The University of Manchester

# An Overview on Algorithmic Results for Automaton (Semi)Groups

Jan Philipp **Wächter**

Department of Mathematics
University of Manchester

This visit is funded by EPSRC project EP/Y008626/1.

23 March 2026

# Algorithmic Problems in Group Theory

Max Dehn's three fundamental problems of algorithmic group theory:

# Algorithmic Problems in Group Theory

Max Dehn's three fundamental problems of algorithmic group theory:

## Definition (Word Problem)

**Constant:** a group $G$
**Input:** a group element $g \in G$
**Question:** is $g = \mathbb{1}$?

# Algorithmic Problems in Group Theory

Max Dehn's three fundamental problems of algorithmic group theory:

**Definition (Word Problem)**

**Constant:** a group $G$
**Input:** a group element $g \in G$
**Question:** is $g = \mathbb{1}$?

**Definition (Conjugacy Problem)**

**Constant:** a group $G$
**Input:** two group elements $g, h \in G$
**Question:** $\exists k \in G : g = k^{-1} h k$ (i. e. are they conjugate)?

# Algorithmic Problems in Group Theory

Max Dehn's three fundamental problems of algorithmic group theory:

## Definition (Word Problem)

**Constant:** a group $G$
**Input:** a group element $g \in G$
**Question:** is $g = \mathbb{1}$?

## Definition (Conjugacy Problem)

**Constant:** a group $G$
**Input:** two group elements $g, h \in G$
**Question:** $\exists k \in G : g = k^{-1}hk$ (i. e. are they conjugate)?

## Definition (Isomorphism Problem)

**Input:** two groups $G$ and $H$
**Question:** are $G$ and $H$ isomorphic?

# Algorithmic Problems in Group Theory

Max Dehn's three fundamental problems of algorithmic group theory:

## Definition (Word Problem)

**Constant:** a group $G$
**Input:** a group element $g \in G$
**Question:** is $g = \mathbb{1}$?

## Definition (Conjugacy Problem)

**Constant:** a group $G$
**Input:** two group elements $g, h \in G$
**Question:** $\exists k \in G : g = k^{-1}hk$
(i. e. are they conjugate)?

## Definition (Isomorphism Problem)

**Input:** two groups $G$ and $H$
**Question:** are $G$ and $H$ isomorphic?

How can we encode an infinite (semi)group?

# Encoding (Semi)Groups

- traditional presentation: $\langle q_1, \dots, q_n \mid \ell_1 = r_1, \dots, \ell_m = r_m \rangle$
  $Q = \{q_1, \dots, q_n\}$: generators, $(\ell_1, r_1), \dots (\ell_m, r_m) \in Q^+ \times Q^+$: relations

# Encoding (Semi)Groups

- traditional presentation: $\langle q_1, \dots, q_n \mid \ell_1 = r_1, \dots, \ell_m = r_m \rangle$
  $Q = \{q_1, \dots, q_n\}$: generators, $(\ell_1, r_1), \dots (\ell_m, r_m) \in Q^+ \times Q^+$: relations
  $\rightsquigarrow$ if both sets are finite: finitely presented (semi)group

# Encoding (Semi)Groups

- traditional presentation: $\langle q_1, \ldots, q_n \mid \ell_1 = r_1, \ldots, \ell_m = r_m \rangle$
  $Q = \{q_1, \ldots, q_n\}$: generators, $(\ell_1, r_1), \ldots (\ell_m, r_m) \in Q^+ \times Q^+$: relations
  $\rightsquigarrow$ if both sets are finite: finitely presented (semi)group
  Here: Any "reasonable" property is undecidable (Markov properties/Adian-Rabin theorem)

# Encoding (Semi)Groups

- traditional presentation: $\langle q_1, \ldots, q_n \mid \ell_1 = r_1, \ldots, \ell_m = r_m \rangle$
  $Q = \{q_1, \ldots, q_n\}$: generators, $(\ell_1, r_1), \ldots (\ell_m, r_m) \in Q^+ \times Q^+$: relations
  $\rightsquigarrow$ if both sets are finite: finitely presented (semi)group
  Here: Any "reasonable" property is undecidable (Markov properties/Adian-Rabin theorem)
- Alternatives: Cayley table, permutations, invertiable matrices, ...

# Encoding (Semi)Groups

- traditional presentation: $\langle q_1, \dots, q_n \mid \ell_1 = r_1, \dots, \ell_m = r_m \rangle$
  $Q = \{q_1, \dots, q_n\}$: generators, $(\ell_1, r_1), \dots (\ell_m, r_m) \in Q^+ \times Q^+$: relations
  $\rightsquigarrow$ if both sets are finite: finitely presented (semi)group
  Here: Any "reasonable" property is undecidable (Markov properties/Adian-Rabin theorem)
- Alternatives: Cayley table, permutations, invertiable matrices, ...
- We will use: automata

# Why use automata?

# Why use automata?

- Many examples of (in particular) groups with interesting properties arise in this way

# Why use automata?

- Many examples of (in particular) groups with interesting properties arise in this way and
- it allows for a finite description of possibly non-finitely presented (semi)groups.

