbf{1}_n) < ([w]^T[K], \mathbf{1}_n)$.

Thus, the extension method amounts to finding a state q, a letter a, and a sequence of words $w_1, w_2, \dots, w_d$ such that

$$\frac{1}{n} = ([q], \mathbf{1}_n) < ([a]^T[q], \mathbf{1}_n) < ([w_1 a]^T[q], \mathbf{1}_n) < \dots$$
$$\dots < ([w_d \cdots w_2 w_1 a]^T[q], \mathbf{1}_n) = 1.$$

Here $d \leq n-2$ because at each step the inner product increases by at least $\frac{1}{n}$. The problem is that so far we don't any linear bound for the lengths of the w_i's.

Extensibility in Linear Terms

For vectors $g_1, g_2 \in \mathbb{R}^n$, we denote their usual inner product by (g_1, g_2). Then for any $K, L \subset Q$, we have $([K], [L]) = |K \cap L|$.

Denote by $\mathbf{1}_n$ the uniform stochastic vector in $\mathbb{R}^n$, that is, the vector with all entries equal to $\frac{1}{n}$. Then the fact that a word w extends a subset $K \subset Q$ (that is, the inequality $|K| < |K \,.\, w^{-1}|$) can be rewritten as $([K], \mathbf{1}_n) < ([w]^T[K], \mathbf{1}_n)$.

Thus, the extension method amounts to finding a state q, a letter a, and a sequence of words $w_1, w_2, \dots, w_d$ such that

$$\frac{1}{n} = ([q], \mathbf{1}_n) < ([a]^T[q], \mathbf{1}_n) < ([w_1 a]^T[q], \mathbf{1}_n) < \dots$$

$$\dots < ([w_d \cdots w_2 w_1 a]^T[q], \mathbf{1}_n) = 1.$$

Here $d \leq n - 2$ because at each step the inner product increases by at least $\frac{1}{n}$. The problem is that so far we don't any linear bound for the lengths of the w_i's.

Extensibility in Linear Terms

For vectors $g_1, g_2 \in \mathbb{R}^n$, we denote their usual inner product by (g_1, g_2). Then for any $K, L \subset Q$, we have $([K], [L]) = |K \cap L|$. Denote by $\mathbf{1}_n$ the uniform stochastic vector in $\mathbb{R}^n$, that is, the vector with all entries equal to $\frac{1}{n}$. Then the fact that a word w extends a subset $K \subset Q$ (that is, the inequality $|K| < |K \,.\, w^{-1}|$) can be rewritten as $([K], \mathbf{1}_n) < ([w]^T[K], \mathbf{1}_n)$.

Thus, the extension method amounts to finding a state q, a letter a, and a sequence of words $w_1, w_2, \dots, w_d$ such that

$$\frac{1}{n} = ([q], \mathbf{1}_n) < ([a]^T[q], \mathbf{1}_n) < ([w_1 a]^T[q], \mathbf{1}_n) < \dots$$

$$\dots < ([w_d \cdots w_2 w_1 a]^T[q], \mathbf{1}_n) = 1.$$

Here $d \leq n - 2$ because at each step the inner product increases by at least $\frac{1}{n}$. The problem is that so far we don't any linear bound for the lengths of the w_i's.

Extensibility in Linear Terms

For vectors $g_1, g_2 \in \mathbb{R}^n$, we denote their usual inner product by (g_1, g_2). Then for any $K, L \subset Q$, we have $([K], [L]) = |K \cap L|$. Denote by $\mathbf{1}_n$ the uniform stochastic vector in $\mathbb{R}^n$, that is, the vector with all entries equal to $\frac{1}{n}$. Then the fact that a word w extends a subset $K \subset Q$ (that is, the inequality $|K| < |K \,.\, w^{-1}|$) can be rewritten as $([K], \mathbf{1}_n) < ([w]^T[K], \mathbf{1}_n)$.

