# DECISION PROBLEMS FOR AUTOMATON SEMIGROUPS

EVAN BERGERON & KLAUS SUTNER

**Abstract**

The word problem is a classic group-theoretic decision problem. It's known to be undecidable in surprisingly small subclasses of groups. We consider a class of semigroups generated by finite automata for which this problem is decidable. We consider several related decision problems for this subclass of semigroups.

CONTENTS



## INTRODUCTION

## BACKGROUND

An *automaton* is a formally a triple $(Q, \Sigma, \delta)$, where $Q$ is some finite state set, $\Sigma$ is a finite alphabet of *symbols*, and $\delta$ is a transformation on $Q \times \Sigma$. Automata are typically viewed as directed graphs with vertex set $Q$ and an edge between $u, v$ if $(u, x)\delta = (v, y)$.

An automaton is said to be *synchronous* when $\delta$ outputs exactly one character for every transition and is *invertible* when every state in $Q$ has some bijection $\pi$ on $\Sigma$ such that $(u, x)\delta = (v, \pi(x))$. A state is a *copy state* if $\pi$ is the identity permutation and is a *toggle state* otherwise.

Each state $q \in Q$ acts on $\Sigma^*$, the set of finite strings over $\Sigma$. We commonly view $\Sigma^*$ as the infinite $|\Sigma|$-nary tree, so we can view $q$ as a transformation sending vertex $w$ to $wq$.

We can extend the action of $Q$ on $\Sigma^*$ to words $q = q_1 \cdots q_n$ over $Q^+$ by

$$wq = (\cdots((wq_1)q_2)\cdots q_n)$$

We adopt the convention of applying functions from the right here. In this way, function composition corresponds naturally with concatenation.

For an automaton $A$, we denote by $S(A)$ the semigroup generated by $Q$ under composition. $A$ is said to be *commutative* or *Abelian* when $S(A)$ is Abelian.

**Definition 1.** *A semigroup $S$ is called an automaton semigroup if there exists an automaton $A$ such that $S \simeq \Sigma(A)$.*

## PRIMARY RESULTS

### *Undecidablity results for submonoids of groups with decidable word problem*

We present a group with a decidable word problem with a submonoid for which IsGroup and IsFinite are undecidable.

**Definition 2.** *A Turing machine is a 7-tuple $(Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$, where $Q$, $\Sigma$, and $\Gamma$ are all finite sets and*

1. *$Q$ is the state set*
2. *$\Sigma$ is the input alphabet*
3. *$\Gamma$ is the tape alphabet, with $\Sigma \subseteq \Gamma$*
4. *$\delta : Q \times \Sigma \to Q \times \Gamma \times \{L, R\}$ is the transition function*
5. *$q_0$ is the start state*
6. *$q_{accept}$ and $q_{reject}$ are the accept and reject states, respectively.*

We can encapsulate the state of a Turing machine by its *configuration*. We typically write $uqv$, where $q$ is the current state, $u$ is the contents of the tape prior to the tapehead, and $v$ is the contents afterward. The tape heads sits on the first character of $v$.

We say configuration $C$ *yields* configuration $C'$ if the Turing machine can transition from $C$ to $C'$ in a single step.

For a Turing machine $T$, we define the group $G_T$ to be the Abelian group generated by all configurations of $T$ (and their imposed inverses), with the following identities

- $C_i = C_j$ if $C_i$ yields $C_j$.
- $uq_{accept}v = u'q_{reject}v' = 1$ for all $u, u', v, v'$.

It's clearly undecidable if a configuration $C$ is reachable from the start configuration. In order to ensure the solvability of $G_T$'s word problem, we modify the input TM to be *self-verifying*.

A *self-verifying Turing machine* $T$ maintains a program counter $p$ on the left end of the tape. At every step, it starts from the start configuration and runs for $p$ steps. If it arrives at its current state, it continues. Otherwise, it transitions to $q_{reject}$.

For every Turing machine $T$, we can construct an equivalent self-verifying TM $T'$. Full proof of this fact can be found in TODO.

**Proposition 1.** *For a self-verifying TM $T$, $G_T$ has a decidable word problem.*

*Proof.* Two strings $w_1$, $w_2$ are equal iff their lengths are the same and they have the same number of characters that lie on the canonical computation. □

We write $S = \langle q_0 \rangle$ for the submonoid of $G_T$ generated by $T$'s start state.

