Finitely Generated Synchronizing Automata

Elena V. Pribavkina, Emanuele Rodaro

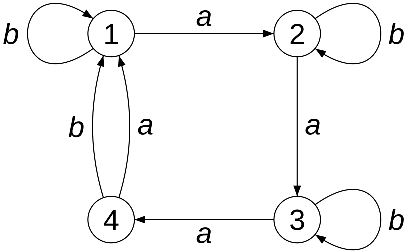
Ural State University, Ekaterinburg, Russia
Università dell'Insubria, Italy

October 6, 2008

Synchronizing Automata

- A deterministic finite automaton (DFA) is a triple $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$.
- A DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ is called *synchronizing* if there is a word w whose action *resets* $\mathscr{A}$, that is, leaves the automaton in one particular state no matter which state in Q it started at: $\delta(q, w) = \delta(q', w)$ for all $q, q' \in Q$. $|Q\,.\,w| = 1$. Here $Q\,.\,v = \{\delta(q, v) \mid q \in Q\}$.
- Any such w is called *synchronizing* or *reset* word for $\mathscr{A}$.

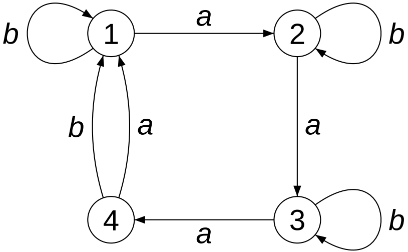
Synchronizing Automata – An Example



A reset word is ba^3ba^3b. Applying this word at any state brings the automaton to the state 1.

In fact it is the shortest reset word for this automaton.

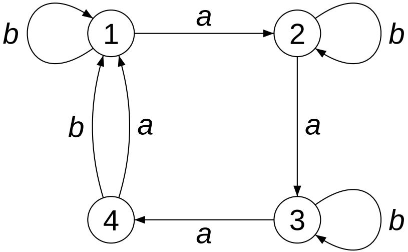
Synchronizing Automata – An Example



A reset word is ba^3ba^3b. Applying this word at any state brings the automaton to the state 1.

In fact it is the shortest reset word for this automaton.

Synchronizing Automata – An Example



A reset word is ba^3ba^3b. Applying this word at any state brings the automaton to the state 1.

In fact it is the shortest reset word for this automaton.

Černý's Conjecture

Suppose a synchronizing automaton has n states. What is the length of the shortest synchronizing word?

In 1964 Jan Černý found an infinite series of n-state reset automata whose shortest reset word has length $(n-1)^2$.

He conjectured that this is the worst case, that is

Any synchronizing automaton with n states has a reset word of length at most $(n-1)^2$.

Černý's conjecture has been proved for many particular classes of synchronizing automata:

automata with zero, monotonic automata, aperiodic automata, automata whose underlying digraph is Eulerian, etc.

A New Class of Synchronizing Automata

- We introduce and characterize a new class of synchronizing automata
- We show that the length of the shortest reset word for any n-state automaton in this class is a linear function of n.
- We also study the complexity of determining whether a given synchronizing automaton belongs to this class.

A New Class of Synchronizing Automata

- We introduce and characterize a new class of synchronizing automata
- We show that the length of the shortest reset word for any n-state automaton in this class is a linear function of n.
- We also study the complexity of determining whether a given synchronizing automaton belongs to this class.

A New Class of Synchronizing Automata

- We introduce and characterize a new class of synchronizing automata
- We show that the length of the shortest reset word for any n-state automaton in this class is a linear function of n.
- We also study the complexity of determining whether a given synchronizing automaton belongs to this class.

Minimal Synchronizing Words

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a synchronizing DFA.
$\mathscr{L}(\mathscr{A})$ denotes the language of all words synchronizing $\mathscr{A}$.

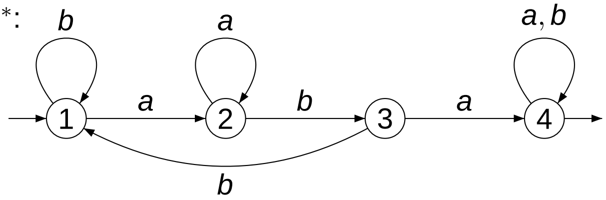
A synchronizing word v is said to be *minimal* if none of its proper prefixes nor suffixes is synchronizing.