# Why use automata?

- Many examples of (in particular) groups with interesting properties arise in this way and
- it allows for a finite description of possibly non-finitely presented (semi)groups.

  ⇝ This makes them algorithmically interesting!

# Why use automata?

- Many examples of (in particular) groups with interesting properties arise in this way and
- it allows for a finite description of possibly non-finitely presented (semi)groups.
  $\rightsquigarrow$ This makes them algorithmically interesting!

Let's look at a famous example!

# Example: Grigorchuk's Group...

# Example: Grigorchuk's Group…

- …is the historically first
  example of a group of intermediate growth
  (i. e. subexponential but superpolynomial).
  "Milnor Problem"

# Example: Grigorchuk's Group...

- ...is the historically first example of a group of intermediate growth (i. e. subexponential but superpolynomial).
  "Milnor Problem"
- ...is a Burnside group.
  "Burnside Problem"

# Example: Grigorchuk's Group...

- ...is the historically first example of a group of intermediate growth (i. e. subexponential but superpolynomial).
  "Milnor Problem"
- ...is a Burnside group.
  "Burnside Problem"
- ...is amenable but not elementary amenable. "Day Problem"

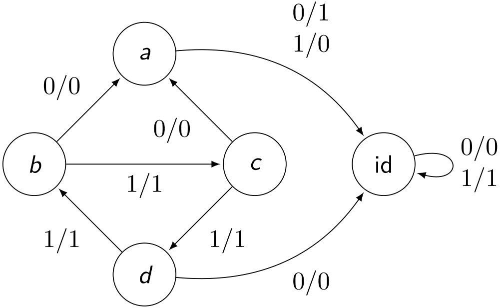
# Example: Grigorchuk's Group…

- …is the historically first example of a group of intermediate growth (i. e. subexponential but superpolynomial).
  "Milnor Problem"
- …is a Burnside group.
  "Burnside Problem"
- …is amenable but not elementary amenable. "Day Problem"
- …is not finitely presented.

# Example: Grigorchuk's Group...

- ...is the historically first example of a group of intermediate growth (i. e. subexponential but superpolynomial).
  "Milnor Problem"
- ...is a Burnside group.
  "Burnside Problem"
- ...is amenable but not elementary amenable. "Day Problem"
- ...is not finitely presented.

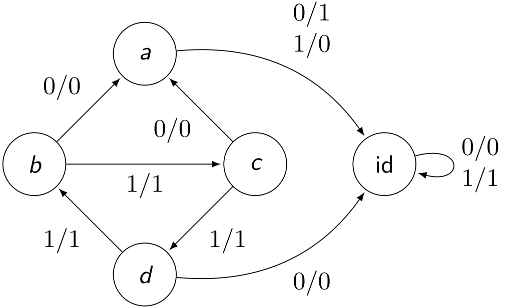
- But: It is generated by the automaton:



# Example: Grigorchuk's Group…

- …is the historically first example of a group of intermediate growth (i. e. subexponential but superpolynomial).
  "Milnor Problem"
- …is a Burnside group.
  "Burnside Problem"
- …is amenable but not elementary amenable. "Day Problem"
- …is not finitely presented.

- But: It is generated by the automaton:



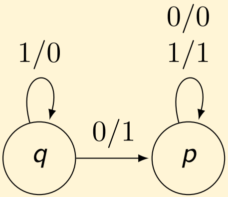

"How does this work?"

⇝ next slides

# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

## Example

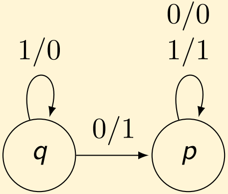


# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,

**Example**

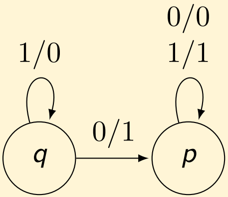


# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

## Example



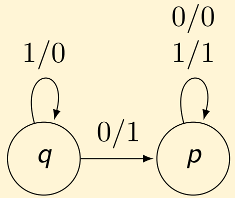
# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

transducer

## Example



# Automata

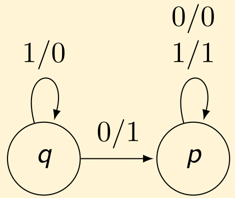
Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

transducer

- without final or initial states.

## Example



# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

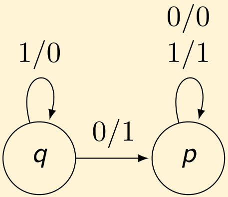
- finite-state,
- letter-to-letter

transducer

- without final or initial states.

An automaton may be:

**Example**



# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

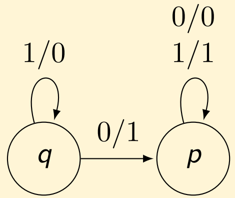
transducer

- without final or initial states.

