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

Faried Abu Zaid[1]

Camelot Management Consultants, CoE Artificial Intelligence for Information Management, Munich
faza@camelot-mc.com

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.

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 a 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 [1, 5, 22]) or if letters of low priority can be superseded by letters with higher priority (cf. [12]).

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. [14, 27]),

---

and queues induce the so-called queue monoids which were first introduced in [13]. 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 [17, 18]. 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 [19]. 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) [19,23]. Besides these results, Kharlampovich et al. considered in [15] so-called Cayley-graph automatic groups (these are the groups having an automatic Cayley-graph in the sense of [16]) 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 [20, 24]. For finite monoids the Cayley-graphs are finite and, hence, the first- and second-order theories are complete for polynomial space and exponential space, respectively [10]. 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, 20] the Cayley-graphs monadic second-order theory is decidable (the first-order theory is even in 2EXPSPACE by [21]).

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 a 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 [8] and an automata-based approach. This gives an answer on a question raised by Huschenbett, Kuske, and Zetzsche [13]. There, they conjectured the undecidability of its first-order logic implying that the graph is not automatic in the sense of [16]. Moreover, we will prove the undecidability of the monadic second-order theory with the help of a well-known result from Seese [28].

2   Preliminaries

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 $uw^{-1} = v$. Thereby, v is the *complementary prefix* of w wrt. u and w the *complementary suffix* of v wrt. u. For $u, v \in A^*$ let $u \sqcap v$ denote the largest suffix of u that is also a prefix of v.

For $m, n, r \in \mathbb{N}$ we write $m =_r n$ iff $m = n$ or $m, n > r$. The function $\exp_r(n)$ is inductively defined by $\exp_0(n) = n$ and $\exp_{r+1}(n) = 2^{\exp_r(n)}$.

Logic on Graphs and Words

Let A be a finite set of labels. An *edge-labeled graph* is a tuple $G = (V^G, (E_a^G)_{a \in A})$ where V is the set of vertices and $E_a^G \subseteq V \times V$ is the set of a-labeled edges. A *word-structure* over A is a tuple $W = (\{0, \ldots, n-1\}, \leqslant^W, (P_a^W)_{a \in A})$ where $\leqslant^W$ is the usual order on $\{0, \ldots, n-1\}$, and $(P_a^W)_{a \in A}$ is a partition of $\{0, \ldots, n-1\}$ (some of the sets P_a^W may be empty). Whenever we use logic to describe properties of a word w then the formula is evaluated on the corresponding word structure W.

Let $\tau = \{R_1, \ldots, R_m, c_1, \ldots, c_n\}$ where R_i is a relation symbol of arity r_i and c_j is a constant symbol. *First-order formulas* (over the vocabulary τ) are build up from variables and constant symbols $\{x_i \mid i \in \mathbb{N}\} \cup \{c_1, \ldots, c_n\}$, the edge relation symbols $\{R_1, \ldots, R_m\}$, the equality symbol $=$, the Boolean connectives $\{\neg, \vee, \wedge, \rightarrow\}$, quantifiers $\{\forall, \exists\}$, and the bracket symbols $\{(,)\}$. We write $G \models \varphi$ to denote that the formula φ is satisfied by the structure G. The *quantifier rank* $\mathrm{qr}(\varphi)$ of a formula φ is the maximal nesting depth of quantifiers within φ. Two structures G and H are *r-equivalent* (denoted $G \equiv_r H$) if they cannot be distinguished by any formula of quantifier rank $\leqslant r$. For a structure G and two tuples $\vec{p}, \vec{q} \in (V^G)^m$ we write $\vec{p} \equiv_r^G \vec{q}$ or say that $\vec{p}$ and $\vec{q}$ are r-equivalent in G whenever $G \models \varphi(\vec{p}) \Leftrightarrow G \models \varphi(\vec{q})$ for all first-order formulas φ with m free variables and quantifier rank at most r. For all the above notations we adopt the convention that we omit superscripts whenever this should not lead to any confusion. For instance we write $\vec{p} \equiv_r \vec{q}$ when the underlying structure G is clear from the context.

The *r-type* of a structure G is the set of all first-order sentences φ of quantifier rank at most r such that $G \models \varphi$. It is well known that there are up to equivalence only finitely many sentences of quantifier rank at most r. Hence the r-type of a structure can be characterized by a sentence, which has also quantifier rank r.