The language $\mathscr{L}(\mathscr{A})$ of all synchronizing words is a two-sided ideal generated by the language of all minimal synchronizing words:

$$\mathscr{L}(\mathscr{A}) = \Sigma^* \mathscr{L}_{min}(\mathscr{A}) \Sigma^*.$$

We consider the class **FG** of synchronizing automata whose language of minimal synchronizing words is finite. Such automata are referred to as *finitely generated synchronizing automata*.

The minimal automaton $\mathscr{A}_{aba}$ recognizing the language $\Sigma^* aba \Sigma^*$:



$\mathscr{L}(\mathscr{A}_{aba}) = \Sigma^* aba \Sigma^* \Rightarrow \mathscr{L}_{min}(\mathscr{A}_{aba}) = \{aba\} \Rightarrow \mathscr{A}_{aba} \in$ **FG**.

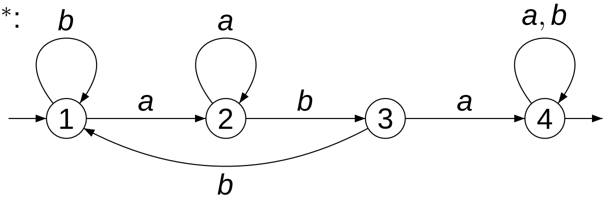
For any word $w \in \Sigma^*$ $\mathscr{A}_w \in$ **FG**.

$\mathscr{A}_w$ has $n = |w| + 1$ states, hence its shortest synchronizing word has length $n - 1$.

In general, the minimal automaton recognizing the language $\Sigma^* M \Sigma^*$ for a finite language M is in **FG**.

Finitely Generated Synchronizing Automata

The minimal automaton $\mathscr{A}_{aba}$ recognizing the language $\Sigma^* aba \Sigma^*$:



$\mathscr{L}(\mathscr{A}_{aba}) = \Sigma^* aba \Sigma^* \Rightarrow \mathscr{L}_{min}(\mathscr{A}_{aba}) = \{aba\} \Rightarrow \mathscr{A}_{aba} \in$ **FG**.

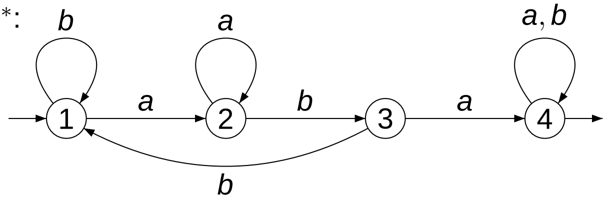
For any word $w \in \Sigma^*$ $\mathscr{A}_w \in$ **FG**.

$\mathscr{A}_w$ has $n = |w| + 1$ states, hence its shortest synchronizing word has length $n - 1$.

In general, the minimal automaton recognizing the language $\Sigma^* M \Sigma^*$ for a finite language M is in **FG**.

Finitely Generated Synchronizing Automata

The minimal automaton $\mathscr{A}_{aba}$ recognizing the language $\Sigma^* aba \Sigma^*$:



$\mathscr{L}(\mathscr{A}_{aba}) = \Sigma^* aba \Sigma^* \Rightarrow \mathscr{L}_{min}(\mathscr{A}_{aba}) = \{aba\} \Rightarrow \mathscr{A}_{aba} \in$ **FG**.

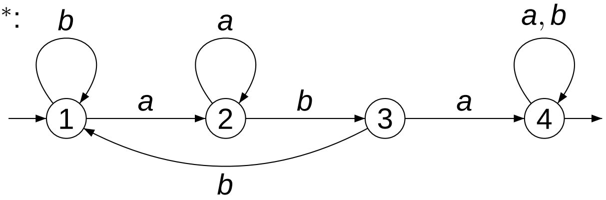
For any word $w \in \Sigma^*$ $\mathscr{A}_w \in$ **FG**.

$\mathscr{A}_w$ has $n = |w| + 1$ states, hence its shortest synchronizing word has length $n - 1$.

In general, the minimal automaton recognizing the language $\Sigma^* M \Sigma^*$ for a finite language M is in **FG**.