An automaton may be:

- deterministic

## Example



# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

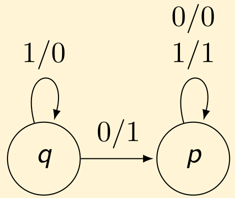
transducer

- without final or initial states.

An automaton may be:

- deterministic
- complete

## Example



# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

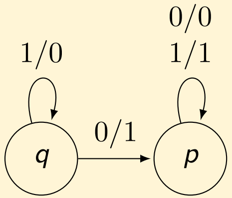
transducer

- without final or initial states.

An automaton may be:

- deterministic
- complete
- invertible

**Example**



# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

transducer

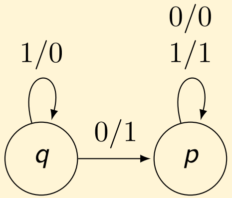
- without final or initial states.

An automaton may be:

- deterministic ✓
- complete (✓)
- invertible –

$\mathscr{S}$-automaton
"semigroup"

## Example



# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

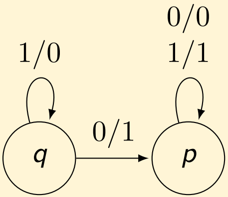
transducer

- without final or initial states.

An automaton may be:

| | | |
|---|---|---|
| deterministic | ✓ | ✓ |
| complete | (✓) | ✓ |
| invertible | – | ✓ |
| | $\mathscr{S}$-automaton "semigroup" | $\mathscr{G}$-automaton "group" |

## Example



# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

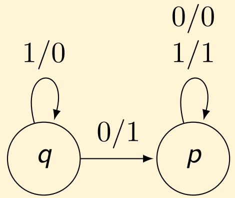
transducer

- without final or initial states.

An automaton may be:

| | | |
|---|---|---|
| deterministic | ✓ | ✓ |
| complete | (✓) | ✓ |
| invertible | – | ✓ |
| | $\mathscr{S}$-automaton "semigroup" | $\mathscr{G}$-automaton "group" |

**Example**




Every state $q$ induces a function $\Sigma^* \to \Sigma^*$

$$u \mapsto q \circ u$$

# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

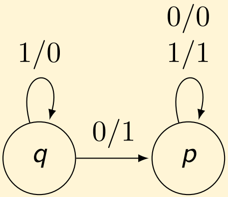
transducer

- without final or initial states.

An automaton may be:

| | | |
|---|---|---|
| • deterministic | ✓ | ✓ |
| • complete | (✓) | ✓ |
| • invertible | – | ✓ |
| | $\mathscr{S}$-automaton "semigroup" | $\mathscr{G}$-automaton "group" |

## Example




Every state $q$ induces a function $\Sigma^* \to \Sigma^*$

$$u \mapsto q \circ u$$

- $p$ induces the identity.

# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

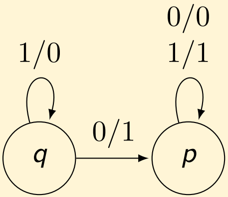
transducer

- without final or initial states.

An automaton may be:

| | | |
|---|---|---|
| deterministic | ✓ | ✓ |
| complete | (✓) | ✓ |
| invertible | – | ✓ |
| | $\mathscr{S}$-automaton "semigroup" | $\mathscr{G}$-automaton "group" |

**Example**




Every state $q$ induces a function $\Sigma^* \to \Sigma^*$

$u \mapsto q \circ u$

- $p$ induces the identity.
- $q \circ 000 = 100$

# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

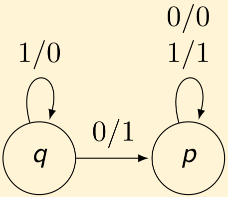
transducer

- without final or initial states.

An automaton may be:

| | | |
|---|---|---|
| deterministic | ✓ | ✓ |
| complete | (✓) | ✓ |
| invertible | – | ✓ |
| | $\mathscr{S}$-automaton "semigroup" | $\mathscr{G}$-automaton "group" |

**Example**




Every state $q$ induces a function $\Sigma^* \to \Sigma^*$

$u \mapsto q \circ u$

- $p$ induces the identity.
- $q \circ 000 = 100$

  $qq \circ 000 = q \circ 100 = 010$

# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

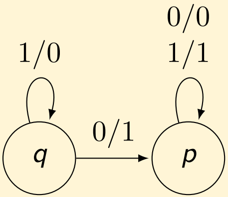
transducer

- without final or initial states.

An automaton may be:

| | $\mathscr{S}$-automaton "semigroup" | $\mathscr{G}$-automaton "group" |
|---|---|---|
| deterministic | ✓ | ✓ |
| complete | (✓) | ✓ |
| invertible | – | ✓ |

## Example




Every state $q$ induces a function $\Sigma^* \to \Sigma^*$

$u \mapsto q \circ u$

- $p$ induces the identity.
- $q \circ 000 = 100$

  $qq \circ 000 = q \circ 100 = 010$

  $qqq \circ 000 = \cdots = 110$

# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

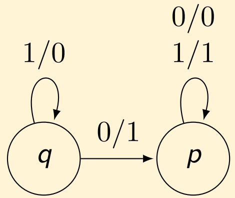
transducer

- without final or initial states.