Ehrenfeucht-Fraïssé-relations (resp. *EF-relations*) for a graph $G = (V, (E_a)_{a \in A})$ are a system $(\mathcal{E}_m^r)_{r, m \in \mathbb{N}}$ where $\mathcal{E}_m^r$ is an equivalence relation on V^m and the following is true for all $r, m \in \mathbb{N}$ and $\vec{p}, \vec{q} \in V^m$:

- If $(p_1, \ldots, p_m)\mathcal{E}_m^0(q_1, \ldots, q_m)$ then the mapping $p_i \mapsto q_i$ is a partial isomorphism, that is $p_i = p_j \Leftrightarrow q_i = q_j$ and $(p_i, p_j) \in E_a \Leftrightarrow (q_i, q_j) \in E_a$ for all $1 \leqslant i, j \leqslant m$ and all $a \in A$.
- If $\vec{p}\mathcal{E}_m^{r+1}\vec{q}$ then for every $p \in V$ there exists a $q \in V$ such that $(\vec{p}, p)\mathcal{E}_{m+1}^r(\vec{q}, q)$.

Ehrenfeucht-Fraïssé-relations are useful to identify r-equivalent tuples in a graph. This is formalized in the following theorem.

▸ **Theorem 2.1** ([7, 9])**.** *Let G be a graph, $(\mathcal{E}_m^r)_{r, m \in \mathbb{N}}$ Ehrenfeucht-Fraïssé-relations for G, and $\vec{p}, \vec{q}$ m-tuples of nodes from G. If $\vec{p}\mathcal{E}_m^r\vec{q}$ then $\vec{p} \equiv_r \vec{q}$.*

3 Queue Monoid and its Cayley-Graph

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 $\bar{a}$). Thereby, $\overline{A}$ is a disjoint copy of A containing all reading actions $\bar{a}$ and $\Sigma := A \uplus \overline{A}$ 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 $\bar{a} \in \overline{A}$ 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 [17, 18], 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*. The neutral element of $\mathcal{Q}(A)$ is $[\varepsilon]_{\equiv} = \{\varepsilon\}$, which we will denote simply by ε.

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.

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** ([13, 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 since we have

$$\varepsilon \circ a\overline{a} = \varepsilon \neq \perp = \varepsilon \circ \overline{a}a \, .$$

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 $\mathcal{R}$ [13, 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 $[u]_\equiv$. 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 $u = abb\overline{ab}$. Then we have

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

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 ($[u]_\equiv$, 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. The concatenation of any transformation $u \in \Sigma^*$ with a single letter is described in the lemma below.

▸ **Lemma 3.5.** *Let $u \in \Sigma^*$ and $a \in A$. Then we have*

$$\chi(ua) = (\lambda(u), \mu(u), \varrho(u)a) \quad and \quad \chi(u\overline{a}) = (\mathsf{rd}(u)as^{-1}, s, s^{-1}\mathsf{wrt}(u))$$

where $s = \mu(u)a \sqcap \mathsf{wrt}(u)$.

Iterating Lemma 3.5 we obtain the following Theorem:

▸ **Theorem 3.6** ([13, Theorem 5.3]). *Let $u, v \in \Sigma^*$. Then $\chi(uv) = (\mathsf{rd}(uv)s^{-1}, s, s^{-1}\mathsf{wrt}(uv))$ where $s = \mu(u)\mathsf{rd}(v) \sqcap \mathsf{wrt}(u)\mu(v)$.* ◂

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 action. 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 $a\overline{b}$ 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 (this is $\mu(u)\mathsf{rd}(v) \sqcap \mathsf{wrt}(u)\mu(v)$).

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 3.7.** 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 [11]. 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 [25, 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 3.8** ([20, 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 3.9.** *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 ε.*
(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:

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

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 [13, Proposition 3.4] we know $u \equiv v$ iff $\delta(u) \equiv \delta(v)$. Hence, δ is an anti-morphism 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 [13, 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 3.6 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 3.10.** $\mathfrak{G}_n$ *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. 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. ◂
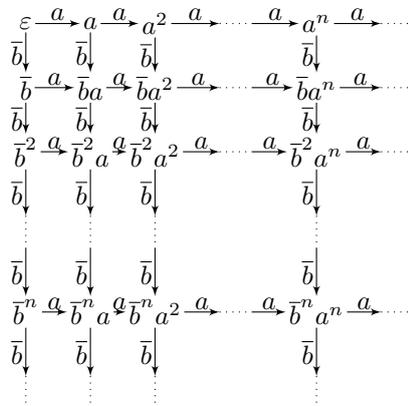
Figure 1 $\mathfrak{C}$ restricted to the nodes reachable by a- and $\overline{b}$-edges, only.

