The Cayley-Graph of the Queue Monoid: Logic and Decidability

Faried Abu Zaid
Technische Universität Ilmenau, Automata and Logics Group
faried.abu-zaid@tu-ilmenau.de

Chris Köcher
Technische Universität Ilmenau, Automata and Logics Group
chris.koecher@tu-ilmenau.de
https://orcid.org/0000-0003-4575-9339

Abstract

We investigate the decidability of logical aspects of graphs that arise as Cayley-graphs of the so-called queue monoids. These monoids model the behavior of the classical (reliable) fifo-queues. We answer a question raised by Huschenbett, Kuske, and Zetzsche and prove the decidability of the first-order theory of these graphs with the help of an - at least for the authors - new combination of the well-known method from Ferrante and Rackoff and an automata-based approach. On the other hand, we prove that the monadic second-order of the queue monoid's Cayley-graph is undecidable.

2012 ACM Subject Classification Theory of computation → Logic, Theory of computation → Models of computation, Information systems → Data structures

Keywords and phrases Queues, Transformation Monoid, Cayley-Graph, Logic, First-Order Theory, MSO Theory, Model Checking

Digital Object Identifier 10.4230/LIPIcs...

1 Introduction

Data structures are one of the most important concepts in nearly all areas of computer science. Important data structures are, e.g., finite memories, counters, and (theoretically) infinite Turing-tapes. But the most fundamental ones are stacks and queues. And although these two data structures look very similar as they have got the same set of operations on them (i.e. writing and reading of a letter), they differ from the computability's point of view: if we equip finite automata with both data structures, then the ones with stacks compute exactly the context-free languages (these are the well-known pushdown automata). But if we equip an finite automaton with queues (in literature they are called queue automata, communicating automata, or channel systems) then we obtain a Turing-complete computation model (cf. [2,3]). This strong model can be weakened with various extensions, e.g., if the queue is allowed to forget some of its contents (cf. [1,5,20]) or if letters of low priority can be superseded by letters with higher priority (cf. [10]).

One possible approach to analyze the difference of the behavior of the data structures is to model them as a monoid of transformations. Then, finite memories induce finite monoids, counters induce the integers with addition, stacks induce the polycyclic monoids (cf. [12,25]), and queues induce the so-called queue monoids which were first introduced in [11]. And while the transformation monoids of the other data structures are very well-understood, we still do not know much about the queue monoid. Further results on the queue monoid (with and without lossiness) can be found in [15, 16]. Here, we only consider the reliable queue monoids. Concretely, we study the Cayley-graph of this monoid.

Cayley-graphs are a natural translation of finitely generated groups and monoids into graph theory and is a fundamental tool to handle these algebraic constructs in combinatorics, topology, and automata theory. Concretely, these are labeled, directed graphs with labels from a fixed generating set Γ of the monoid $\mathcal{M}$. Thereby, the elements from $\mathcal{M}$ are the graph's nodes and there is an a-labeled edge (where $a \in \Gamma$) from $x \in \mathcal{M}$ to $y \in \mathcal{M}$ iff $xa = y$ holds in $\mathcal{M}$. For groups, we already know many results on their Cayley-graphs. For example, the group's Cayley-graph has decidable first-order theory if, and only if, its existential first-order theory is decidable and if, and only if, the group's word problem is decidable [17]. Moreover, a group's Cayley-graph has decidable monadic second-order theory if, and only if, the group is context-free (that is, if the group's word problem is context-free) [17, 21]. Besides these results, Kharlampovich et al. considered in [13] so-called Cayley-graph automatic groups (these are the groups having an automatic Cayley-graph in the sense of [14]) which links to the rich theory of automatic structures.

Unfortunately, there are not that many studies on Cayley-graphs of monoids. In particular, there are monoids with decidable word problem but undecidable existential first-order theory of their Cayley-graph [18, 22]. For finite monoids the Cayley-graphs are finite and, hence, the first- and second-order theories are decidable in polynomial space [8]. For polycyclic monoids the Cayley-graphs are automatic, complete $|A|$-ary trees (where A is the underlying alphabet) with an additional node every other node is connected with (this is the zero element resp. error state). Therefore, due to [6, 18] the Cayley-graphs monadic second-order theory is decidable (the first-order theory is even in 2EXPSPACE by [19]).

In this paper we want to consider logics on the Cayley-graph of the queue monoid. Concretely, we will see that this graph's first-order theory is decidable by giving an primitive recursive (but non-elementary) algorithm which combines two well-known methods from model theory in a (at least for the authors) new way: the method of Ferrante and Rackoff [7] and an automata-based approach. This gives an answer on a question raised by Huschenbett, Kuske, and Zetzsche [11]. There, they conjectured the undecidability of its first-order logic implying that the graph is not automatic in the sense of [14]. Moreover, we will prove the undecidability of the monadic second-order theory with the help of a well-known result from Seese [26].

2 Preliminaries

For $m, n, r \in \mathbb{N}$ we write $m =_r n$ iff $m = n$ or $m, n > r$.

2.1 Graphs

Let $\mathfrak{G} = (V, E)$ be a graph (possibly with some constants $c_1, \ldots, c_n \in V$). For $u, v \in V$ we denote by $d^{\mathfrak{G}}(u, v)$ the length of a shortest path from u to v, where we set $d(u, v) = \infty$ whenever u and v are not connected in $\mathfrak{G}$. For $S \subseteq V$ we denote by $\mathfrak{G}_{\restriction S}$ the induced subgraph of $\mathfrak{G}$ with vertex set S. For a nonempty set $S \subseteq V$ and $r \in \mathbb{N}$ let $\mathcal{N}_r^{\mathfrak{G}}(S)$ denote the *r-neighborhood* of S (and the constants $c_1, \ldots, c_n$) in $\mathfrak{G}$, that is $\mathfrak{G}_{\restriction\{u \in V \mid \min\{d(u,v) \mid v \in S \cup \{c_1, \ldots, c_n\}\} \leqslant r\}}$. For a tuple $\vec{u} = (u_1, \ldots, u_k) \in V^k$ we will also write $\mathcal{N}_r^{\mathfrak{G}}(\vec{u})$ instead of $\mathcal{N}_r^{\mathfrak{G}}(\{u_1, \ldots, u_k\})$.

2.2 Combinatorics on Words

Let A be an alphabet. We use $\preceq$ to denote the *prefix-relation* and $\sqsubseteq$ for the *suffix-relation* on A^*. If $u = vw$ we write $v^{-1}u = w$ and $uw^{-1} = v$. Let $\mathrm{pref}_r(u)$ denote the maximal prefix of u of length at most r. For $u, v \in A^*$ let $u \sqcap v$ denote the largest suffix of u that is also a prefix of v.

In a first lemma we prove that the complementary prefix and suffix of u resp. v wrt. $u \sqcap v$ can be shortened to words of length at most $2r$ having the same prefixes and suffixes:

▸ **Lemma 2.1.** *Let $r \in \mathbb{N}$ and $u, v, w \in A^*$ with $uw \sqcap wv = w$. Then there are words u', v' of length $\mathcal{O}(r)$ such that*

- $\mathrm{suf}_r(uw) = \mathrm{suf}_r(u'w)$,
- $\mathrm{suf}_r(wv) = \mathrm{suf}_r(wv')$,
- $\mathrm{pref}_r(wv) = \mathrm{pref}_r(wv')$,
- $u'w \sqcap wv' = w$.

Proof. Set $u' = \mathrm{suf}_r(u)$. Additionally, if $|v| \leqslant 2r$ set $v' := v$, and otherwise, set $v' := \mathrm{pref}_r(v)\,\mathrm{suf}_r(v)$. Then the first three equations are obviously satisfied. Now assume $u'w \sqcap wv' \neq w$, i.e., there is $w' \in A^*$ with $|w'| > |w|$, $w' \preceq wv'$, and $w' \sqsubseteq u'w$. Since $|u'w| \leqslant r + |w|$ we have $w' \preceq w\,\mathrm{pref}_r(v) \preceq wv$. Additionally, we have $w' \sqsubseteq u'w \sqsubseteq uw$ implying $|uw \sqcap wv| \geqslant |w'| > |w|$. This is a contradiction to the definition of w. ◂

A *period* of a word u is a word v such that $u \preceq v^\omega$. Obviously every word u has a unique smallest period, which we denote by $\sqrt{u}$. The *left-exponent* of u in v is the largest number n such that $v = u^n w$, and it is denoted by $\mathrm{lexp}(u, v)$. The *right-remainder*, $v \mod u$, of v with respect to u is defined as $(u^{\mathrm{lexp}(u,v)})^{-1}v$, that is the unique w such that $v = u^{\mathrm{lexp}(u,v)}w$. In particular we have $v = \sqrt{v}^{\mathrm{lexp}(\sqrt{v},v)}(v \mod \sqrt{v})$ for every $v \in A^*$. A word u is *primitive* if there is no v with $|v| < |u|$ and $u = v^n$ for some $n \in \mathbb{N}$. For $u, v \in A^*$ let $u\Delta v = (y, z)$, where y, z are minimal such that there exists an x with $u = xy$ and $v = xz$. For $\vec{v}, \vec{w} \in (A^*)^k$ let $\vec{v}\Delta\vec{w} = (v_1\Delta w_1, \ldots, v_k\Delta w_k) \in (A^*)^{2k}$ and $|\vec{w}| := \sum_{i=1}^{k} |w_i|$.