An automaton may be:

| | | |
|---|---|---|
| • deterministic | ✓ | ✓ |
| • complete | (✓) | ✓ |
| • invertible | – | ✓ |
| | $\mathscr{S}$-automaton "semigroup" | $\mathscr{G}$-automaton "group" |

## Example




Every state $q$ induces a function $\Sigma^* \to \Sigma^*$

$u \mapsto q \circ u$

- $p$ induces the identity.
- $q \circ 000 = 100$

  $qq \circ 000 = q \circ 100 = 010$

  $qqq \circ 000 = \cdots = 110 \rightsquigarrow$ binary increment

# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

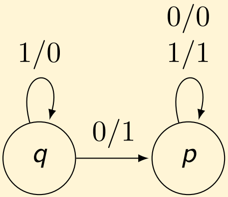
transducer

- without final or initial states.

An automaton may be:

| | $\mathscr{S}$-automaton "semigroup" | $\mathscr{G}$-automaton "group" |
|---|---|---|
| • deterministic | ✓ | ✓ |
| • complete | (✓) | ✓ |
| • invertible | – | ✓ |

## Example




"Adding Machine"

Every state $q$ induces a function $\Sigma^* \to \Sigma^*$

$u \mapsto q \circ u$

- $p$ induces the identity.
- $q \circ 000 = 100$

  $qq \circ 000 = q \circ 100 = 010$

  $qqq \circ 000 = \cdots = 110 \rightsquigarrow$ binary increment

# Automata

Here: An automaton $\mathcal{T} = (Q, \Sigma, \delta)$ is a

- finite-state,
- letter-to-letter

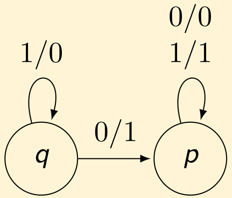
transducer

- without final or initial states.

An automaton may be:

| | | |
|---|---|---|
| deterministic | ✓ | ✓ |
| complete | (✓) | ✓ |
| invertible | – | ✓ |
| | $\mathscr{S}$-automaton "semigroup" | $\mathscr{G}$-automaton "group" |

## Example




"Adding Machine"

Every state $q$ induces a function $\Sigma^* \to \Sigma^*$

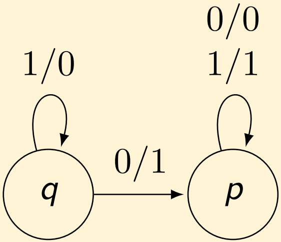
$u \mapsto q \circ u$

- $p$ induces the identity.
- $q \circ 000 = 100$
  $qq \circ 000 = q \circ 100 = 010$
  $qqq \circ 000 = \cdots = 110 \rightsquigarrow$ binary increment
- for $\mathscr{G}$-automata: bijections

# Automaton Monoid

- monoid $\mathscr{M}(\mathcal{T})$ generated by $\mathcal{T}$: closure under composition of the functions induced by the states plus identity "automaton monoid"

## Example

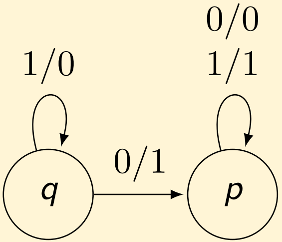



- $p$: identity
- $q$: increment

# Automaton Monoid

- monoid $\mathscr{M}(\mathcal{T})$ generated by $\mathcal{T}$: closure under composition of the functions induced by the states plus identity "automaton monoid"

## Example



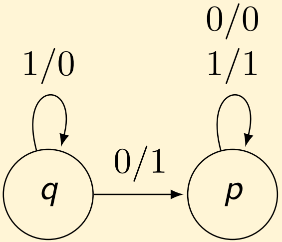

- $p$: identity
- $q$: increment

$\mathscr{M}(\mathcal{T}) \simeq \mathbb{N}_0$

# Automaton Monoid and Groups

- monoid $\mathscr{M}(\mathcal{T})$ generated by $\mathcal{T}$: closure under composition of the functions induced by the states plus identity "automaton monoid"
- group $\mathscr{G}(\mathcal{T})$ generated by $\mathcal{T}$: include inverse functions "automaton group"

## Example



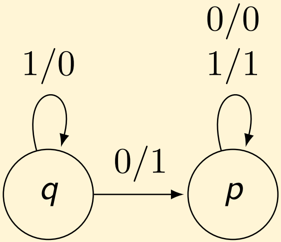

- $p$: identity
- $q$: increment

$$\mathscr{M}(\mathcal{T}) \simeq \mathbb{N}_0$$

# Automaton Monoid and Groups

- monoid $\mathscr{M}(\mathcal{T})$ generated by $\mathcal{T}$: closure under composition of the functions induced by the states plus identity "automaton monoid"
- group $\mathscr{G}(\mathcal{T})$ generated by $\mathcal{T}$: include inverse functions "automaton group"