With the help of a famous result from Seese (cf. [28]), we may now prove the undecidability of the monadic second-order theory of the queue monoid's Cayley-graph.

▸ **Corollary 3.11.** $\mathsf{MSOTh}(\mathfrak{C})$ *is undecidable.*

Proof. Due to [26] 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 3.10, each planar graph is minor of an induced subgraph of $\mathsf{ud}(\mathfrak{C})$. Hence, by [28, Theorem 5] $\mathsf{MSOTh}(\mathsf{ud}(\mathfrak{C}))$ is undecidable. Since $\mathsf{ud}(\mathfrak{C})$ is first-order interpretable in $\mathfrak{C}$, $\mathsf{MSOTh}(\mathfrak{C})$ is undecidable as well. ◂

4 Combinatorics on Words

Before diving into the proof of the Cayley-graph's first-order theory we have to prove some combinatorial statements concerning words.

Let $\mathrm{pref}_r(u)$ denote the maximal prefix of u of length at most r. 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. In terms of $\mathfrak{C}$'s first-order theory we only have to consider words $u \in \Sigma^*$ having "short" $\lambda(u)$ and $\varrho(u)$.

▸ **Lemma 4.1.** *Let* $r \in \mathbb{N}$ *and* $u, v, w \in A^*$ *with* $uw \sqcap wv = w$. *Then there are words* u', v' *of length* $\leqslant 2r$ *such that*
- $\mathrm{suf}_r(uw) = \mathrm{suf}_r(u'w)$,
- $\mathrm{suf}_r(wv) = \mathrm{suf}_r(wv')$,
- $\mathrm{pref}_r(wv) = \mathrm{pref}_r(wv')$, *and*
- $u'w \sqcap wv' = w$.

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. ◂

▸ **Remark.** The condition $uw \sqcap wv = w$ in Lemma 4.1 cannot be simplified to $u \sqcap v = \varepsilon$. For example, let $u = v = a$ and $w = baa$. Then only the first equation is satisfied.

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 $u \neq \varepsilon$ 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*, $v \bmod u$, 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 $v = \sqrt{v}^{\,\mathrm{lexp}(\sqrt{v},v)}(v \bmod \sqrt{v})$ 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 $v, w \in A^*$ let $v \Delta w = (y, z)$, where y, z are minimal such that there exists an x with $v = xy$ and $w = xz$. For $\vec{v}, \vec{w} \in (A^*)^k$ let $\vec{v} \Delta \vec{w} = (v_1 \Delta w_1, \ldots, v_k \Delta w_k) \in ((A^*)^2)^k$ and $|\vec{w}| := \sum_{i=1}^{k} |w_i|$.

▸ **Definition 4.2.** Let $u \in A^*$ be a word. A word $v \in A^*$ is a *border* of u (denoted by $v \underset{\sqsubseteq}{\preceq} u$) if $v \preceq u$ and $v \sqsubseteq u$. 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 $u_i \underset{\sqsubseteq}{\preceq} u_{i+1}$. 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 $u_i \underset{\subsetneq}{\precneqq} v \underset{\subsetneq}{\precneqq} u_{i+1}$.

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

Let $u \in \Sigma^*$ be any element from the $\mathfrak{C}$ and $(u_0, \ldots, u_n)$ be the complete border-decomposition of $\mathsf{rd}(u) \sqcap \mathsf{wrt}(u)$. Then the characteristics $(\mathsf{rd}(u)u_i^{-1}, u_i, u_i^{-1}\mathsf{wrt}(u))$ describe all the words having the same projections to write and read actions, resp., as u. In the decidability proof of $\mathsf{FOTh}(\mathfrak{C})$ we consider these words since these are all close to each other in $\mathfrak{C}$.

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 4.4.** 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 $w_i \leq w$.
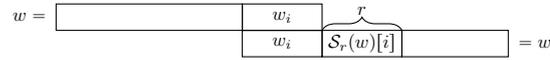
■ **Figure 2** Definition of $\mathcal{S}_r(w)$.