**Proposition 2.** *It is undecidable whether $S$ is a group.*

*Proof.* If $T$ halts, $S$ is the trivial group. Otherwise, $S$ is the commutative free monoid of rank 1. □

**Proposition 3.** *It is undecidable whether $S$ is finite.*

*Proof.* $S$ is finite iff $T$ halts. □

We follow a proof strategy similar to [5].

We define the *Knapsack Problem* as follows: given as input generators $g_1 \dots g_k$ and a target group element $g$, do there exist natural numbers $a_1 \dots a_k$ such that

$$g_1^{a_1} \cdots g_k^{a_K} = g$$

We prove that this problem is undecidable for automaton semigroups by reducing from Hilbert's tenth problem.

*It is decidable if an Abelian automaton semigroup generates a group*

Reduces to a system of equations. Abelian automaton semigroups can be written as a system of matrix equations: residuation is a linear operation here. We can then also write down the set of matrix equations for the inverse automaton.[1] Exactly what question do we then ask to verify there is a solution? Something about asking if the space spanned by the equations for $A$ has any intersection with $\mathbb{N}^n$.

This result is sign of hope: it's known that the IsGroup question is undecidable for finitely presented semigroups[3].

*Residuation Fixed Point is Decidable for Abelian automaton semigroups*

Take the matrix representing residuation for some word $w \in \Sigma^*$ and find if it has any eigenvectors in $\mathbb{N}^n$.

*Misc*

**Proposition 4.** *In an Abelian, minimal transducer $A$, every state has in-degree at most 2.*

*Proof.* Consider a state $s$. Every parent of $s$ is either a copy or a toggle state. If $s$ had two copy state parents, this contradicts minimality.

If $s$ had two toggle state parents $s_1$, $s_2$, then either $s = \partial_a s_1 = \partial_a s_2$ or $s = \partial_a s_1 = \partial_{\bar{a}} s_2$. Certainly, the first case contradicts minimality, since then

$$\partial_{\bar{a}} s_1 = \theta \partial_a s_1 = \theta \partial_a s_2 = \partial_{\bar{a}} s_2$$

and so $s_1 = s_2$. For the second case, then TODO I have another argument for this. □

## OPEN QUESTIONS

- Automorphism membership question
- IsGroup question for nonabelian automaton semigroups
- Isomorphism problem for automaton semigroups
- Residuation Fixed point problem
- All automaton semigroups are recursively presented. If these presentations are regular, or context-free, does that affect the soluability of these questions?
- Finiteness
- Having an identity

[1] Interesting to note that there's some duality here: if the semigroup of $A$ is a group, then so is the semigroup of $A^{-1}$ (and they are equal).

- Having a zero
- Bounded automata, etc

## REFERENCES


[1] L. Bartholdi and P. V. Silva. Groups defined by automata. *CoRR*, abs/1012.1531, 2010.

[2] A. J. Cain. Automaton semigroups. *TCS*, 410(47–49):5022–5038, 2009.

[3] A. J. Cain and V. Maltcev. Decision problems for finitely presented and one-relation semigroups and monoids. *International Journal of Algebra and Computation*, 19(6):747–770, 2009.

[4] O. Kharlampovich, B. Khoussainov, and A. Miasnikov. From automatic structures to automatic groups. *ArXiv e-prints*, July 2011.

[5] D. Konig, M. Lohrey, and G. Zetzsche. Knapsack and subset sum problems in nilpotent, polycyclic, and co-context-free groups. *Contemporary Mathematics*, 2015.

[6] Y. Muntyan. *Automata Groups*. PhD thesis, Texas A&M University, May 2009.

[7] V. Nekrashevych. *Self-Similar Groups*, volume 117 of *Math. Surveys and Monographs*. AMS, 2005.

[8] V. Nekrashevych and S. Sidki. *Automorphisms of the binary tree: state-closed subgroups and dynamics of 1/2-endomorphisms*. Cambridge University Press, 2004.

[9] K. Sutner and K. Lewi. Iterating inverse binary transducers. *Journal of Automata, Languages, and Combinators*, 8(2):18–24, 1976.