Thus, the extension method amounts to finding a state q, a letter a, and a sequence of words $w_1, w_2, \ldots, w_d$ such that

$$\frac{1}{n} = ([q], \mathbf{1}_n) < ([a]^T[q], \mathbf{1}_n) < ([w_1 a]^T[q], \mathbf{1}_n) < \ldots$$
$$\cdots < ([w_d \cdots w_2 w_1 a]^T[q], \mathbf{1}_n) = 1.$$

Here $d \leq n - 2$ because at each step the inner product increases by at least $\frac{1}{n}$. The problem is that so far we don't any linear bound for the lengths of the w_i's.

Extensibility in Linear Terms

For vectors $g_1, g_2 \in \mathbb{R}^n$, we denote their usual inner product by (g_1, g_2). Then for any $K, L \subset Q$, we have $([K], [L]) = |K \cap L|$. Denote by $\mathbf{1}_n$ the uniform stochastic vector in $\mathbb{R}^n$, that is, the vector with all entries equal to $\frac{1}{n}$. Then the fact that a word w extends a subset $K \subset Q$ (that is, the inequality $|K| < |K \,.\, w^{-1}|$) can be rewritten as $([K], \mathbf{1}_n) < ([w]^T[K], \mathbf{1}_n)$.

Thus, the extension method amounts to finding a state q, a letter a, and a sequence of words $w_1, w_2, \dots, w_d$ such that

$$\frac{1}{n} = ([q], \mathbf{1}_n) < ([a]^T[q], \mathbf{1}_n) < ([w_1 a]^T[q], \mathbf{1}_n) < \dots$$
$$\dots < ([w_d \cdots w_2 w_1 a]^T[q], \mathbf{1}_n) = 1.$$

Here $d \leq n - 2$ because at each step the inner product increases by at least $\frac{1}{n}$. The problem is that so far we don't any linear bound for the lengths of the w_i's.

Extensibility in Linear Terms

For vectors $g_1, g_2 \in \mathbb{R}^n$, we denote their usual inner product by (g_1, g_2). Then for any $K, L \subset Q$, we have $([K],[L]) = |K \cap L|$. Denote by $\mathbf{1}_n$ the uniform stochastic vector in $\mathbb{R}^n$, that is, the vector with all entries equal to $\frac{1}{n}$. Then the fact that a word w extends a subset $K \subset Q$ (that is, the inequality $|K| < |K\,.\,w^{-1}|$) can be rewritten as $([K], \mathbf{1}_n) < ([w]^T[K], \mathbf{1}_n)$.
Thus, the extension method amounts to finding a state q, a letter a, and a sequence of words $w_1, w_2, \dots, w_d$ such that

$$\frac{1}{n} = ([q], \mathbf{1}_n) < ([a]^T[q], \mathbf{1}_n) < ([w_1 a]^T[q], \mathbf{1}_n) < \dots \\ \dots < ([w_d \cdots w_2 w_1 a]^T[q], \mathbf{1}_n) = 1.$$

Here $d \leq n-2$ because at each step the inner product increases by at least $\frac{1}{n}$. The problem is that so far we don't any linear bound for the lengths of the w_i's.

Extensibility in Linear Terms

For vectors $g_1, g_2 \in \mathbb{R}^n$, we denote their usual inner product by (g_1, g_2). Then for any $K, L \subset Q$, we have $([K], [L]) = |K \cap L|$. Denote by $\mathbf{1}_n$ the uniform stochastic vector in $\mathbb{R}^n$, that is, the vector with all entries equal to $\frac{1}{n}$. Then the fact that a word w extends a subset $K \subset Q$ (that is, the inequality $|K| < |K \,.\, w^{-1}|$) can be rewritten as $([K], \mathbf{1}_n) < ([w]^T[K], \mathbf{1}_n)$.

Thus, the extension method amounts to finding a state q, a letter a, and a sequence of words $w_1, w_2, \dots, w_d$ such that

$$\frac{1}{n} = ([q], \mathbf{1}_n) < ([a]^T[q], \mathbf{1}_n) < ([w_1a]^T[q], \mathbf{1}_n) < \dots$$
$$\dots < ([w_d \cdots w_2w_1a]^T[q], \mathbf{1}_n) = 1.$$

Here $d \leq n - 2$ because at each step the inner product increases by at least $\frac{1}{n}$. The problem is that so far we don't any linear bound for the lengths of the w_i's.