▸ **Definition 2.2.** Let $u \in A^*$ be a word. A word $v \in A^*$ is a *border* of u (denoted by $v \underset{\sqsubsetneq}{\precneq} u$) if $v \precneq u$ and $v \sqsubsetneq u$. A *border-decomposition* of u is a sequence of words $\varepsilon = u_0, u_1, \ldots, u_n = u$ such that for all $0 \leqslant i < n$ it holds that $u_i \underset{\sqsubsetneq}{\precneq} u_{i+1}$. A border-decomposition $u_0, u_1, \ldots, u_n$ is *complete* if there is no $1 \leqslant i < n$ and $v \in A^*$ with $u_i \underset{\sqsubsetneq}{\precneq} v \underset{\sqsubsetneq}{\precneq} u_{i+1}$.

Hence, a complete border-decomposition of $u \in A^*$ is the sequence of all borders of u ordered by word length. So, it is easy to observe that each word $u \in A^*$ has exactly one complete border-decomposition.

▸ **Example 2.3.** The complete border-decomposition of *ababa* is $(\varepsilon, a, aba, ababa)$.

From the complete border-decomposition of a word w we derive the so called skeleton of w containing the inner words v of all bordered words uvu in w.

▸ **Definition 2.4.** Let $w \in A^*$ and $\vec{w} = (w_0, \ldots, w_n)$ be the complete border-decomposition of w. The *r-skeleton* of w, denoted by $\mathcal{S}_r(w)$, is the word of length n over the alphabet $\Gamma = A^{\leqslant r}$ with $\mathcal{S}_r(w)[i] = \mathrm{pref}_r(w_i^{-1}w)$ for each $0 \leqslant i \leqslant n-1$. Note that $w_i^{-1}w$ is always defined since $w_i \preceq w$.

▸ **Example 2.5.** Let $u = bababa$ and $v = ababab$. Then $u \sqcap v = ababa$ and the complete border-decomposition of $u \sqcap v$ is $(\varepsilon, a, aba, ababa)$. The 2-skeleton of $u \sqcap v$ is the word depicted below.

$$ab \longrightarrow ba \longrightarrow ba$$

By definition the i-th element of the border-decomposition of $u \sqcap v$ corresponds to the i-th word in the skeleton (which we understand as letter). We will use this correspondence to translate back and forth between an Ehrenfeucht-Fraïssé game played on the Cayley-graph of a queue monoid and games played on certain skeletons which are derived from the game played on the Cayley-graph.

▸ **Lemma 2.6.** *Let $r \in \mathbb{N}$, $w \in A^*$ and $n \in \mathbb{N}$ be the length of $\mathcal{S}_r(w)$. Then a word $v \in A^*$ can be constructed from w such that $|v| = \mathcal{O}(2^{nr})$and $\mathcal{S}_r(w) = \mathcal{S}_r(v)$.*

$|v| = \mathcal{O}(r2^n)$?

Proof. Let $\vec{w} = (w_0, \dots, w_n)$ be the complete border-decomposition of w. At first, assume $|\mathcal{S}_r(w)[n-1]| < r$ (i.e., the last component is small). Then there are two possibilities: on the one hand $w = w_{n-1}xw_{n-1}$ and $|xw_{n-1}| < r$. In this case we have $|w| < 2r = \mathcal{O}(2^{nr})$. On the other hand we have $w = xw_{n-1} = w_{n-1}y$ where $|x| = |y| < \min\{|w_{n-1}|, r\}$, i.e., the prefix and the suffix w_{n-1} overlap in w_n. Then it is easy to see that x is a period of w_{n-1} and of w_n. Concretely, there is a prefix p of x and a number $k \in \mathbb{N}$ such that $w = x^k p$ and $w_{n-1} = x^{k-1}p$. In particular, all word $x^i p$ with $1 \leqslant i \leqslant k$ are borders of w which implies $k \leqslant n$. Hence we have $|w| \leqslant |x| \cdot (k+1) \leqslant r \cdot (n+1) = \mathcal{O}(2^{nr})$. Therefore, in both cases we are ready and we can assume $|\mathcal{S}_r(w)[n-1]|$ from now on.

We construct v inductively as follows: We set $v_0 := \varepsilon$. Now let $a, b \in A$ be distinct with $\mathcal{S}_r(w)[0] \in aA^*$. Then $x \lneqq \mathcal{S}_r(w)[0]b^{2n+r}$ implies $x = \varepsilon$. Hence, we set, for $0 \leqslant i < n$, $v_{i+1} := v_i x_i v_i$ where $x_i = \mathcal{S}_r(w)[i]\, b^{n-i} a^i b^{n+r}$. Finally, we set $v := v_n$.

Before we can prove $\mathcal{S}_r(w) = \mathcal{S}_r(v)$ we need to prove the following two properties of $(v_0, \dots, v_n)$:

(a) For each $0 \leqslant i \leqslant n$ $\sqrt{v_{i+1}} = v_i x_i$ and

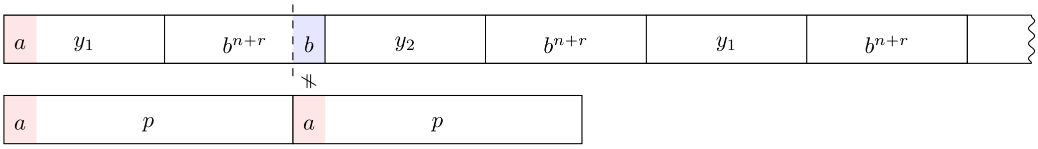
(b) $\vec{v} = (v_0, \dots, v_n)$ is a complete border-decomposition of v.

Proof of (a). We observe that $v_i x_i$ is a period of v_{i+1} and we prove by induction on $0 \leqslant i \leqslant n$ that this period is minimal. For $i = 0$ this is trivial since $v_1 \in aA^{r-1}b^{2n+r}$ and $a \neq b$. So now let $i > 0$. We suppose that there is a period p of v_{i+1} with $|p| < |v_i x_i|$. Then, for $y_j := x_j(b^{n+r})^{-1}$ for $0 \leqslant j \leqslant i$, the word v_{i+1} is an alternation of words y_j and b^{r+n} which are all of length $r+n$. Note that by construction we have $y_j \neq b^{n+r}$ (since each y_j contains at least one a) as well as $y_j \neq y_k$ if $j \neq k$ for each $0 \leqslant j, k \leqslant i$. Additionally, each second occurrence of a y_j-block is y_1. We now consider two cases:

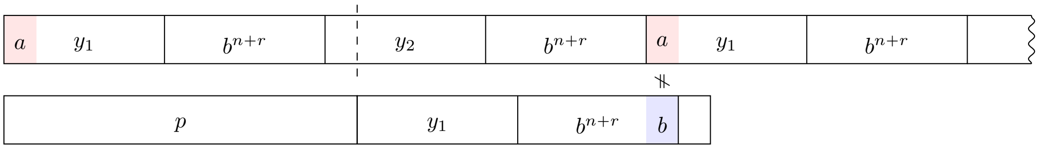
First, assume that $|p|$ is not a divisor of $n+r$. If $|p| < n+r$ then the distance between each two occurrences of a in p^ω is at most $|p| < n+r$ but v_{i+1} contains at least one b^{n+r}-block. Hence, we have $|p| > n+r$. If $\lfloor \frac{|p|}{n+r} \rfloor$ is odd (cf. Fig. 1a), p starts with a and ends in a block of the form b^{n+r}, but does not contain all of these $n+r$ many b's. Since p start with an a, a first repetition of p this first a is different from the b at this position in v_{i+1}, i.e., p is not a period of v_{i+1}. Otherwise, if $\lfloor \frac{|p|}{n+r} \rfloor$ is even (cf. Fig. 1b), then the prefix of $p^{-1}v_{i+1}$ of length $|p|$ contains at most one y_1-block and this overlaps with a b^{n+r}-block. Hence, there is a position in the first repetition of p containing a b which is different from the a at this position in v_{i+1}.

