DECISION PROBLEMS IN INVERTIBLE AUTOMATA

EVAN BERGERON & KLAUS SUTNER

May 5, 2017

Abstract

We consider a variety of decision problems in groups and semigroups induced by invertible Mealy machines. Notably, we present proof that the automorphism membership problem in decidable in these semigroups. In addition, we prove undecidability of a Knapsack variant. A discussion of iteration and orbit rationality follows.

CONTENTS

BACKGROUND

An *automaton* is a formally a triple (Q, Σ, δ), where Q is some finite state set, Σ is a finite alphabet of *symbols*, and δ 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 δ outputs exactly one character for every transition and is called *invertible* when every state in Q has some bijection π on Σ such that $(u, x)\delta = (v, \pi(x))$. A state is a *copy state* if π is the identity permutation and is a *toggle state* otherwise.

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

We extend the action of Q on Σ^* 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. We write $G(A)$ for the group generated by the elements of Q and their inverses.

One may also speak about $S(A)$ and $G(A)$ without explicit reference to an automaton A. As such, we call a semigroup S an *automaton semigroup* if there is some automaton A with $S \simeq \Sigma(A)$. *Automaton groups* G are similarly defined.

Invertible automata have recently been usefully applied the group theory. A classic result here is Grigorchuk's group of intermediate growth, generated by the 5 state invertible machine shown in figure 1.

Decision Problems

Automaton semigroups exhibit many interesting and nuanced computability properties. While it is an easy result that the WORD PROBLEM is solvable in such semigroups, similar group-theoretic problems such as the CONJUGACY

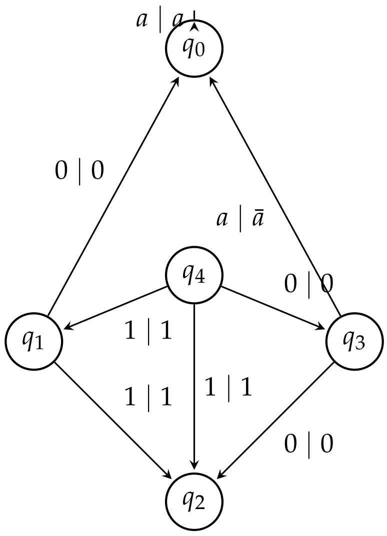

Figure 1: Grigorchuk's machine

problem and finiteness problem have been shown to be undecidable ([**?**], and [**?**], respectively).

Various other semigroup theoretic decision problems have recently been considered for small classes of semigroups by Cain in [**?**]. We consider a subset of his distinguished properties in the automaton semigroup case here.

decidable automorphism membership

Given an automaton A and an automorphism f, we decide if $f \in \mathcal{S}(A)$.

We're concerned only with self-similar automorphisms, as it makes little computational sense otherwise. (Regardless, all members of $\mathcal{S}(A)$ are self-similar). We assume we are given f in its wreath product form

$$f = (f_0, f_1)\sigma$$

where the set of transductions reachable by residuation is finite. Call the set of such transductions F and let $n = |F|$.

If h and h' are transductions, we write $h \equiv h'$ if h and h' are equivalent when restricted to $\mathbf{2}^n$.

Claim: Equivalent transductions on $\mathbf{2}^*$ form a regular language

Obviously, $G = \{g \mid g : \mathbf{2}^n \to \mathbf{2}^n\}$ is finite. Define a DFA with state set G, start state `id` and transitions

$$g_i \xrightarrow{a} g_i a$$

for $a \in F$. As usual, we have function application on the right (that is, $xg_i a = a(g_i(x))$).

Then for $f \in F$, we have that the following language is regular:

$$L_f := \{l \mid l \equiv f\}.$$

we can represent this as a DFA. Further, residuation is rational, so compose those two DFAs. That's how we residuate.

Corollary 1. IsGroup *is decidable for invertible automaton groups: simply check membership of q^{-1} for $q \in Q$.*

Corollary 2. IsMonoid *is decidable.*

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

1. *Q is the state set*
2. *Σ is the input alphabet*
3. *Γ 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. □

knapsack is undecidable for automaton semigroups

We follow a proof strategy similar to [**?**].

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.

OPEN QUESTIONS

- All automaton semigroups are recursively presented. If these presentations are regular, or context-free, does that affect the soluability of these questions?
- Having a zero
- Isomorphism problem
- Bounded automata, etc