Note that it is convenient for our purpose to consider $\mathcal{S}_r(w)$ to be a word over an alphabet, which in itself consists of words of bounded length rather than to consider $\mathcal{S}_r(w)$ as a sequence of words.

▸ **Example 4.5.** Let $u = babab a$ 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.

$$ab \longrightarrow ba \longrightarrow ba$$

Skeletons will play a crucial role in Section 5. We will prove the decidability of the Cayley-graph of a queue-monoid by translating back and forth between an Ehrenfeucht-Fraïssé game played on the Cayley-graph (presented as EF-relations) and games played on certain skeletons which are derived from the game played on the Cayley-graph.

▸ **Lemma 4.6.** *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 $\mathcal{S}_r(w) = \mathcal{S}_r(v)$.*

Proof. Let $\vec{w} = (w_0, \ldots, w_n)$ 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 $x \sqsubseteq \mathcal{S}_r(w)[0]b^{2n+r}$ 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 $(v_0, \ldots, v_n)$:
(a) For each $0 \leqslant i \leqslant n$ $\sqrt{v_{i+1}} = v_i x_i$ and
(b) $\vec{v} = (v_0, \ldots, v_n)$ 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. 3a), 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. 3b), 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}.

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

(b) Case $\left\lfloor \frac{|p|}{n+r} \right\rfloor$ is even
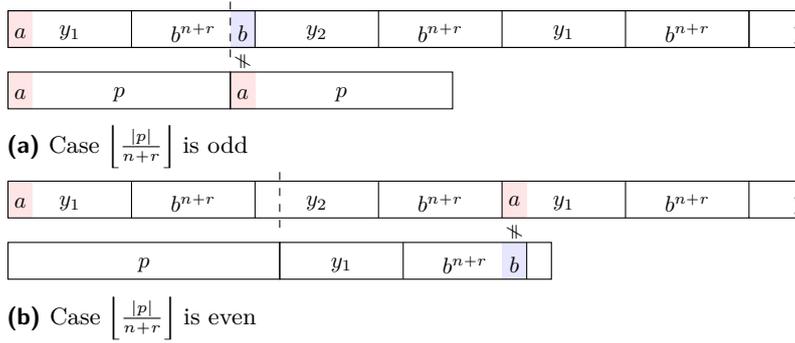
Figure 3

Now, assume $|p|$ is a divisor of $n + r$. Then we can understand the blocks of length $n + r$ as letters of the alphabet $\{b^{n+r}, y_1, \ldots, y_i\}$. 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 $\vec{v} = (v_0, \ldots, v_n)$ is a border-decomposition of $v = v_n$. We prove now by induction on $0 \leqslant i < n$ that $(v_0, \ldots, v_{i+1})$ is a complete border-decomposition of v_i. The case $i = 0$ is easy to verify since $v_1 \in aA^{r-1}b^{2n+r}$. So, let $i \geqslant 1$. Assume there is $u \in A^*$ with $v_i \underset{\neq}{\preceq} u \underset{\neq}{\preceq} v_{i+1}$. 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. 4). Let $x, y \in A^*$ such that $u = xx_i v_i = y$. Then we have $|x| = |y|$ and $m := xx_i y \underset{\neq}{\preceq} u$. 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

$$v_{k-1}x_{k-1}v_{k-1} = v_k = m = xx_iy \,.$$

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.

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 $m \underset{\neq}{\preceq} u$ 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

$$|p| = |v_{i+1}| - |u| < |v_{i+1}| - |v_i| = |v_i x_i| \,.$$
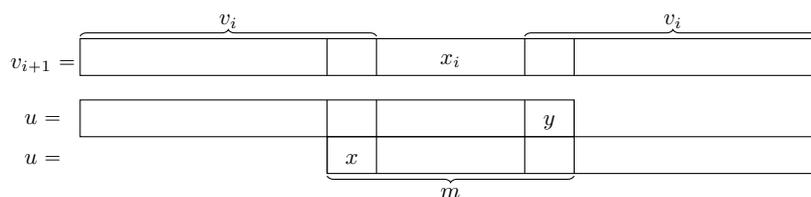
Figure 4

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 $v_i \subsetneqq u \subsetneqq v_{i+1}$, 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 4.6 the *r-instantiation of V*.