Finitely Generated Synchronizing Automata

The minimal automaton $\mathscr{A}_{aba}$ recognizing the language $\Sigma^* aba \Sigma^*$:



$\mathscr{L}(\mathscr{A}_{aba}) = \Sigma^* aba \Sigma^* \Rightarrow \mathscr{L}_{min}(\mathscr{A}_{aba}) = \{aba\} \Rightarrow \mathscr{A}_{aba} \in$ **FG**.

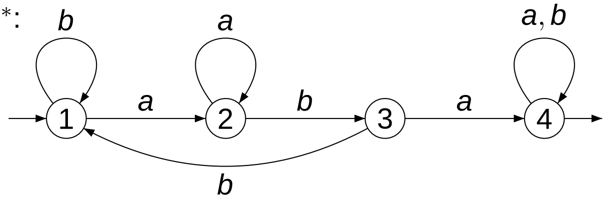
For any word $w \in \Sigma^*$ $\mathscr{A}_w \in$ **FG**.

$\mathscr{A}_w$ has $n = |w| + 1$ states, hence its shortest synchronizing word has length $n - 1$.

In general, the minimal automaton recognizing the language $\Sigma^* M \Sigma^*$ for a finite language M is in **FG**.

Finitely Generated Synchronizing Automata

The minimal automaton $\mathscr{A}_{aba}$ recognizing the language $\Sigma^* aba \Sigma^*$:



$\mathscr{L}(\mathscr{A}_{aba}) = \Sigma^* aba \Sigma^* \Rightarrow \mathscr{L}_{min}(\mathscr{A}_{aba}) = \{aba\} \Rightarrow \mathscr{A}_{aba} \in$ **FG**.

For any word $w \in \Sigma^*$ $\mathscr{A}_w \in$ **FG**.

$\mathscr{A}_w$ has $n = |w| + 1$ states, hence its shortest synchronizing word has length $n - 1$.

In general, the minimal automaton recognizing the language $\Sigma^* M \Sigma^*$ for a finite language M is in **FG**.

Characterization of **FG**

- $T \subseteq Q$ is *reachable* if there is $v \in \Sigma^*$ with $T = Q \,.\, v$.
- $\mathcal{S}(T)$ is the set of all words *stabilizing* T:

$$\mathcal{S}(T) = \{w \in \Sigma^* \mid T \,.\, w = T\}$$

- By $\mathcal{R}(T)$ we denote the set of all words bringing T to a singleton:
$$\mathcal{R}(T) = \{w \in \Sigma^* \mid |T \,.\, w| = 1\}$$
- Let $w \in \Sigma^*$, by $m(w) \subseteq Q$ we denote the maximal fixed set with respect to w.

Theorem 1

A synchronizing automaton $\mathscr{A}$ is in **FG** iff for any reachable subset $T \subseteq Q$ with $1 < |T| < |Q|$, for each $w \in \mathcal{S}(T)$

$$\mathcal{R}(T) = \mathcal{R}(m(w))$$

Characterization of **FG**

- $T \subseteq Q$ is *reachable* if there is $v \in \Sigma^*$ with $T = Q \,.\, v$.
- $\mathcal{S}(T)$ is the set of all words *stabilizing* T:

$$\mathcal{S}(T) = \{w \in \Sigma^* \mid T \,.\, w = T\}$$

- By $\mathcal{R}(T)$ we denote the set of all words bringing T to a singleton:

$$\mathcal{R}(T) = \{w \in \Sigma^* \mid |T \,.\, w| = 1\}$$

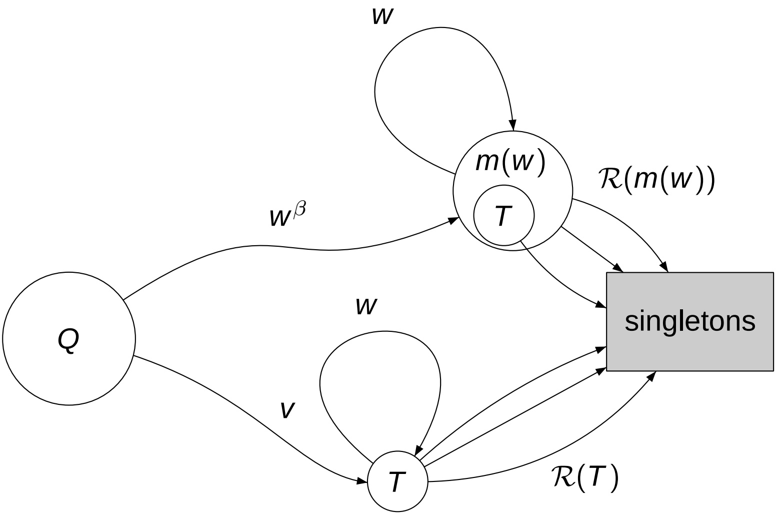
- Let $w \in \Sigma^*$, by $m(w) \subseteq Q$ we denote the maximal fixed set with respect to w.