Jungers's Dualization

Raphaël Jungers (The synchronizing probability function of an automaton, SIAM J. Discrete Math. 26 (2011) 177–192) has suggested an interesting idea that in our notation can be described as follows: one should substitute the uniform stochastic vector $\mathbf{1}_n$ by an adaptive positive stochastic vector p which can depend on both the automaton $\mathscr{A}$ and the given proper subset $K \subset Q$ but has the property that there exists a word v of length at most $|Q|$ such that $([v]^T[K], p) > ([K], p)$. Jungers has explored this idea using techniques from linear programming and has proved that such a positive stochastic vector indeed exists for every synchronizing automaton and every proper subset.

Warning: Here we encounter a dual problem since it is not clear how to find a linear upper bound for the number of extension steps.

Jungers's Dualization

Raphaël Jungers (The synchronizing probability function of an automaton, SIAM J. Discrete Math. 26 (2011) 177–192) has suggested an interesting idea that in our notation can be described as follows: one should substitute the uniform stochastic vector $\mathbf{1}_n$ by an adaptive positive stochastic vector p which can depend on both the automaton $\mathscr{A}$ and the given proper subset $K \subset Q$ but has the property that there exists a word v of length at most $|Q|$ such that $([v]^T[K], p) > ([K], p)$. Jungers has explored this idea using techniques from linear programming and has proved that such a positive stochastic vector indeed exists for every synchronizing automaton and every proper subset.

Warning: Here we encounter a dual problem since it is not clear how to find a linear upper bound for the number of extension steps.

Raphaël Jungers (The synchronizing probability function of an automaton, SIAM J. Discrete Math. 26 (2011) 177–192) has suggested an interesting idea that in our notation can be described as follows: one should substitute the uniform stochastic vector $\mathbf{1}_n$ by an adaptive positive stochastic vector p which can depend on both the automaton $\mathscr{A}$ and the given proper subset $K \subset Q$ but has the property that there exists a word v of length at most $|Q|$ such that $([v]^T[K], p) > ([K], p)$. Jungers has explored this idea using techniques from linear programming and has proved that such a positive stochastic vector indeed exists for every synchronizing automaton and every proper subset.

Warning: Here we encounter a dual problem since it is not clear how to find a linear upper bound for the number of extension steps.

Markov Chains

Assume that $\Sigma = \{a_1, a_2, \ldots, a_k\}$. Each positive stochastic vector $\pi \in \mathbb{R}^k_+$ defines a probability distribution on Σ. Consider a process in which an agent randomly walks on the underlying graph of $\mathscr{A}$, choosing for each move an edge labeled a_i with probability $p(a_i)$. This is a Markov chain with the transition matrix

$$S = S(\mathscr{A}, \pi) = \sum_{i=1}^{k} p(a_i)[a_i].$$

By basic properties of Markov chains, there exists the stationary distribution $\alpha \in \mathbb{R}^n_+$ of this Markov chain, that is, a unique positive stochastic vector satisfying $S\alpha = \alpha$.

Markov Chains

Assume that $\Sigma = \{a_1, a_2, \ldots, a_k\}$. Each positive stochastic vector $\pi \in \mathbb{R}^k_+$ defines a probability distribution on Σ. Consider a process in which an agent randomly walks on the underlying graph of $\mathscr{A}$, choosing for each move an edge labeled a_i with probability $p(a_i)$. This is a Markov chain with the transition matrix

$$S = S(\mathscr{A}, \pi) = \sum_{i=1}^{k} p(a_i)[a_i].$$

By basic properties of Markov chains, there exists the stationary distribution $\alpha \in \mathbb{R}^n_+$ of this Markov chain, that is, a unique positive stochastic vector satisfying $S\alpha = \alpha$.

Markov Chains

Assume that $\Sigma = \{a_1, a_2, \ldots, a_k\}$. Each positive stochastic vector $\pi \in \mathbb{R}^k_+$ defines a probability distribution on Σ. Consider a process in which an agent randomly walks on the underlying graph of $\mathscr{A}$, choosing for each move an edge labeled a_i with probability $p(a_i)$.