Now, assume $|p|$ is a divisor of $n+r$. Then we can understand the blocks of length $n+r$ as letters of the alphabet $\{b^{n+r}, y_1, \dots, y_i\}$. Since there is no y_i-block in v_i we have $|p| \geqslant |v_i y_i|$. Since p starts with y_1 and y_i is followed by b^{n+r}, p has length at least $|v_i x_i|$.

Proof of (b). By construction, it is easy to see that $\vec{v} = (v_0, \dots, v_n)$ is a border-decomposition of $v = v_n$. We prove now by induction on $0 \leqslant i < n$ that $(v_0, \dots, v_{i+1})$ is a complete border-decomposition of v_i. The case $i = 0$ is easy to verify since $v_1 \in aA^{r-1}b^{2n+r}$.



(a) Case $\left\lfloor \frac{|p|}{n+r} \right\rfloor$ is odd



(b) Case $\left\lfloor \frac{|p|}{n+r} \right\rfloor$ is even

Figure 1

So, let $i \geqslant 1$. Assume there is $u \in A^*$ with $v_i \precneqq u \precneqq v_{i+1}$. Let u be of minimal length satisfying this inequality. Then there are two possible cases:

First, suppose $|u| \geqslant |v_i x_i|$ holds, i.e., the prefix and suffix u overlap in v_i and the overlap contains at most x_i (cf. Fig. 2). Let $x, y \in A^*$ such that $u = x x_i v_i = y$. Then we have $|x| = |y|$ and $m := x x_i y \precneqq u$. Hence, by minimality of u we have $|m| \leqslant |v_i|$ and therefore, by induction hypothesis, $m = v_k$ for some $0 < k \leqslant i$. This implies

$$v_{k-1} x_{k-1} v_{k-1} = v_k = m = x x_i y \,.$$

Since $|x| = |y|$ and $|x_i| = |x_{k-1}|$ we have $x_i = x_{k-1}$, which is a contradiction to the construction of the x_i's.

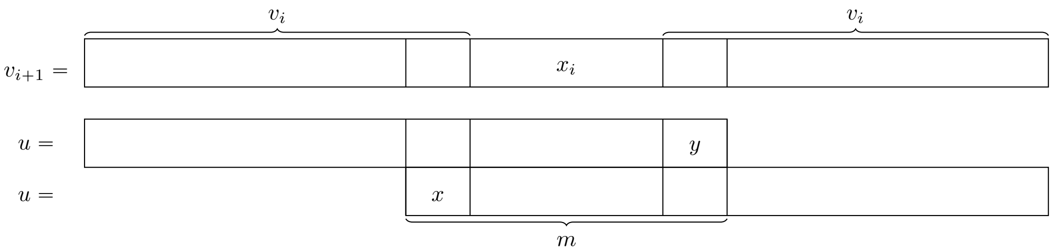


Figure 2

Now, suppose $|u| < |v_i x_i|$. If $|u| \geqslant \frac{|v_{i+1}|}{2}$ (i.e., the prefix and suffix u in v_i overlap) then there is a word $m \in A^*$ such that $m \precneqq u$ holds. Hence, by minimality of u and by induction hypothesis we have $m = v_k$ for some $0 \leqslant k \leqslant i$. Since $|m| < |x_i| = |x_1|$ we have $m = \varepsilon$, i.e., we have $|u| = \frac{|v_{i+1}|}{2}$.

Suppose $|u| \leqslant \frac{|v_{i+1}|}{2}$ (i.e., the prefix and suffix u in v_i do not overlap). Then there is a word $p \in A^*$ such that $v_{i+1} = pu$. Since u is a prefix of v_{i+1} and $|p| > \frac{|v_{i+1}|}{2}$, u also is a prefix of p. Hence, p is a period of v_{i+1} and we have

$$|p| = |v_{i+1}| - |u| < |v_{i+1}| - |v_i| = |v_i x_i| \,.$$

This is a contradiction to property a stating that $v_i x_i$ is the minimal period of v_{i+1}.

So, in both cases we have seen that there is no $v_i \lneqq u \lneqq v_{i+1}$, i.e., $(v_0, \ldots, v_{i+1})$ is a complete border-decomposition.

Finally, let $0 \leqslant i < n$. Then we have

$$\mathcal{S}_r(v)[i] = \mathrm{pref}_r(v_i^{-1}v) = \mathrm{pref}_r(\mathcal{S}_r(w)[i]\, s) = \mathcal{S}_r(w)[i]$$

for some $s \in A^*$, i.e., $\mathcal{S}_r(v) = \mathcal{S}_r(w)$. Additionally, we have $|v_i| = 2|v_{i-1} + 2n + 2r$ for $1 \leqslant i \leqslant n$ and $|v_0| = 0$ which results in $|v| = |v_n| = (2^n - 1)(2n + 2r) = \mathcal{O}(2^{nr})$. ◂

Let $V \in (A^{\leqslant r})^*$ be the r-skeleton of some word $w \in A^*$. We call the word $v \in A^*$ constructed in the proof of Lemma 2.6 the *r-instantiation of V*.

3 Queue Monoid and its Cayley-Graph

3.1 Definition of the Monoid

The queue monoid models the behavior of a (reliable) fifo-queue whose entries come from an alphabet A. Consequently, the state of a queue is a word from A^*. The basic actions of our queue are writing of the symbol $a \in A$ of the queue (denoted by a) and reading the symbol $a \in A$ from the queue (denoted by $\overline{a}$). Thereby, $\overline{A}$ is a disjoint copy of A containing all reading actions $\overline{a}$ and $\Sigma := A \cup \overline{A}$ is the set of all basic actions. To simplify notation, for a word $u = a_1 a_2 \ldots a_n \in A^*$ we write $\overline{u}$ for the word $\overline{a_1}\,\overline{a_2} \ldots \overline{a_n}$.

Formally, the action $a \in A$ appends the letter a to the state of the queue and the action $\overline{a} \in \overline{A}$ tries to cancel the letter a from the beginning of the current state of the queue. Thereby, if the state does not start with this symbol, the queue will end up in an error state which we denote by $\perp$. Note that in contrast to (partially) lossy queues which we considered in [15, 16], these queues cannot forget any part of their content. Hence, these ideas lead to the following definition:

▸ **Definition 3.1.** Let $\perp \notin A^*$. The function $\circ\colon (A^* \cup \{\perp\}) \times \Sigma^* \to (A^* \cup \{\perp\})$ is defined for each $s \in A^*$, $a, b \in A$, and $u \in \Sigma^*$ as follows:

(1) $s \circ \varepsilon = s$

(2) $s \circ au = sa \circ u$

(3) $bs \circ \overline{a}u = \begin{cases} s \circ u & \text{if } a = b \\ \perp & \text{otherwise} \end{cases}$

(4) $\varepsilon \circ \overline{a}u = \perp \circ u = \perp$

With the help of this function we may now identify sequences of actions that are acting equally. This is finally used to define the monoid of queue actions.

▸ **Definition 3.2.** Let $u, v \in \Sigma^*$. Then u and v *act equally* (denoted by $u \equiv v$) if $s \circ u = s \circ v$ holds for each $s \in A^*$.

Since $s \circ uv = (s \circ u) \circ v$, the resulting relation $\equiv$ is a congruence on the free monoid Σ. Hence, the quotient $\mathcal{Q}(A) := \Sigma^*/_{\equiv}$ is a monoid which we call the *monoid of queue actions* or for short *queue monoid*.

Note that the queue monoids $\mathcal{Q}(A)$ for alphabets A of different size are not isomorphic. Though, all of the following results hold for any alphabet A with $|A| \geqslant 2$. Hence, we may fix an arbitrary alphabet A from now on and write $\mathcal{Q}$ instead of $\mathcal{Q}(A)$.

▸ Remark. Let $A = \{a\}$ be a singleton. Then a queue on this alphabet acts like a partially blind counter since $a^n \circ a = a^{n+1}$ and $a^{n+1} \circ \overline{a} = a^n$. In other words, $\mathcal{Q}(\{a\})$ is the bicyclic semigroup.