Theorem 1

A synchronizing automaton $\mathscr{A}$ is in **FG** iff for any reachable subset $T \subseteq Q$ with $1 < |T| < |Q|$, for each $w \in \mathcal{S}(T)$

$$\mathcal{R}(T) = \mathcal{R}(m(w))$$

Visualization



Corollary

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a synchronizing automaton such that there is a letter $a \in \Sigma$ with $Q\,.\,a = Q$ and there are no letters $b \in \Sigma$ with $|Q\,.\,b| = 1$. Then $\mathscr{L}_{min}(\mathscr{A})$ is infinite.

Theorem 2

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a finitely generated synchronizing automaton with n states. There is a synchronizing word of length at most $3n - 5$.

Remark

Take any letter $a \in \Sigma$, then either a^k or $a^k \tau a^k$ is synchronizing ($k \leq n - |m(a)|$ and $|\tau| \leq n - 1$).

Corollary

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a synchronizing automaton such that there is a letter $a \in \Sigma$ with $Q \,.\, a = Q$ and there are no letters $b \in \Sigma$ with $|Q \,.\, b| = 1$. Then $\mathscr{L}_{min}(\mathscr{A})$ is infinite.

Theorem 2

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a finitely generated synchronizing automaton with n states. There is a synchronizing word of length at most $3n - 5$.

Remark

Take any letter $a \in \Sigma$, then either a^k or $a^k \tau a^k$ is synchronizing ($k \leq n - |m(a)|$ and $|\tau| \leq n - 1$).

Corollary

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a synchronizing automaton such that there is a letter $a \in \Sigma$ with $Q \,.\, a = Q$ and there are no letters $b \in \Sigma$ with $|Q \,.\, b| = 1$. Then $\mathscr{L}_{min}(\mathscr{A})$ is infinite.

Theorem 2

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a finitely generated synchronizing automaton with n states. There is a synchronizing word of length at most $3n - 5$.

Remark

Take any letter $a \in \Sigma$, then either a^k or $a^k \tau a^k$ is synchronizing ($k \leq n - |m(a)|$ and $|\tau| \leq n - 1$).

Decidability of FINITENESS Problem

FINITENESS

Input: A synchronizing DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$.
Question: Is $\mathscr{A}$ finitely generated?

FINITENESS Problem is decidable:

- $\mathscr{L}_{min}(\mathscr{A}) = \mathscr{L}(\mathscr{A}) \setminus (\Sigma\mathscr{L}(\mathscr{A}) \cup \mathscr{L}(\mathscr{A})\Sigma)$.
- The language $\mathscr{L}(\mathscr{A})$ is regular (it is recognized by the power automaton $\mathcal{P}(\mathscr{A})$ with Q as an initial state and singletons as terminal ones).
- If $\mathscr{A}$ has n states, then $\mathcal{P}(\mathscr{A})$ has at most $2^n - 1$ states.
- $\mathscr{L}_{min}(\mathcal{A})$ is recognized by an automaton with $O(2^{3n})$ states, thus checking the finiteness takes $O(2^{6n})$.

Decidability of FINITENESS Problem

FINITENESS

Input: A synchronizing DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$.
Question: Is $\mathscr{A}$ finitely generated?

FINITENESS Problem is decidable:

- $\mathscr{L}_{min}(\mathscr{A}) = \mathscr{L}(\mathscr{A}) \setminus (\Sigma\mathscr{L}(\mathscr{A}) \cup \mathscr{L}(\mathscr{A})\Sigma)$.
- The language $\mathscr{L}(\mathscr{A})$ is regular (it is recognized by the power automaton $\mathcal{P}(\mathscr{A})$ with Q as an initial state and singletons as terminal ones).
- If $\mathscr{A}$ has n states, then $\mathcal{P}(\mathscr{A})$ has at most $2^n - 1$ states.
- $\mathscr{L}_{min}(\mathcal{A})$ is recognized by an automaton with $O(2^{3n})$ states, thus checking the finiteness takes $O(2^{6n})$.