5 Decidability of the FO-Theory

Recall that the Cayley-graph of the queue monoid $\mathcal{Q}$ induced by A is denoted by $\mathfrak{C} = (\mathcal{Q}, (E_\alpha)_{\alpha \in \Sigma})$. In order to ease the notation we let elements of $\mathfrak{C}$ inherit some properties from their projections to the read and write actions. For $p, q \in \mathcal{Q}$ let $|p| = |(\mathsf{rd}(p), \mathsf{wrt}(p))|$, $p\Delta q = (\mathsf{rd}(p), \mathsf{wrt}(p))\Delta(\mathsf{rd}(q), \mathsf{wrt}(q))$, and we call $|p\Delta q|$ the *(Δ-)distance* of p and q. Note that Δ defines a metric on $\mathfrak{C}$. Further for $\vec{p} = (p_1, \ldots, p_k) \in \mathcal{Q}^k$ let $\mathcal{N}_r(\vec{p}) = \{q \in \mathcal{Q} \mid \exists 1 \leqslant i \leqslant k \colon |p_i\Delta q| \leqslant r \lor |q| \leqslant r\}$ be the *(Δ-)neighborhood* of $\vec{p}$ of radius r (*r-neighborhood*). Note that we implicitly add the origin of $\mathfrak{C}$ to $\vec{p}$ when we compute the neighborhood. Moreover we define the notion of a border-decomposition and an r-skeleton for an element $p \in \mathcal{Q}$ as the border-decomposition and the r-skeleton of $\mathsf{rd}(p) \sqcap \mathsf{wrt}(p)$.

Let us first give an intuitive outline of our decidability proof. We follow a classical proof strategy due to Ferrante and Rackoff [8]. Roughly speaking we show that there is some fixed primitive recursive function $f : \mathbb{N} \to \mathbb{N}$ such 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 $(\vec{p}, p) \equiv_r (\vec{q}, q)$. This implies that in order to evaluate a formula $Qx\varphi(\vec{p})$ where φ has quantifier rank r and $Q \in \{\exists, \forall\}$ we can restrict the quantification of x to the $f(r+1)$-neighborhood of $\vec{p}$. Since the r-neighborhood of each element $p \in \mathcal{Q}$ is finite and effectively computable for every radius r, we can use the above observation to implement a decision procedure for the theory of $\mathfrak{C}$. In order to achieve this goal we exploit the fact that first-order logic cannot measure distances between two nodes that are more than exponentially far away in the quantifier rank. Therefore our task for a given quantifier rank $r > 0$ is to find for every p that is far away from a tuple $\vec{p}$ an element p' that is closer (but not yet too close) to $\vec{p}$ such that the neighborhoods of p and p' of a suitably chosen radius are not distinguishable with the remaining quantifier rank $r - 1$. What makes this task more complex than for most other examples of Cayley-graphs with decidable first-order theory that can be found in the literature is that the Cayley-graph of the queue monoid is in some sense less local. In fact, the neighborhood-structure of an element p does not only depend on suffixes of bounded

length of $\mathsf{rd}(p)$ and $\mathsf{wrt}(p)$ (as it would be the case for instance for the direct product of two free monoids). We solve this problem via the notion of skeletons. Our proof reveals that the r-type of the 2^{r+1}-neighborhood of an element p is basically determined by the $(r+1)$-type of the $3 \cdot 2^{r+1}$-skeleton of $\mathsf{rd}(p) \sqcap \mathsf{wrt}(p)$. This will be the core of our proof.

Let us start off by making some technical preparations in order to formulate the core idea precisely.

▸ **Definition 5.1.** 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 $\mathsf{rd}(q) \sqcap \mathsf{wrt}(q) = vx$ for some $x \in A^{\leqslant r}$ and $|\mathsf{wrt}(q) \Delta v| \leqslant r$.

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. Next we describe how we translate back and forth between elements of the Cayley-graph and positions in a skeleton. However we can not guarantee that every element in close proximity to a given element p can be associated with a position in the r-skeleton of p because small changes to the read and write actions might change the border-decomposition dramatically. But we can modify r and p slightly to circumvent this problem.

▸ **Definition 5.2.** For $q \in \mathcal{Q}$ with $|\mathsf{rd}(q)| \geqslant r$ let $\mathrm{rc}_r(q)$ be the element q' with $\mathsf{wrt}(q') = \mathsf{wrt}(q)$, $\mathsf{rd}(q') = \mathsf{rd}(q) \, \mathsf{suf}_r(\mathsf{rd}(q))^{-1}$, and $\mu(q') = \mathsf{rd}(q') \sqcap \mathsf{wrt}(q')$. In other words, rc_r just cuts the last r read actions and pushes read and write actions as far together as possible.