3.2 Basic Properties

Now, we want to recall some basic properties considering the equivalence relation $\equiv$. The first important fact expresses the equivalence in terms of some commutations of write and read actions under certain contexts.

▸ **Theorem 3.3** ([11, Theorem 4.3]). *The equivalence relation $\equiv$ is the least congruence on the free monoid Σ^* satisfying the following equations for all $a, b \in A$:*

(1) $a\overline{b} \equiv \overline{b}a$ *if* $a \neq b$

(2) $a\overline{a}\overline{b} \equiv \overline{a}a\overline{b}$

(3) $ba\overline{a} \equiv b\overline{a}a$ ◂

A very frequently used notation is the following: the *projections to write and read actions*, resp., are defined as $\mathsf{wrt}, \mathsf{rd}\colon \Sigma^* \to A^*$ by $\mathsf{wrt}(a) = \mathsf{rd}(\overline{a}) = a$ and $\mathsf{wrt}(\overline{a}) = \mathsf{rd}(a) = \varepsilon$ for all $a \in A$. In other words, $\mathsf{wrt}(u)$ can be derived from u by deletion of all read actions and $\mathsf{rd}(u)$ can be obtained from u by deletion of all the write actions and by suppression of the overlines. Due to Theorem 3.3 all words contained in a single equivalence class of $\equiv$ have the same projections. Hence we use them for equivalence classes as well. Though, equality of these projections of two words does not imply equivalence of these words. For example, $u = \overline{a}a$ and $v = a\overline{a}$ have the same projections $\mathsf{wrt}(u) = \mathsf{rd}(u) = a = \mathsf{wrt}(v) = \mathsf{rd}(v)$ but are not equivalent according to Theorem 3.3.

The non-equivalence of the two words above is very easy to prove. Also (non-)equivalence of two arbitrary words is decidable in polynomial time: for this purpose we compute normal forms of the equivalence classes of $\equiv$. We do this by ordering the equations from Theorem 3.3 from left to right resulting in a terminating and confluent semi-Thue system $\mathcal{R}$ [11, Lemma 4.1]. Then, for any word $u \in \Sigma^*$ there is a unique, irreducible word $\mathsf{nf}(u)$ with $u \to^* \mathsf{nf}(u)$, the so-called *normal form* of u resp. of its equivalence class $[u]$. In this word $\mathsf{nf}(u)$ the read actions from u are moved to the left as far as the equations from above allow.

▸ **Example 3.4.** Let $a, b \in A$ with $a \neq b$ and $u = abb\overline{a}\overline{b}$. Then we have

$$abb\overline{a}\overline{b} \xrightarrow{(1)} ab\overline{a}b\overline{b} \xrightarrow{(1)} a\overline{a}bb\overline{b} \xrightarrow{(3)} a\overline{a}b\overline{b}b\,.$$

Since we cannot apply any rule from Theorem 3.3 anymore, we have $\mathsf{nf}(u) = a\overline{a}b\overline{b}b$.

From the definition of $\mathcal{R}$ we obtain that a word is in normal form if it starts with a sequence of read operations followed by an alternating sequence of write and read actions, where all of the read actions $\overline{a}$ appear straight behind the write action a. Finally, the normal form ends with a sequence of write actions. Concretely, the set of all normal forms is

$$\mathsf{NF} := \{\mathsf{nf}(u) \mid u \in \Sigma^*\} = \overline{A}^*\{a\overline{a} \mid a \in A\}^* A^*\,.$$

Let $u \in \Sigma^*$. Then the normal form $\mathsf{nf}(u)$ is uniquely defined by three words $u_1, u_2, u_3 \in A^*$ such that $\mathsf{nf}(u) = \overline{u_1}a_1\overline{a_1}\ldots a_n\overline{a_n}u_3$ where $u_2 = a_1 \ldots a_n$. Thereby, we denote the word u_1 by $\lambda(u)$, the word u_2 by $\mu(u)$, and u_3 by $\varrho(u)$. Hence, we can define the *characteristics* of u ($[u]$, resp.) by the triple

$$\chi(u) := (\lambda(u), \mu(u), \varrho(u))\,.$$

Hence, from these characteristics $\chi(u)$ we can obtain the projections of u on its write and read actions as well: $\mathsf{wrt}(u) = \mu(u)\varrho(u)$ and $\mathsf{rd}(u) = \lambda(u)\mu(u)$.

From now on, we will use these characteristics to represent the elements of $\mathcal{Q}$. In other words, we may understand $\mathcal{Q}$ as a triple of words (i.e., $(A^*)^3$) with a special type of concatenation which is described in the following Theorem:

▸ **Theorem 3.5** ([11, Theorem 5.3]). *Let $u, v \in \Sigma^*$. Then*

$$\chi(uv) = (\lambda(u)r, s, t\varrho(v))$$

where $s = \mu(u)\mathsf{rd}(v) \sqcap \mathsf{wrt}(u)\mu(v)$, $rs = \mu(u)\mathsf{rd}(v)$, and $st = \mathsf{wrt}(u)\mu(v)$. ◂

In other words, the multiplication of two words $u, v \in \Sigma^*$ can be understood as follows: at first we move the read actions from $\overline{\mathsf{rd}(v)}$ to the left such that each of its letters is directly preceded by exactly one write actions. If this is not possible (because $\lambda(v)$ is longer than $\varrho(u)$) we move the letters from $\overline{\mu(u)\lambda(v)}$ to the left until there is an alternating word of write and read actions. Now, if there is an infix $a\bar{b}$ with $a \neq b$ all of these read actions move one position to the left. We iterate this last step until there is no such infix. It is easy to see, that the new alternating word contains equal subsequences of write and read actions, respectively. Thereby, the read actions are the longest suffix of $\overline{\mu(u)\mathsf{rd}(v)}$ and the write actions the longest prefix of $\mathsf{wrt}(u)\mu(v)$ such that the equality of these subsequences holds.

3.3 The Monoid's Cayley-Graph

In this subsection we first recall the definition of Cayley-graphs for arbitrary, finitely generated monoids. Afterwards, we give some common properties as well as some special characteristics of the queue monoid's Cayley-graph.

▸ **Definition 3.6.** Let $\mathcal{M}$ be a monoid generated by a finite set $\Gamma \subseteq \mathcal{M}$. The *(right) Cayley-graph* of $\mathcal{M}$ is the edge-labeled, directed graph $\mathfrak{C}(\mathcal{M}, \Gamma) := (\mathcal{M}, (E_a)_{a\in\Gamma})$ with

$$E_a = \{(x, y) \in \mathcal{M} \mid y = xa\}$$

for each $a \in \Gamma$.

Similar to the right Cayley-graph, we may define the *left Cayley-graph* of $\mathcal{M}$ as the edge-labeled, directed graph $\mathfrak{LC}(\mathcal{M}, \Gamma) = (\mathcal{M}, (F_a)_{a\in\Gamma})$ with $F_a = \{(x, y) \in \mathcal{M} \mid y = ax\}$ for all $a \in \Gamma$.

▸ Remark. There is a strong relation between left and right Cayley-graphs of a monoid and Green's relations which are first introduced and studied in [9]. Recall that $x\mathcal{R}y$ iff $x\mathcal{M} = y\mathcal{M}$ for every $x, y \in \mathcal{M}$ and, similarly, $x\mathcal{L}y$ iff $\mathcal{M}x = \mathcal{M}y$. Then by [23, Proposition V.1.1] we have $x\mathcal{R}y$ ($x\mathcal{L}y$) if, and only if, x is strongly connected to y in $\mathfrak{C}(\mathcal{M}, \Gamma)$ ($\mathfrak{LC}(\mathcal{M}, \Gamma)$, resp.).

The concrete shape of the Cayley-graph of a monoid heavily depends on the chosen set of generators. For example, $\{-1, 1\}$ and $\{-2, 3\}$ are generating sets of $(\mathbb{Z}, +)$, but the resulting Cayley-graphs are not isomorphic (even if we remove the labels). Though, the chosen generating set has no influence on decidability and complexity of the FO and MSO theory of the Cayley-graph since the both problems are logspace reducible on each other (which we denote by $\approx_{\log}$):

▸ **Proposition 3.7** ([18, Proposition 3.1]). *Let Γ_1 and Γ_2 be two finite generating sets of the monoid $\mathcal{M}$. Then*

(1) $\mathsf{FOTh}(\mathfrak{C}(\mathcal{M}, \Gamma_1)) \approx_{\log} \mathsf{FOTh}(\mathfrak{C}(\mathcal{M}, \Gamma_2))$ *and*

(2) $\mathsf{MSOTh}(\mathfrak{C}(\mathcal{M}, \Gamma_1)) \approx_{\log} \mathsf{MSOTh}(\mathfrak{C}(\mathcal{M}, \Gamma_2))$. ◂

From now on we only consider the Cayley-graph of the queue monoid $\mathcal{Q}$. To simplify notation we write $\mathfrak{C}$ instead of $\mathfrak{C}(\mathcal{Q}, \Sigma)$ and $\mathfrak{LC}$ instead of $\mathfrak{LC}(\mathcal{Q}, \Sigma)$. First we prove some properties of $\mathfrak{C}$ and $\mathfrak{LC}$.