Another Algorithm

Our characterization gives rise to the following algorithm FINCHECK($\mathscr{A}$):

- From a synchronizing $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ build the power automaton $\mathcal{P}(\mathscr{A}) = \langle \mathcal{Q}, \Sigma, \delta \rangle$.
- For each T of $\mathcal{Q}$ do:
- For each H of $\mathcal{Q}$ with $T \subseteq H$ do:
- If $\mathcal{S}(H) \cap \mathcal{S}(T) \neq \varnothing$, then
  - If $\mathcal{R}(T) \neq \mathcal{R}(H)$, then exit and return NO
- Otherwise exit and return YES

The cost of this algorithm is $O\left(2^{2n}3^n\right)$, which is slightly better than the straight-forward one, but still exponential.
Find the complexity class of FINITENESS

Complexity of FINITENESS

- FINITENESS is in PSPACE (Pawel Gawrychowski).
- We have proved that FINITENESS is in co-NP-hard.

Complexity of FINITENESS

- FINITENESS is in PSPACE (Pawel Gawrychowski).
- We have proved that FINITENESS is in co-NP-hard.

Co-NP-Hardness: Strategy

- Introduce an auxiliary problem

REACHABILITY

Input: A DFA $\mathcal{A} = \langle Q, \Sigma, \delta \rangle$ and a subset $H \subseteq Q$.
Question: Is there a word $w \in \Sigma^*$ such that $Q \,.\, w = H$?

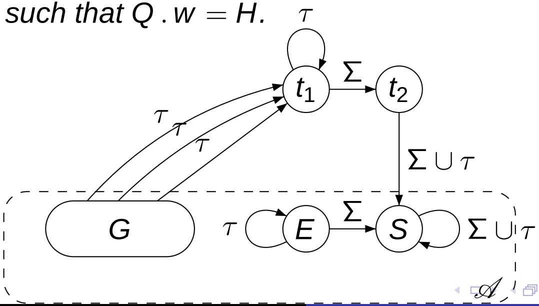
- Reduce a particular set of instances I of REACHABILITY to instances of the complement of FINITENESS.
- Reduce any instance of SAT to an instance belonging to the set I.

A particular class $\mathcal{N}$ of nilpotent automata $\mathscr{A}$ with a sink state S and a non-empty set $\mathcal{E}_{\mathscr{A}}$ of states E satisfying $E\,.\,\Sigma = S$. We define the set I of instances of REACHABILITY:

$$I = \{(\mathscr{A}, H) \mid \mathscr{A} \in \mathcal{N}, H = \{S, E\}, E \in \mathcal{E}_{\mathscr{A}}\}.$$

Proposition

Let $(\mathscr{A}, H) \in I$. Then there is a synchronizing automaton $\mathscr{A}'$ such that the language $\mathscr{L}_{min}(\mathscr{A}')$ is infinite iff there exists $w \in \Sigma^+$ such that $Q\,.\,w = H$.



Co-NP-hardness: Second Reduction

- Let $\chi = \{c_1, \ldots, c_p\}$ be a set of clauses over n variables $X_1, \ldots, X_n$.
- Let $\Sigma_n = \{a_1, b_1, \ldots, a_n, b_n\}$, $\gamma_i = \{a_i, b_i\}$.
- $x_i \in \gamma_i = \{a_i, b_i\}$, $\chi(x_i) = \{c_{i_1}, \ldots, c_{i_k}\}$ is the set of clauses containing positive literal X_i if $x_i = a_i$, and of clauses containing negative literal $\neg X_i$ if $x_i = b_i$.
- We say that the set $x_1, \ldots, x_n$ with $x_i \in \gamma_i$ is a *satisfiable assignment* for χ iff:

$$\bigcup_{i=1}^{n} \chi(x_i) = \chi$$

Co-NP-hardness: Second Reduction

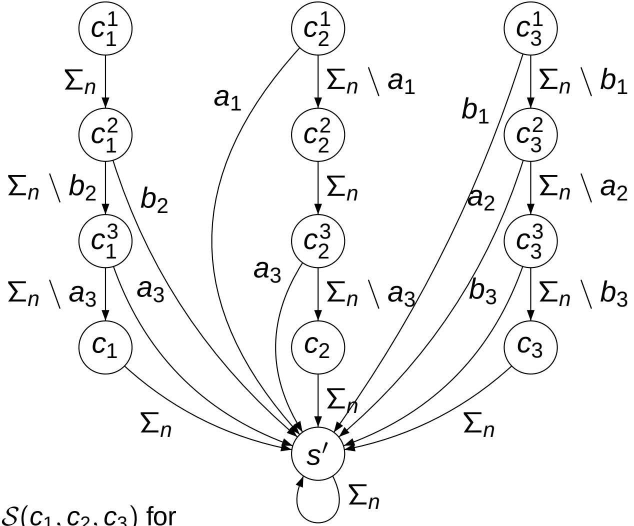


Figure: $\mathcal{S}(c_1, c_2, c_3)$ for
$c_1 = \neg X_2 \vee X_3, c_2 = X_3 \vee X_1, c_3 = X_2 \vee \neg X_1 \vee \neg X_3$.

Co-NP-hardness: Second Reduction

Proposition

Let $\mathcal{S}(c_1, \ldots, c_p)$ be constructed as above and let $w = x_1 \ldots x_n \in \Sigma_n^+$ with $x_i \in \gamma_i$ for $i = 1, \ldots, n$. Then w resets $\mathcal{S}(c_1, \ldots, c_p)$ iff $x_1, \ldots, x_n$ is a satisfiable assignment.

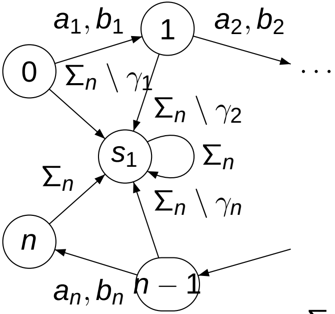


Figure: Automaton A_n.

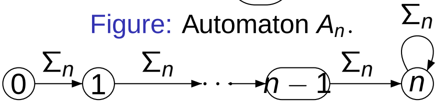


Figure: Automaton B_n

Proposition

The automaton $\mathcal{V}_n = A_n \times B_n = \langle Q_{A,B}, \Sigma_n, \delta_{A,B} \rangle$ is nilpotent with the sink state $s = (s_1, n)$ and possesses a state e such that $e\,.\,\Sigma_n = s$. Moreover $Q_{A,B}\,.\,w = \{s, e\}$ iff $w = x_1 \ldots x_n$ with $x_i \in \gamma_i$.

Proposition

Let $c_1, \ldots, c_p$ be clauses over $n \geq 2$ variables. The automaton $\mathscr{A} = \mathcal{V}_n \times \mathcal{S}(c_1, \ldots, c_p) = \langle Q, \Sigma_n, \delta \rangle$ belongs to $\mathcal{N}$. Moreover, putting $S = (s, s')$, $E = (e, s')$, and $H = \{S, E\}$, we have $(\mathscr{A}, H) \in I$ and there is a word $w \in \Sigma_n^+$ such that $Q \,.\, w = H$ iff the boolean formula $\wedge_{i=1}^{p} c_i$ is satisfiable.

Theorem

The problem FINITENESS is co-NP-hard.

Open Problems

- What is the precise complexity class of FINITENESS? (Is it in co-NP? If for some $w \in \mathcal{S}(T)$, $\mathcal{R}(m(w)) \subsetneq \mathcal{R}(T)$, then we have to show that there is a $v \in \mathcal{R}(T) \setminus \mathcal{R}(m(w))$ such that $|v|$ is polynomially bounded in the number of states in Q.)
- The characterization is given in terms of the power automaton. Is there a characterization in terms of the transition monoid of $\mathscr{A}$?
- If $\mathscr{L}_{min}(\mathscr{A})$ is finite, give an upper bound for the number of generators $|\mathscr{L}_{min}(\mathscr{A})|$.
- If $\mathscr{L}_{min}(\mathscr{A})$ is finite, give a bound for the length of the longest word in $\mathscr{L}_{min}(\mathscr{A})$.
- Is the bound $3n - 5$ for the length of the shortest synchronizing word for the class **FG** precise?

THANK YOU!