▸ **Definition 5.3.** 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 $|p' \Delta p| \leqslant r$ and $|\mu(p')| \geqslant 2r$ 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.

Another important ingredient of our proof is to construct "small" r-equivalent words from a given word w. This is routine since it can be achieved by a simple automata-theoretic approach.

▸ **Lemma 5.4** ([29]). *From a given alphabet Γ, a word $v \in \Gamma^*$, and $r \in \mathbb{N}$ one can compute an automaton $\mathcal{A}$ in time $\exp_{r+1}(f(r))$ with $L(\mathcal{A}) = \{w \in \Gamma^* \mid w \equiv_r v\}$ for some primitive recursive function 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 $\exp_{r+1}(2, f(r))$ where $f(r)$ is an upper bound for the size of the formula φ (which can be chosen to be primitive recursive). ◂

We use this idea to define a family of equivalence relations $(\mathcal{E}_m^r)_{r, m \in \mathbb{N}}$. For $r, m \in \mathbb{N}$ and $\vec{p}, \vec{q} \in \mathcal{Q}^m$ let $\vec{p}\mathcal{E}_m^r\vec{q}$ iff

(1) If $|p_i \Delta \varepsilon| \leqslant 4 \exp_{r+2}(2, f(r))$ then $p_i = q_i$ where f is the function from Lemma 5.4.

(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 with $\min = \min X$:

    (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) $\mathsf{suf}_{2^{r+m+2}}(\mathsf{rd}(p_i)) = \mathsf{suf}_{2^{r+m+2}}(\mathsf{rd}(q_i))$ and
$\mathsf{suf}_{2^{r+m+2}}(\mathsf{wrt}(p_i)) = \mathsf{suf}_{2^{r+m+2}}(\mathsf{wrt}(q_i))$ for all $i \in X$.

    (c) For all $j \in X$ it holds that $|p_{\min} \Delta p_j| \leqslant \sum_{s=r}^{r+m} 2^s$ (and therefore also $|q_{\min} \Delta q_j| \leqslant \sum_{s=r}^{r+m} 2^s$).

    (d) Let U be the $3 \cdot 2^{r+m+1}$-skeleton of $\mathsf{rc}_{2^{r+m+2}}(p_{\min})$ and V be the $3 \cdot 2^{r+m+1}$-skeleton $\mathsf{rc}_{2^{r+m+2}}(q_{\min})$. Then for all $j \in X$ we have that either $\mu(p_j) = \mu(q_j)$ or $|\mu(p_j)| \geqslant 2^{r+m+2}$ and 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 $\{p_j \mid j \in X\}$ and $n_1, \ldots, n_k$ are the positions in U that are associated with $\{q_j \mid j \in X\}$ then $(V, m_1, \ldots, m_k) \equiv_{r+1} (U, n_1, \ldots, n_k)$.

We show that $(\mathcal{E}_m^r)_{r,m \in \mathbb{N}}$ are indeed EF-relations for $\mathfrak{C}$.

▸ **Lemma 5.5.** *For all $m \in \mathbb{N}_{>0}$ and all $\vec{p}, \vec{q} \in \mathcal{Q}^m$: If $\vec{p}\mathcal{E}_m^0\vec{q}$ then the mapping $p_i \mapsto q_i$ is a partial isomorphism.*