▸ **Proposition 3.8.** *The following statements hold:*
(1) $\mathsf{FOTh}(\mathfrak{C}) \approx_{\log} \mathsf{FOTh}(\mathfrak{LC})$ *and* $\mathsf{MSOTh}(\mathfrak{C}) \approx_{\log} \mathsf{MSOTh}(\mathfrak{LC})$.
(2) $\mathfrak{C}$ *is an acyclic graph with root* $[\varepsilon]$.
(3) $\mathfrak{C}$ *has unbounded (in-)degree.*

Proof. At first, we prove (1). Let the duality function $\delta\colon \Sigma^* \to \Sigma^*$ be defined as follows:

$$\delta(\varepsilon) = \varepsilon,\ \delta(au) = \delta(u)\overline{a}\,,\ \text{and}\ \delta(\overline{a}u) = \delta(u)a$$

for all $u \in \Sigma^*$ and $a \in A$. In other words, δ reverses the order of the actions and inverts writing and reading of a letter a. From [11, Proposition 3.4] we know $u \equiv v$ iff $\delta(u) \equiv \delta(v)$. Hence, δ is an automorphism on $\mathcal{Q}$ and $(p, q) \in E_\alpha$ iff $(\delta(p), \delta(q)) \in F_{\delta(\alpha)}$ for all $p, q \in \mathcal{Q}$ and $\alpha \in \Sigma$. Let $\varphi \in \mathsf{FO}[(E_\alpha)_{\alpha\in\Sigma}]$ ($\varphi \in \mathsf{MSO}[(E_\alpha)_{\alpha\in\Sigma}]$, resp.). We construct φ' by replacing any atom "$E_\alpha(x, y)$" in φ by "$F_{\delta(\alpha)}(x, y)$". Then

$$\mathfrak{C} \models \varphi[q_1, \ldots, q_k] \iff \mathfrak{LC} \models \varphi'[\delta(q_1), \ldots, \delta(q_k)]$$

for any $q_1, \ldots, q_k \in \mathcal{Q}$. In particular, $\varphi \in \mathsf{FOTh}(\mathfrak{C})$ iff $\varphi' \in \mathsf{FOTh}(\mathfrak{LC})$ (resp. $\varphi \in \mathsf{MSOTh}(\mathfrak{C})$ iff $\varphi' \in \mathsf{MSOTh}(\mathfrak{LC})$). Finally, the converse reduction is symmetric to the one described above.

Now, we prove (2). Due to [11, Corollary 4.7] we have $p\mathcal{R}q$ iff $p = q$ for all $p, q \in \mathcal{Q}$. Then, by the remark above $p, q \in \mathcal{Q}$ are strongly connected iff $p = q$, i.e., there are no cycles in $\mathfrak{C}$.

Next, to prove (3) let $n \in \mathbb{N}$ and $a, b \in A$ with $a \neq b$. Set $w_k = \overline{a^k}(a\overline{a})^{n-k}a^k$ for any $0 \leqslant k \leqslant n$. Then $w_k \equiv w_\ell$ (i.e. $[w_k] = [w_\ell]$) iff $k = \ell$ for any $0 \leqslant k, \ell \leqslant n$. By Theorem 3.5 we have $\chi(w_k\overline{b}) = (a^n b, \varepsilon, a^n)$, i.e. $w_k\overline{b} \equiv w_\ell\overline{b}$ for any $0 \leqslant k, \ell \leqslant n$. Hence, we have $([w_k], [\overline{a^n b}a^n]) \in E_{\overline{b}}$ for all $0 \leqslant k \leqslant n$, i.e., the node $[\overline{a^n b}a^n]$ has in-degree $> n$. ◂

By $\mathfrak{G}_n$ we denote the $n \times n$*-grid* for $n \in \mathbb{N}$. This is an undirected graph with n^2 many nodes which we denote by $v_{i,j}$ for any $1 \leqslant i, j \leqslant n$. Thereby, we have an edge between $v_{i,j}$ and $v_{k,\ell}$ if, and only if, $|j - \ell| + |i - k| = 1$ holds. Additionally, for a Γ-labeled, directed graph $\mathfrak{G} = (V, (E_a)_{a\in\Gamma})$ we denote the unlabeled and undirected version by $\mathsf{ud}(\mathfrak{G}) = (V, E)$. Here, we have an edge $(v, w) \in E$ if, and only if, there is an $a \in \Gamma$ such that $(v, w) \in E_a$ or $(w, v) \in E_a$. Then, in $\mathsf{ud}(\mathfrak{C})$ we can find $\mathfrak{G}_n$ for any $n \in \mathbb{N}$:

▸ **Proposition 3.9.** $\mathfrak{G}_n$ *is an induced subgraph of* $\mathsf{ud}(\mathfrak{C})$ *for any* $n \in \mathbb{N}$.

Proof. Let $a, b \in A$ be distinct. Then the submonoid $\mathcal{M}$ of $\mathcal{Q}$ generated by a and $\overline{b}$ is the free commutative monoid on $\{a, \overline{b}\}$ by Theorem 3.3(1). Its Cayley-graph $\mathfrak{C}(\mathcal{M}, \{a, \overline{b}\})$ is an infinite grid with labeled, directed edges (see Fig. 3). Then, $\mathfrak{G}_n$ is an induced subgraph of $\mathsf{ud}(\mathfrak{C}(\mathcal{M}, \{a, \overline{b}\}))$. Since in $\mathfrak{C}$ there are no edges with labels other than a or $\overline{b}$ between the nodes from $\mathcal{M}$, $\mathsf{ud}(\mathfrak{C}(\mathcal{M}, \{a, \overline{b}\}))$ is an induced subgraph of $\mathsf{ud}(\mathfrak{C})$ as well implying our claim. ◂

With the help of a famous result from Seese (cf. [26]), we may now prove the undecidability of the monadic second-order theory of the queue monoid's Cayley-graph.

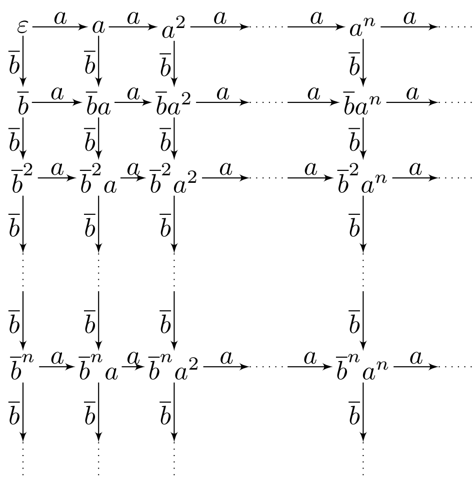


Figure 3 $\mathfrak{C}$ restricted to the nodes reachable by a- and $\bar{b}$-edges, only.

▶ **Corollary 3.10.** $\mathsf{MSOTh}(\mathfrak{C})$ *is undecidable.*

Proof. Due to [24] each planar graph is a minor of some grid $\mathfrak{G}_n$. Since each $\mathfrak{G}_n$ is an induced subgraph of $\mathsf{ud}(\mathfrak{C})$ by Proposition 3.9, each planar graph is minor of an induced subgraph of $\mathsf{ud}(\mathfrak{C})$. Hence, by [26, Theorem 5] $\mathsf{MSOTh}(\mathsf{ud}(\mathfrak{C}))$ is undecidable. Since $\mathsf{ud}(G)$ is interpretable in $\mathsf{FOTh}(\mathfrak{C})$, $\mathsf{MSOTh}(\mathfrak{C})$ is undecidable as well. ◀

4 Decidability of the FO-Theory

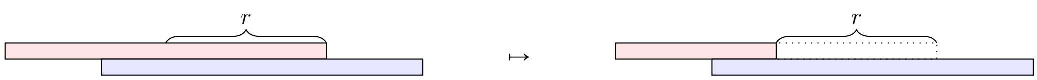
Recall that the Cayley-graph of the queue monoid $\mathcal{Q}$ induced by A is denoted by $\mathfrak{C} = (\mathcal{Q}, (E_\alpha)_{\alpha\in\Sigma})$. For $p, q \in \mathcal{Q}$ let $p\Delta q = (\mathsf{rd}(p), \mathsf{wrt}(p))\Delta(\mathsf{rd}(q), \mathsf{wrt}(q))$ and we call $|p\Delta q|$ the (Δ-)distance of p and q.