## Example




- $p$: identity
- $q$: increment

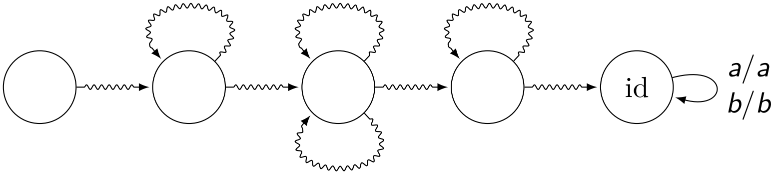
$\mathscr{M}(\mathcal{T}) \simeq \mathbb{N}_0$

$\mathscr{G}(\mathcal{T}) \simeq \mathbb{Z}$

# Sidki's Activity for Automata

# Sidki's Activity for Automata

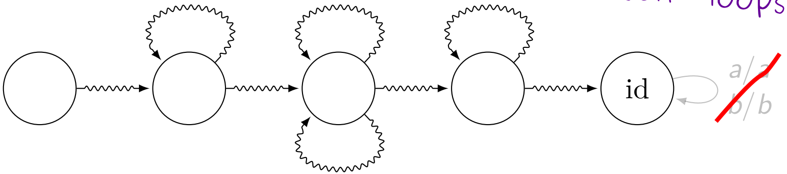
Consider a subautomaton/run ending in an identity state:



# Sidki's Activity for Automata

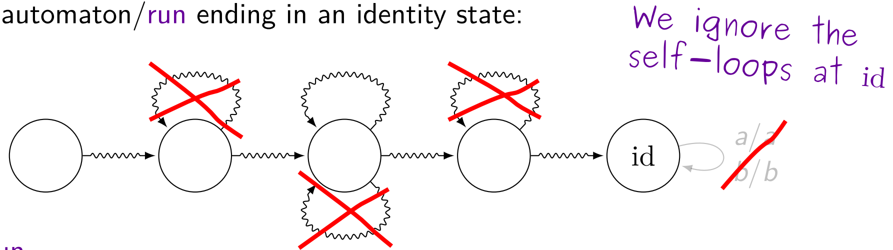
Consider a subautomaton/run ending in an identity state:

We ignore the self-loops at id



# Sidki's Activity for Automata

Consider a subautomaton/run ending in an identity state:



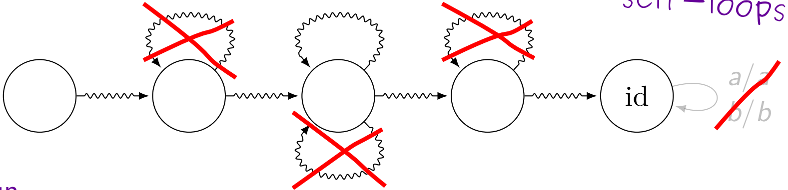

If every such run...

...has at most one cycle, the automaton has bounded activity.

# Sidki's Activity for Automata

Consider a subautomaton/run ending in an identity state:

We ignore the self-loops at id



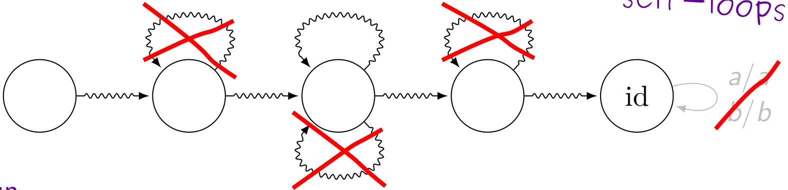

If every such run...

...has at most one cycle, the automaton has bounded activity. e.g. Grigorchuk's group

# Sidki's Activity for Automata

Consider a subautomaton/run ending in an identity state:

We ignore the self-loops at id




If every such run…

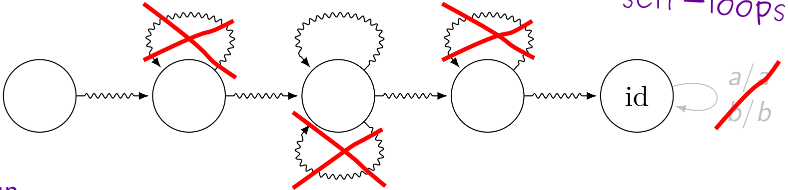
…has at most one cycle, the automaton has bounded activity. e.g. Grigorchuk's group

There is a generalization to monoids (Bartholdi, Godin, Klimann, Picantin; 2018)

# Sidki's Activity for Automata

Consider a subautomaton/run ending in an identity state:

We ignore the self-loops at id




If every such run...

...has at most one cycle, the automaton has bounded activity. e.g. Grigorchuk's group

There is a generalization to monoids (Bartholdi, Godin, Klimann, Picantin; 2018)

but without the above geometric characterization!