This is a Markov chain with the transition matrix

$$S = S(\mathscr{A}, \pi) = \sum_{i=1}^{k} p(a_i)[a_i].$$

By basic properties of Markov chains, there exists the stationary distribution $\alpha \in \mathbb{R}^n_+$ of this Markov chain, that is, a unique positive stochastic vector satisfying $S\alpha = \alpha$.

Markov Chains

Assume that $\Sigma = \{a_1, a_2, \ldots, a_k\}$. Each positive stochastic vector $\pi \in \mathbb{R}^k_+$ defines a probability distribution on Σ. Consider a process in which an agent randomly walks on the underlying graph of $\mathscr{A}$, choosing for each move an edge labeled a_i with probability $p(a_i)$. This is a Markov chain with the transition matrix

$$S = S(\mathscr{A}, \pi) = \sum_{i=1}^{k} p(a_i)[a_i].$$

By basic properties of Markov chains, there exists the stationary distribution $\alpha \in \mathbb{R}^n_+$ of this Markov chain, that is, a unique positive stochastic vector satisfying $S\alpha = \alpha$.

Markov Chains

Assume that $\Sigma = \{a_1, a_2, \ldots, a_k\}$. Each positive stochastic vector $\pi \in \mathbb{R}^k_+$ defines a probability distribution on Σ. Consider a process in which an agent randomly walks on the underlying graph of $\mathscr{A}$, choosing for each move an edge labeled a_i with probability $p(a_i)$. This is a Markov chain with the transition matrix

$$S = S(\mathscr{A}, \pi) = \sum_{i=1}^{k} p(a_i)[a_i].$$

By basic properties of Markov chains, there exists the stationary distribution $\alpha \in \mathbb{R}^n_+$ of this Markov chain, that is, a unique positive stochastic vector satisfying $S\alpha = \alpha$.

Berlinkov's Result

Theorem (Berlinkov, 2012)

Let $\mathscr{A}$ be a synchronizing automaton with n states and k letters, $\pi \in \mathbb{R}^k_+$ a positive stochastic vector, and α the stationary distribution of the Markov chain with the transition matrix $S(\mathscr{A}, \pi)$. Then there exist a state q, a letter a, and a sequence of words $w_1, w_2, \dots, w_d$ of length at most n such that

$$([q], \alpha) < ([a]^T[q], \alpha) < ([w_1 a]^T[q], \alpha) < \dots$$
$$\dots < ([w_d \cdots w_2 w_1 a]^T[q], \alpha) = 1.$$

Berlinkov's Result

Theorem (Berlinkov, 2012)

Let $\mathscr{A}$ be a synchronizing automaton with n states and k letters, $\pi \in \mathbb{R}_+^k$ a positive stochastic vector, and α the stationary distribution of the Markov chain with the transition matrix $S(\mathscr{A}, \pi)$. Then there exist a state q, a letter a, and a sequence of words $w_1, w_2, \ldots, w_d$ of length at most n such that

$$([q], \alpha) < ([a]^T[q], \alpha) < ([w_1 a]^T[q], \alpha) < \ldots$$
$$\cdots < ([w_d \cdots w_2 w_1 a]^T[q], \alpha) = 1.$$

An immediate application: a new proof of the Černý conjecture for automata with Eulerian digraphs.

Berlinkov's Result

Theorem (Berlinkov, 2012)

Let $\mathscr{A}$ be a synchronizing automaton with n states and k letters, $\pi \in \mathbb{R}^k_+$ a positive stochastic vector, and α the stationary distribution of the Markov chain with the transition matrix $S(\mathscr{A}, \pi)$. Then there exist a state q, a letter a, and a sequence of words $w_1, w_2, \dots, w_d$ of length at most n such that

$$([q], \alpha) < ([a]^T[q], \alpha) < ([w_1 a]^T[q], \alpha) < \dots$$
$$\dots < ([w_d \cdots w_2 w_1 a]^T[q], \alpha) = 1.$$

An immediate application: a new proof of the Černý conjecture for automata with Eulerian digraphs. In this case the matrix $S(\mathscr{A}, \pi)$ is doubly stochastic whence the uniform vector $\mathbf{1}_n$ is its stationary distribution and $d \leq n - 2$.