Let us first give an intuitive description of our decidability proof. We follow a classical proof strategy due to Ferrante and Rackoff [7]. More precisely we show that for every two $(r+1)$-equivalent tuples $\vec{p}, \vec{q} \in \mathcal{Q}^n$ and every $p \in \mathcal{Q}$ there is a q in the $f(r+1)$-neighborhood of the tuple $\vec{q}$ such that $(\vec{p}, p) \equiv_r (\vec{q}, q)$ for some fixed primitive recursive function f. This implies that in order to evaluate a formula $Qx\varphi(\vec{p})$ we can restrict quantification

▶ **Definition 4.1.** Let V be an r-skeleton. We say that $q \in \mathcal{Q}$ is *compatible* with V if V has an instantiation v such that $\mathsf{rd}(q) = \mathsf{rd}(v)x$ for some $x \in A^{\leqslant r}$ and $|\mathsf{wrt}(q)\Delta\mathsf{wrt}(v)| \leqslant r$.

Intuitively, q being compatible to an r-skeleton V means that we can obtain an element q' with r-skeleton V by deleting up to r many read actions and modifying the write actions arbitrarily up to distance r. We use this notion in order to translate elements of the Cayley-graph into positions of an r-skeleton.

▶ **Definition 4.2.** For $q \in \mathcal{Q}$ with $|\mu(q)| \geqslant r$ let $\mathrm{rc}_r(q)$ be the element q' with $\mathsf{wrt}(q') = \mathsf{wrt}(q)$, $\mathsf{rd}(q') = \mathsf{rd}(q)\,\mathrm{suf}_r(\mathsf{rd}(q))^{-1}$, and $\mu(q') = \mu(q)\,\mathrm{suf}_r(\mu(q))^{-1}$.



We describe the way, in which we associate positions in an r-skeleton with elements of $\mathcal{Q}$ and vice versa.

▸ **Definition 4.3.** Let $p, q \in \mathcal{Q}$ and let U and V be the $3r$-skeletons of $\mathrm{rc}_{2r}(p)$ and $\mathrm{rc}_{2r}(q)$ respectively. If we suppose that $(m_1, \ldots, m_k)$ are positions in V and $(n_1, \ldots, n_k)$ are positions in U such that $(U, m_1, \ldots, m_k) \equiv_\ell (V, n_1, \ldots, n_k)$ for some $\ell \geqslant 1$. For $p' \in \mathcal{Q}$ with $|p' \Delta p| \leqslant r$ we associate a position m_{k+1} in U as follows: Let $(u_1, \ldots, u_m)$ be the complete border-decomposition of $\mathsf{rd}(\mathrm{rc}_{2r}(p))$ and $(v_1, \ldots, v_n)$ be the complete border-decomposition of $\mathsf{rd}(\mathrm{rc}_{2r}(q))$. As p' has distance at most r from p we have that $\mathsf{rd}(p') = \mathsf{rd}(\mathrm{rc}_{2r}(p))x$ for some $x \in A^{\leqslant 2r}$. Therefore there is an $i \leqslant m$ such that $\mu(p') = u_i x$. Then i is the position that is associated with p'.

Now let n_{k+1} be such that $(U, m_1, \ldots, m_{k+1}) \equiv_{\ell-1} (V, n_1, \ldots, n_{k+1})$ we associate an element q' with n_{k+1} as follows: Let q' be the element with $\mathsf{rd}(q') = \mathsf{rd}(\mathrm{rc}_{2r}(q)) u_{m_{k+1}}^{-1} \mu(p')$, $\mathsf{wrt}(q') \Delta \mathsf{wrt}(\mathrm{rc}_r(q)) = \mathsf{wrt}(p') \Delta \mathsf{wrt}(\mathrm{rc}_{2r}(p))$, and $\mu(q') = v_{m_{k+1}} u_{n_{k+1}}^{-1} \mu(p')$. Note that q' is well defined since $V[j]$ is labeled by $\mathrm{pref}_{2^{r+2}}(u_i^{-1}\mu(p))$. Therefore $v_j \, \mathrm{pref}_{2^{r+1}}(v_i^{-1}\mu(p))$ is a prefix of $\mathsf{wrt}(q')$ by construction.

The basic idea behind this definition is to ensure that the neighborhood structure of the elements p' and q' is ℓ-equivalent. We use this idea to define a family of equivalence relations $(E^r_m)_{r,m \in \mathbb{N}}$. For $r, m \in \mathbb{N}$ and $\vec{p}, \vec{q} \in \mathcal{Q}^m$ let $\vec{p} E^r_m \vec{q}$ iff

(1) $\mathrm{suf}_{2^r}(\mathsf{rd}(p_i)) = \mathrm{suf}_{2^r}(\mathsf{rd}(q_i))$ and $\mathrm{suf}_{2^r}(\mathsf{wrt}(p_i)) = \mathrm{suf}_{2^r}(\mathsf{wrt}(q_i))$ for all $1 \leqslant i \leqslant m$.

(2) $|p_i \Delta p_j| =_{2^r} |q_i \Delta q_j|$ for all $1 \leqslant i, j \leqslant m$ and if $|p_i \Delta p_j| \leqslant 2^r$ then also $p_i \Delta p_j = q_i \Delta q_j$.

(3) There is a partition $X_1, \ldots, X_k$ of $\{1, \ldots, m\}$ such that for $X \neq X' \in \{X_1, \ldots, X_k\}$ it holds that:

  (a) If $i \in X$, $j \in X'$ it holds that $|p_i \Delta p_j| > 2^r$ (and therefore $|q_i \Delta q_j| > 2^r$).

  (b) Let $i = \min X$. Then for all $j \in X$ it holds that $|p_i \Delta p_j| \leqslant \sum_{s=r+m-i}^{r} 2^s$ (and therefore also $|q_i \Delta q_j| \leqslant \sum_{s=r+m-i}^{r} 2^s$).

  (c) Let $i = \min X$ and let U be the $3 \cdot 2^{r+m-i+1}$-skeleton of $\mathrm{rc}_{2^{r+m-i+2}}(p_i)$ and V be the $3 \cdot 2^{r+m-i+1}$-skeleton $\mathrm{rc}_{2^{r+m-i+2}}(q_i)$. Then for all $j \in X$ we have that p_j is compatible with U and q_j is compatible with V. Further if $m_1, \ldots, m_k$ are the positions in U that are associated with $\{p_j \mid j \in X_i\}$ and $n_1, \ldots, n_k$ are the positions in U that are associated with $\{q_j \mid j \in X_i\}$ then $(V, m_1, \ldots, m_k) \equiv_{r+1} (U, n_1, \ldots, n_k)$.

▸ **Lemma 4.4.** *For all $m \in \mathbb{N}_{>0}$ and all $\vec{p}, \vec{q} \in \mathcal{Q}^m$: If $\vec{p} E^0_m \vec{q}$ then the mapping $p_i \mapsto q_i$ is a partial isomorphism.*

Proof. We need to show that $(p_i, p_j) \in E_\alpha \Rightarrow (q_i, q_j) \in E_\alpha$ for all $i, j \leqslant m$ and all $\alpha \in \Sigma$. Let $\vec{p}, \vec{q} \in \mathcal{Q}^m$ with $\vec{p} E^0_m \vec{q}$. Suppose $(p_i, p_j) \in E_\alpha$ for some $\alpha \in \Sigma$. Then $|p_i \Delta p_j| = 1$. Hence $p_i \Delta p_j = q_i \Delta q_j$ by (2). Since the distance between p_i and p_j and between q_i and q_j is 1, there are 2^ℓ-skeletons (for some $\ell \geqslant m - \min\{i, j\} + 2$) U, V such that p_i and p_j can be translated into positions m_1, m_2 in U and q_i and q_j can be translated into position n_1, n_2 in V such that $(U, m_1, m_2) \equiv_1 (V, n_1, n_2)$. There are two possible types of configurations for p_i and p_j such that they can be connected by an edge. First, it might be the case that $\mathsf{rd}(p_i) = \mathsf{rd}(p_j)$, $\mathsf{wrt}(p_i)\alpha = \mathsf{wrt}(p_j)$, and $\mu(p_i) = \mu(p_j)$. In this case $m_1 = m_2$ and therefore $n_1 = n_2$, which implies that $\mathsf{rd}(q_i) = \mathsf{rd}(q_j)$, $\mathsf{wrt}(q_i)\alpha = \mathsf{wrt}(q_j)$, and $\mu(q_i) = \mu(q_j)$. Therefore $(q_i, q_j) \in E_\alpha$.