# Word Problem

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | | | | |

## Definition (Word Problem)

The word problem of an automaton groups is the problem

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** is $\boldsymbol{p} = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

# Word Problem

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | | | | |

## Definition (Word Problem)

The word problem of an automaton groups is the problem

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** is $\boldsymbol{p} = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

For monoids:
**Constant:** $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p}, \boldsymbol{q} \in Q^*$
**Question:** $\boldsymbol{p} = \boldsymbol{q}$ in $\mathscr{M}(\mathcal{T})$?

# Word Problem

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | | | PSPACE-complete D'A RW 2017 | |

## Definition (Word Problem)

The word problem of an automaton groups is the problem

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** is $\boldsymbol{p} = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

For monoids:
**Constant:** $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p}, \boldsymbol{q} \in Q^*$
**Question:** $\boldsymbol{p} = \boldsymbol{q}$ in $\mathscr{M}(\mathcal{T})$?

# Word Problem

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSpace-complete<br>W., Weiß; 2020 | | PSpace-complete<br>D'A RW 2017<br>by group case | |

## Definition (Word Problem)

The word problem of an automaton groups is the problem

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** is $\boldsymbol{p} = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

For monoids:
**Constant:** $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p}, \boldsymbol{q} \in Q^*$
**Question:** $\boldsymbol{p} = \boldsymbol{q}$ in $\mathscr{M}(\mathcal{T})$?

# Word Problem

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSPACE-complete W., Weiß; 2020 | $\in$ LOGSPACE | PSPACE-complete D'A RW 2017 by group case | |

## Definition (Word Problem)

The word problem of an automaton groups is the problem

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** is $\boldsymbol{p} = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

For monoids:
**Constant:** $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p}, \boldsymbol{q} \in Q^*$
**Question:** $\boldsymbol{p} = \boldsymbol{q}$ in $\mathscr{M}(\mathcal{T})$?

# Word Problem

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSPACE-complete<br>W., Weiß; 2020 | $\in$ LOGSPACE<br>$NC^1$-hard<br>follows f. Barrington; 1989 | PSPACE-complete<br>D'A RW 2017<br>by group case | |

## Definition (Word Problem)

The word problem of an automaton groups is the problem

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** is $\boldsymbol{p} = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

For monoids:
**Constant:** $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p}, \boldsymbol{q} \in Q^*$
**Question:** $\boldsymbol{p} = \boldsymbol{q}$ in $\mathscr{M}(\mathcal{T})$?

# Word Problem

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSPACE-complete<br>W., Weiß; 2020 | $\in$ LOGSPACE<br>$NC^1$-hard<br>follows f. Barrington; 1989 | PSPACE-complete<br>D'A RW 2017<br>by group case | *open* |

## Definition (Word Problem)

The word problem of an automaton groups is the problem

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** is $\boldsymbol{p} = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

For monoids:
**Constant:** $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p}, \boldsymbol{q} \in Q^*$
**Question:** $\boldsymbol{p} = \boldsymbol{q}$ in $\mathscr{M}(\mathcal{T})$?

# Word Problem

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSPACE-complete<br>W., Weiß; 2020 | ∈ LOGSPACE<br>$\mathrm{NC}^1$-hard<br>follows f. Barrington; 1989 | PSPACE-complete<br>D'A RW 2017<br>by group case | *open*<br>∈ PSPACE<br>$\mathrm{NC}^1$-hard |

## Definition (Word Problem)

The word problem of an automaton groups is the problem

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** is $\boldsymbol{p} = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

For monoids:
**Constant:** $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p}, \boldsymbol{q} \in Q^*$
**Question:** $\boldsymbol{p} = \boldsymbol{q}$ in $\mathscr{M}(\mathcal{T})$?

# Dehn's Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSPACE-complete<br>W., Weiß; 2020 | ∈ LOGSPACE<br>$NC^1$-hard<br>follows f. Barrington; 1989 | PSPACE-complete<br>D'A RW 2017<br>by group case | *open*<br>∈ PSPACE<br>$NC^1$-hard |
| conjugacy problem | | | | |

# Dehn's Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSpace-complete<br>W., Weiß; 2020 | ∈ LogSpace<br>$NC^1$-hard<br>follows f. Barrington; 1989 | PSpace-complete<br>D'A RW 2017<br>by group case | *open*<br>∈ PSpace<br>$NC^1$-hard |
| conjugacy problem | undecidable<br>Šunić, Ventura; 2012 | | | |

# Dehn's Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSpace-complete<br>W., Weiß; 2020 | ∈ LogSpace<br>$\mathrm{NC}^1$-hard<br>follows f. Barrington; 1989 | PSpace-complete<br>D'A RW 2017<br>by group case | *open*<br>∈ PSpace<br>$\mathrm{NC}^1$-hard |
| conjugacy problem | undecidable<br>Šunić, Ventura; 2012 | *open* | | |

# Dehn's Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSPACE-complete<br>W., Weiß; 2020 | ∈ LOGSPACE<br>$NC^1$-hard<br>follows f. Barrington; 1989 | PSPACE-complete<br>D'A RW 2017<br>by group case | *open*<br>∈ PSPACE<br>$NC^1$-hard |
| conjugacy problem | undecidable<br>Šunić, Ventura; 2012 | *open* | *n/a* | *n/a* |

# Dehn's Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| word problem | PSpace-complete<br>W., Weiß; 2020 | ∈ LogSpace<br>$\mathrm{NC}^1$-hard<br>follows f. Barrington; 1989 | PSpace-complete<br>D'A RW 2017<br>by group case | *open*<br>∈ PSpace<br>$\mathrm{NC}^1$-hard |
| conjugacy problem | undecidable<br>Šunić, Ventura; 2012 | *open* | *n/a* | *n/a* |
| isomorphism problem | undecidable<br>follows from ŠV | *open* | undecidable<br>by group case | *open* |

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order problem | | | | |

## Definition (Order Problem)

The order problem of an automaton group is:

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** $\exists n > 0 : \boldsymbol{p}^n = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | | | | |

## Definition (Order Problem)

The order problem of an automaton group is:

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** $\exists n > 0 : \boldsymbol{p}^n = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

## Definition (Torsion Problem)

The torsion problem of an aut. monoid is:

**Constant:** a $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^*$
**Question:** $\exists m, n : \boldsymbol{p}^{m+n} = \boldsymbol{p}^m$ in $\mathscr{M}(\mathcal{T})$?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | | | undecidable Gillibert; 2014 | |

## Definition (Order Problem)

The order problem of an automaton group is:

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** $\exists n > 0 : \boldsymbol{p}^n = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

## Definition (Torsion Problem)

The torsion problem of an aut. monoid is:

**Constant:** a $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^*$
**Question:** $\exists m, n : \boldsymbol{p}^{m+n} = \boldsymbol{p}^m$ in $\mathscr{M}(\mathcal{T})$?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | | undecidable<br>Gillibert; 2014 | |

## Definition (Order Problem)

The order problem of an automaton group is:

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** $\exists n > 0 : \boldsymbol{p}^n = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

## Definition (Torsion Problem)

The torsion problem of an aut. monoid is:

**Constant:** a $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{*}$
**Question:** $\exists m, n : \boldsymbol{p}^{m+n} = \boldsymbol{p}^m$ in $\mathscr{M}(\mathcal{T})$?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>B$^2$SZ; 2013 | undecidable<br>Gillibert; 2014 | |

Bondarenko$^2$, Sidki, Zapata

## Definition (Order Problem)

The order problem of an automaton group is:

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** $\exists n > 0 : \boldsymbol{p}^n = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

## Definition (Torsion Problem)

The torsion problem of an aut. monoid is:

**Constant:** a $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{*}$
**Question:** $\exists m, n : \boldsymbol{p}^{m+n} = \boldsymbol{p}^m$ in $\mathscr{M}(\mathcal{T})$?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2SZ$; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |

Bartholdi, Godin, Klimann, Picantin

## Definition (Order Problem)

The order problem of an automaton group is:

**Constant:** a $\mathscr{G}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{\pm *}$
**Question:** $\exists n > 0 : \boldsymbol{p}^n = \mathbb{1}$ in $\mathscr{G}(\mathcal{T})$?

## Definition (Torsion Problem)

The torsion problem of an aut. monoid is:

**Constant:** a $\mathscr{S}$-automaton $\mathcal{T} = (Q, \Sigma, \delta)$
**Input:** $\boldsymbol{p} \in Q^{*}$
**Question:** $\exists m, n : \boldsymbol{p}^{m+n} = \boldsymbol{p}^m$ in $\mathscr{M}(\mathcal{T})$?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2SZ$; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | | | | |

## Definition (Finiteness Problem)

The finiteness problem of an automaton group/monoid is the problem

**Input:** a $\mathscr{G}/\mathscr{S}$-automaton $\mathcal{T}$

**Question:** is $\mathscr{G}(\mathcal{T})/\mathscr{M}(\mathcal{T})$ finite?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2$SZ; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | | | undecidable<br>Gillibert; 2014 | |

## Definition (Finiteness Problem)

The finiteness problem of an automaton group/monoid is the problem

**Input:** a $\mathscr{G}/\mathscr{S}$-automaton $\mathcal{T}$

**Question:** is $\mathscr{G}(\mathcal{T})/\mathscr{M}(\mathcal{T})$ finite?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2SZ$; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | *open* | | undecidable<br>Gillibert; 2014 | |

## Definition (Finiteness Problem)

The finiteness problem of an automaton group/monoid is the problem

**Input:** a $\mathscr{G}/\mathscr{S}$-automaton $\mathcal{T}$

**Question:** is $\mathscr{G}(\mathcal{T})/\mathscr{M}(\mathcal{T})$ finite?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2SZ$; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | *open* | decidable<br>Bodarenko, W.; 2021 | undecidable<br>Gillibert; 2014 | |