Second, it might be that $\mathsf{rd}(p_i)a = \mathsf{rd}(p_j)$ (where $\alpha = \overline{a}$), $\mathsf{wrt}(p_i) = \mathsf{wrt}(p_j)$, and $\mu(p_j)a^{-1}$ is the largest suffix w of $\mu(p_i)$ such that wa is a prefix of $\mathsf{wrt}(p_i)$. This property can be translated into a formula φ on (U, m_1, m_2) of quantifier rank 1. As $(U, m_1, m_2) \equiv_1 (V, n_1, n_2)$, $(V, n_1, n_2) \models \varphi$ and therefore $(q_i, q_j) \in E_\alpha$. ◂

In order to prove the main technical lemma we need to construct a "small" r-equivalent words from a given word w. This is routine since it can be achieved by a simple automata-theoretic approach.

▸ **Lemma 4.5.** *From a given alphabet Γ, a word $v \in \Gamma^*$, and $r \in \mathbb{N}$ one can compute an automaton $\mathcal{A}$ in time $\exp_{r+1}(2, f(r))$ with $L(\mathcal{A}) = \{w \in \Gamma^* \mid w \equiv_r v\}$ for some polynomial f.*

Proof sketch. Construct a first-order formula φ that characterizes the r-type of v. From φ compute an automaton $\mathcal{A}_\varphi$ with $L(\mathcal{A}_\varphi) = \{w \in \Gamma^* \mid w \equiv_r v\}$. One easily show via induction on r that the size of the automaton $\mathcal{A}$ is at most $\exp_{r+1}(2, f(r))$ for some suitable polynomial f. ◂

▸ **Lemma 4.6.** *For all $m, r \in \mathbb{N}$ and all $\vec{p}, \vec{q} \in \mathcal{Q}^m$:*

$$\vec{p} E_m^{r+1} \vec{q} \Rightarrow \forall p \in \mathcal{Q} \exists q \in \mathcal{N}_{\exp_{r+2}(2,f(r))}(\vec{q}) : (\vec{p}, p) E_{m+1}^r (\vec{q}, q)$$

for some polynomial f.

Proof. Let $\vec{p}, \vec{q} \in \mathcal{Q}^m$ with $(\vec{p}, \vec{q}) \in E_m^{r+1}$ and let $X_1, \ldots, X_k$ be a partition of $\{1, \ldots, m\}$ with the properties described in (3). Further let $X_i(\vec{p}) = \{p_j \mid j \in X_i\}$ and $X_i(\vec{q}) = \{q_j \mid j \in X_i\}$. Consider $p \in \mathcal{Q}$. We distinguish three cases. If p has distance $\leqslant 4 \exp_{r+2}(2, f(r))$ from ε then we choose $q = p$.

From now on suppose p has distance $> 4 \exp_{r+2}(2, f(r))$ from ε. We consider the case that p has distance $> 2^r$ from every p_i. Since the distance from ε is exactly $|\overline{\pi}_1(p)| + 2|\mu(p)| + |\varrho(p)|$ it follows that $|\overline{\pi}_1(p)| > \exp_{r+2}(2, f(r))$ or $|\mu(p)| > \exp_{r+2}(2, f(r))$ or $|\varrho(p)| > \exp_{r+2}(2, f(r))$. Basically, we want to use the $3 \cdot 2^{r+1}$-skeleton of p to construct a suitable answer q. However, we need to cut the last 2^{r+1} read actions in order to avoid certain problems that would occur if we want to translate elements in close proximity to p into positions of the $3 \cdot 2^{r+1}$-skeleton. Let $p' = \mathrm{rc}_{2^{r+2}}(p)$. Consider the $3 \cdot 2^{r+1}$-skeleton $V = \mathcal{S}_{3 \cdot 2^{r+1}}(p')$. By Lemma 4.5 we can construct a $3 \cdot 2^{r+1}$-skeleton W of length at most $\exp_{r+1}(2, f(r))$. From W we construct the 2^{r+2}-instantiation w. Using Lemma 2.1 we can choose words u, v of length at most 2^{r+1} such that

- $\mathrm{suf}_{2^r}(uw) = \mathrm{suf}_{2^r}(\mathsf{rd}(p)\, \mathrm{suf}_{2^r}(\mathsf{rd}(p))^{-1})$,
- $\mathrm{suf}_{2^r}(wv) = \mathrm{suf}_{2^r}(\mathsf{wrt}(p))$,
- $\mathrm{pref}_{2^r}(wv) = \mathrm{pref}_{2^r}(\mathsf{wrt}(p))$, and
- $uw \sqcap wv = w$.

Let $(v_0, v_1, \ldots, v_m)$ be the complete border-decomposition of $\mathsf{wrt}(p') \sqcap \mathsf{rd}(p')$ and let $(w_0, w_1, \ldots, w_n)$ be the complete border-decomposition of w. Let i be the index of $\mu(p')$ in $(v_0, v_1, \ldots, v_m)$. Because $\mathcal{S}_{3 \cdot 2^{r+1}}(p') \equiv_{r+1} W$ there is a $j \in \{0, \ldots, n\}$ such that $(\mathcal{S}_{2^{r+2}}(p'), i) \equiv_r (W, j)$. Now let q be the element associated to j.

Finally, if p has distance $\leqslant 2^r$ from some p_i then let $Y \in \{X_1, \ldots, X_k\}$ be such that $i \in Y$ and let $j = \min Y$. Let U be the $3 \cdot 2^{r+m-j+1}$-skeleton of $\mathrm{rc}_{2^{r+m-j+2}}(p_j)$ and V be the $3 \cdot 2^{r+m-j+1}$-skeleton of $\mathrm{rc}_{2^{r+m-j+2}}(q_j)$. Then p is compatible with U. Let $m_1, \ldots, m_\ell$ be the positions in U that are associated with the elements $\{q_s \mid s \in Y\}$, $m_{\ell+1}$ the position in U that is associated with p, and $n_1, \ldots, n_\ell$ be the positions associated with $\{q_s \mid s \in Y\}$ in V. Since $(U, m_1, \ldots, m_\ell) \equiv_{r+2} (V, n_1, \ldots, n_\ell)$ by Property (3c) there exists a $n_{\ell+1}$ with $(U, m_1, \ldots, m_{\ell+1}) \equiv_{r+1} (V, n_1, \ldots, n_{\ell+1})$. From $n_{\ell+1}$ we compute the associated element q in the $(\sum_{s=r+m-i}^{r} 2^s)$-neighborhood of q_j. The construction of q ensures that Properties (1) to (3) are fulfilled for $(\vec{p}, p)$ and $(\vec{q}, q)$ by adding $\ell + 1$ to Y. Hence $(\vec{p}, p) E_m^r (\vec{q}, q)$. ◂

The Lemmata 4.4 and 4.6 ensure that E_m^r-equivalent tuples are also r-equivalent.

▸ **Corollary 4.7.** *For all $\vec{p} \in \mathcal{Q}^m$, $p \in \mathcal{Q}$, and $r \in \mathbb{N}$ there exists an element $q \in \mathcal{N}_{\exp_{r+2}(2,f(r))}(\vec{p})$ with $(\mathfrak{C}, \vec{p}, p) \equiv_r (\mathfrak{C}, \vec{p}, q)$ for some polynomial f.*

▶ **Lemma 4.8.** *For every $p \in \mathcal{Q}$ and every r there are at most $|A|^{4r}(\min\{|\mathsf{rd}(p)|, |\mathsf{wrt}(p)|\}+r)$ many elements in the r-neighborhood of a node $p \in \mathcal{Q}$.*

Proof. Every element q in the r-neighborhood of p can be characterized by the tuple $p\Delta q = (u, v, w, x) \in (A^{\leqslant r})^4$ and $\mu(q)$. Once we have fixed $p\Delta q \in (A^{\leqslant r})^4$ (and therefore fixed $\mathsf{rd}(q)$ and $\mathsf{wrt}(q)$) there are at most $\min\{|\mathsf{rd}(q)|, |\mathsf{wrt}(q)|\} \leqslant \min\{|\mathsf{rd}(p)|, |\mathsf{wrt}(p)|\}+r$ possible values for $\mu(q)$. ◀

▶ **Theorem 4.9.** $\mathsf{FOTh}(\mathfrak{C})$ *is primitive recursive.*

Proof. We use the standard model-checking algorithm for first-order logic but restrict quantification to the $\exp_{r+1}(2, f(r))$-neighborhood of the current variable assignment. The correctness of this procedure is guaranteed by Corollary 4.7. We see that the values $|\mathsf{rd}(p)|$ and $|\mathsf{wrt}(p)|$ are bounded by $r \exp_{r+1}(2, f(r))$ Hence, by Lemma 4.8 the algorithm needs to consider at most $|A|^{4r} r(\exp_{r+1}(2, f(r)) + 1)$ many Elements, which leads to a runtime of $|\varphi| \cdot (|A|^{4r} r(\exp_{r+1}(2, f(r)) + 1))^r$, which is obviously a primitive recursive function. ◀

5 Conclusion and Open Problems

We studied the Cayley-graph of the queue monoid and the logics of these graphs. Concretely, we have shown the decidability of the Cayley-graph's first order theory and the undecidability of the monadic second-order theory. This answers a question from Huschenbett et al. in [11].

In Table 1 is a comparison of our results compared to other fundamental data structures.

Data Structure	Transformation Monoid $\mathcal{M}$	$\mathsf{FOTh}(\mathfrak{C}(\mathcal{M}, \Gamma))$	$\mathsf{MSOTh}(\mathfrak{C}(\mathcal{M}, \Gamma))$
finite monoid	finite monoid	PSPACE [8]	PSPACE [8]
counter	$(\mathbb{Z}, +)$	2EXPSPACE [19]	decidable [18]
stack	polycyclic monoid	2EXPSPACE [19]	decidable [6, 18]
queue	queue monoid	primitive recursive	undecidable

Table 1 Comparison of the decidability of logics on Cayley-graphs of fundamental data structures.

There are still some questions open relating to the queue monoid: in this paper we have given an primitive recursive but non-elementary upper bound on the complexity of the first-order theory of the queue monoid's Cayley-graph. So, one may ask for tight upper and lower bounds. Another open question concern the automaticity of the queue monoid. While it is neither automatic in the sense of Khnoussainov and Nerode [14] nor automatic in the sense of Thurston et al. [4] due to [11], we still do not know whether the Cayley-graph of the queue monoid is automatic. Finally, the decidability of the first-order theory of the (partially) lossy queue monoid's (cf. [15, 16]) Cayley-graph is left open as well and is worth to be studied.

References

1 Parosh A. Abdulla and Bengt Jonsson. Verifying programs with unreliable channels. *Information and Computation*, 127(2):91–101, 1996. doi:10.1006/inco.1996.0053.

2 Benedikt Bollig. *Formal Models of Communicating Systems: Languages, Automata, and Monadic Second-Order Logic*. Springer-Verlag, Berlin Heidelberg, 2006. doi:10.1007/3-540-32923-4.

3 Daniel Brand and Pitro Zafiropulo. On Communicating Finite-State Machines. *Journal of the ACM*, 30(2), 1983. doi:10.1145/322374.322380.

4 James W Cannon, David BA Epstein, Derek F Holt, Silvio VF Levy, Michael S Paterson, and William P Thurston. Word processing in groups. *Jones and Barlett Publ., Boston, MA*, 1992.

5 Gérard Cécé, Alain Finkel, and S. Purushotaman Iyer. Unreliable channels are easier to verify than perfect channels. *Information and Computation*, 124(1):20–31, 1996. doi:10.1006/inco.1996.0003.

6 Christian Delhommé, Teodor Knapik, and D. Gnanaraj Thomas. Using Transitive–Closure Logic for Deciding Linear Properties of Monoids. In *Mathematical Foundations of Computer Science 2003*, Lecture Notes in Computer Science, pages 378–387. Springer, Berlin, Heidelberg, August 2003. doi:10.1007/978-3-540-45138-9_32.

7 Jeanne Ferrante and Charles W. Rackoff. *The Computational Complexity of Logical Theories*. Number 718 in Lecture Notes in Mathematics. Springer, 1979. doi:10.1007/BFb0062837.

8 Erich Grädel. Finite Model Theory and Descriptive Complexity. In *Finite Model Theory and Its Applications*, Texts in Theoretical Computer Science an EATCS Series, pages 125–230. Springer, Berlin, Heidelberg, 2007. doi:10.1007/3-540-68804-8_3.

9 James A. Green. On the structure of semigroups. *Annals of Mathematics*, pages 163–172, 1951.

10 Christoph Haase, Sylvain Schmitz, and Philippe Schnoebelen. The power of priority channel systems. *Logical Methods in Computer Science*, 10(4:4), 2014.

11 Martin Huschenbett, Dietrich Kuske, and Georg Zetzsche. The monoid of queue actions. *Semigroup forum*, 95(3):475–508, 2017. doi:10.1007/s00233-016-9835-4.

12 Mark Kambites. Formal languages and groups as memory. *Communications in Algebra*, 37(1):193–208, 2009. doi:10.1080/00927870802243580.

13 Olga Kharlampovich, Bakhadyr Khoussainov, and Alexei Miasnikov. From automatic structures to automatic groups. *Groups, Geometry, and Dynamics*, 8(1):157–198, 2014. doi:10.4171/GGD/221.

14 Bakhadyr Khoussainov and Anil Nerode. Automatic presentations of structures. In *Logic and Computational Complexity*, volume 960 of *Lecture Notes in Computer Science*, pages 367–392. Springer, 1995. doi:10.1007/3-540-60178-3_93.

15 Chris Köcher. Rational, Recognizable, and Aperiodic Sets in the Partially Lossy Queue Monoid. In *STACS'18*, volume 96 of *LIPIcs*, pages 45:1–45:14. Dagstuhl Publishing, 2018. doi:10.4230/LIPIcs.STACS.2018.45.

16 Chris Köcher, Dietrich Kuske, and Olena Prianychnykova. The Inclusion Structure of Partially Lossy Queue Monoids and their Trace Submonoids. *RAIRO - Theoretical Informatics and Applications*, 2018.

17 Dietrich Kuske and Markus Lohrey. Logical aspects of Cayley-graphs: The group case. *Annals of Pure and Applied Logic*, 131(1):263–286, 2005. doi:10.1016/j.apal.2004.06.002.

18 Dietrich Kuske and Markus Lohrey. Logical aspects of Cayley-graphs: The monoid case. *International Journal of Algebra and Computation*, 16(02):307–340, 2006. doi:10.1142/S0218196706003001.

19 Dietrich Kuske and Markus Lohrey. Automatic structures of bounded degree revisited. *The Journal of Symbolic Logic*, 76(4):1352–1380, December 2011. doi:10.2178/jsl/1318338854.

20 Benoît Masson and Philippe Schnoebelen. On verifying fair lossy channel systems. In *MFCS'02*, volume 2420 of *Lecture Notes in Computer Science*, pages 543–555. Springer, 2002. doi:10.1007/3-540-45687-2_45.

21 David E. Muller and Paul E. Schupp. The theory of ends, pushdown automata, and second-order logic. *Theoretical Computer Science*, 37:51–75, January 1985. doi:10.1016/0304-3975(85)90087-8.

22 Paliath Narendran and Friedrich Otto. Some results on equational unification. In *10th International Conference on Automated Deduction*, Lecture Notes in Computer Science, pages 276–291. Springer, Berlin, Heidelberg, July 1990. doi:10.1007/3-540-52885-7_94.

23 Jean-Éric Pin. Mathematical foundations of automata theory. *Lecture notes LIAFA, Université Paris*, 7, 2010.

24 Neil Robertson and Paul D. Seymour. Graph minors, Part III: Planar tree-width. *Journal of Combinatorial Theory, Series B*, 36(1):49–64, February 1984. doi:10.1016/0095-8956(84)90013-3.

25 Jacques Sakarovitch. Kleene's theorem revisited. In *Trends, Techniques, and Problems in Theoretical Computer Science*, Lecture Notes in Computer Science, pages 39–50. Springer, Berlin, Heidelberg, October 1986. doi:10.1007/3540185356_29.

26 D. Seese. The structure of the models of decidable monadic theories of graphs. *Annals of Pure and Applied Logic*, 53(2):169–195, July 1991. doi:10.1016/0168-0072(91)90054-P.