## Definition (Finiteness Problem)

The finiteness problem of an automaton group/monoid is the problem

**Input:** a $\mathscr{G}/\mathscr{S}$-automaton $\mathcal{T}$

**Question:** is $\mathscr{G}(\mathcal{T})/\mathscr{M}(\mathcal{T})$ finite?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2$SZ; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | *open* | decidable<br>Bodarenko, W.; 2021 | undecidable<br>Gillibert; 2014 | decidable<br>D'A RW; arXiv 2024 |

## Definition (Finiteness Problem)

The finiteness problem of an automaton group/monoid is the problem

**Input:** a $\mathscr{G}/\mathscr{S}$-automaton $\mathcal{T}$

**Question:** is $\mathscr{G}(\mathcal{T})/\mathscr{M}(\mathcal{T})$ finite?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>B$^2$SZ; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | *open* | decidable<br>Bodarenko, W.; 2021 | undecidable<br>Gillibert; 2014 | decidable<br>D'A RW; arXiv 2024 |
| freeness problem | | | | |

## Definition (Freeness Problem)

The freeness problem of an automaton group/monoid is the problem

**Input:** a $\mathscr{G}/\mathscr{S}$-automaton $\mathcal{T}$

**Question:** is $\mathscr{G}(\mathcal{T})/\mathscr{M}(\mathcal{T})$ free?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2$SZ; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | *open* | decidable<br>Bodarenko, W.; 2021 | undecidable<br>Gillibert; 2014 | decidable<br>D'A RW; arXiv 2024 |
| freeness problem | | | undecidable<br>D'A RW; 2024 | |

## Definition (Freeness Problem)

The freeness problem of an automaton group/monoid is the problem

**Input:** a $\mathscr{G}/\mathscr{S}$-automaton $\mathcal{T}$
**Question:** is $\mathscr{G}(\mathcal{T})/\mathscr{M}(\mathcal{T})$ free?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2SZ$; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | *open* | decidable<br>Bodarenko, W.; 2021 | undecidable<br>Gillibert; 2014 | decidable<br>D'A RW; arXiv 2024 |
| freeness problem | *open* | | undecidable<br>D'A RW; 2024 | |

## Definition (Freeness Problem)

The freeness problem of an automaton group/monoid is the problem

**Input:** a $\mathscr{G}/\mathscr{S}$-automaton $\mathcal{T}$

**Question:** is $\mathscr{G}(\mathcal{T})/\mathscr{M}(\mathcal{T})$ free?

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2SZ$; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | *open* | decidable<br>Bodarenko, W.; 2021 | undecidable<br>Gillibert; 2014 | decidable<br>D'A RW; arXiv 2024 |
| freeness problem | *open* | decidable<br>almost trivial | undecidable<br>D'A RW; 2024 | |

## Theorem (Sidki; 2004)

*An automaton group of polynomial activity cannot contain a free subgroup of rank* $2$.

# Further Problems

| | general automaton groups | bounded automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2SZ$; 2013 | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | *open* | decidable<br>Bodarenko, W.; 2021 | undecidable<br>Gillibert; 2014 | decidable<br>D'A RW; arXiv 2024 |
| freeness problem | *open* | decidable<br>almost trivial | undecidable<br>D'A RW; 2024 | *open* |

**Theorem (Sidki; 2004)**

*An automaton group of polynomial activity cannot contain a free subgroup of rank $2$.*

Thank you!

# Summary

| | general automaton groups | bounded automaton groups | finitary automaton groups | general complete automaton monoids | bounded automaton monoids |
|---|---|---|---|---|---|
| word problem | PSPACE-complete<br>W., Weiß; 2020 | ∈ LOGSPACE<br>$NC^1$-hard<br>by finitary case | regular<br>$NC^1$-complete<br>Barrington; 1989 | PSPACE-complete<br>D'A RW 2017<br>by group case | *open*<br>∈ PSPACE<br>$NC^1$-hard |
| uniform word problem | PSPACE-complete<br>by non-unif. case | *open*<br>∈ PSPACE<br>coNP-hard | coNP-complete<br>Kotowsky, W.; 2023 | PSPACE-complete<br>by non-unif. case | *open*<br>∈ PSPACE<br>coNP-hard |
| conjugacy problem | undecidable<br>Šunić, Ventura; '12 | *open* | $NC^1$-complete<br>by word problem | *n/a* | *n/a* |
| isomorphism problem | undecidable<br>follows from ŠV | *open* | *trivial*<br>decidable<br>complexity *open* | undecidable<br>by group case | *open* |
| order/torsion problem | undecidable<br>Gillibert; 2018<br>Barth., Mitro.; 2020 | decidable<br>$B^2SZ$; 2013 | $NC^1$-complete<br>by word problem | undecidable<br>Gillibert; 2014 | decidable<br>BGKP; 2018 |
| finiteness problem | *open* | decidable<br>Bodarenko, W.; 2021 | *trivial*<br>decidable | undecidable<br>Gillibert; 2014 | decidable<br>D'A RW; arXiv 2024 |
| freeness problem | *open* | decidable<br>by order & finit. prob. | *trivial*<br>decidable | undecidable<br>D'A RW; 2024 | *open* |