Proof. We need to show that $(p_i, p_j) \in E_a \Rightarrow (q_i, q_j) \in E_a$ for all $i, j \leqslant m$ and all $a \in \Sigma$. Let $\vec{p}, \vec{q} \in \mathcal{Q}^m$ with $\vec{p}\mathcal{E}_m^0\vec{q}$. Suppose $(p_i, p_j) \in E_a$ for some $a \in \Sigma$. Then $|p_i \Delta p_j| = 1$. Hence $p_i \Delta p_j = q_i \Delta q_j$ by (2). Let $X_1, \ldots, X_k$ be the partition from Property 3. Since the distance between p_i and p_j and between q_i and q_j is 1 we derive from Property (3a) that i and j belong to the same $X \in \{X_1, \ldots, X_k\}$. Let $\ell = \min X$. If $|\mu(p_i)| < 2^{m+2}$ then, by Property (3d) and (3b), $\mu(p_i) = \mu(q_i)$. In this case $(p_i, p_j) \in E_a \Leftrightarrow (q_i, q_j) \in E_a$ obviously holds. Otherwise there are $3 \cdot 2^{m+1}$-skeletons 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 $\mathsf{rd}(p_i) = \mathsf{rd}(p_j)$, $\mathsf{wrt}(p_i)a = \mathsf{wrt}(p_j)$, and $\mu(p_i) = \mu(p_j)$. In this case $m_1 = m_2$ and therefore $n_1 = n_2$, which implies that $\mathsf{rd}(q_i) = \mathsf{rd}(q_j)$, $\mathsf{wrt}(q_i)a = \mathsf{wrt}(q_j)$, and $\mu(q_i) = \mu(q_j)$. Therefore $(q_i, q_j) \in E_a$.

Second, it might be that $\mathsf{rd}(p_i)a = \mathsf{rd}(p_j)$ (where $a = \bar{b}$), $\mathsf{wrt}(p_i) = \mathsf{wrt}(p_j)$, 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 the formula of quantifier rank 1. Let $(w_0, \ldots, w_n)$ be the complete border-decomposition of $\mathsf{rc}_{2^{m+2}}(p_\ell)$ and $v := w_{m_1}^{-1}\mu(p_i) \in A^{\leqslant 3 \cdot 2^{m+1}}$. Then

$$\varphi(x_1, x_2) \coloneqq x_2 \leqslant x_1 \wedge \bigvee_{s \in A^{\leqslant 3 \cdot 2^{m+1}} : (va) \leqslant s} P_s(x_2) \wedge \forall y : \left(x_2 < y < x_1 \to \bigwedge_{s \in A^{\leqslant 3 \cdot 2^{m+1}} : va \leqslant s} \neg P_s(y) \right).$$

Hence $U \models \varphi(m_1, m_2)$ and since $(U, m_1, m_2) \equiv_1 (V, n_1, n_2)$ also $V \models \varphi(n_1, n_2)$ and therefore $(q_i, q_j) \in E_a$. ◂

▸ **Lemma 5.6.** *For all $m, r \in \mathbb{N}$ and all $\vec{p}, \vec{q} \in \mathcal{Q}^m$:*

$$\vec{p}\mathcal{E}_m^{r+1}\vec{q} \Rightarrow \forall p \in \mathcal{Q} \exists q \in \mathcal{N}_{\exp_{r+3}(g(r+m))}(\vec{q}) : (\vec{p}, p)\mathcal{E}_{m+1}^r(\vec{q}, q)$$

for some primitive recursive function g.

Proof. Let f be the primitive recursive function from Lemma 5.4. Let $\vec{p}, \vec{q} \in \mathcal{Q}^m$ with $(\vec{p}, \vec{q}) \in \mathcal{E}_m^{r+1}$ and let $X_1, \ldots, X_k$ be a partition of $\{1, \ldots, m\}$ with the properties described in (3). 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 $|\bar{\pi}_1(p)| + 2|\mu(p)| + |\varrho(p)|$ it follows that $|\bar{\pi}_1(p)| > \exp_{r+2}(f(r))$ or $|\mu(p)| > \exp_{r+2}(f(r))$ or $|\varrho(p)| > \exp_{r+2}(f(r))$. Let $p' = \mathrm{rc}_{2^{r+m+2}}(p)$. Consider the $3 \cdot 2^{r+m+1}$-skeleton $V = \mathcal{S}_{3 \cdot 2^{r+m+1}}(p')$. By Lemma 5.4 we can find a $3 \cdot 2^{r+m+1}$-skeleton W of length at most $(m+1) \exp_{r+2}(f(r+1))$ with $V \equiv_{r+1} W$ and $3 \cdot 2^{r+m+1}$-instantiation w with $|w| \leqslant c \cdot 2^{(m+1) \exp_{r+2}(f(r+1)) \cdot 3 \cdot 2^{r+m+1}} \leqslant \exp_{r+3}(g(r+m))$ (for a suitable primitive recursive function g). Using Lemma 4.1, words u, v of length at most $(m+1)2^{r+m+3}$ such that

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

such that every element x with $\mathrm{rd}(x) = uw$ and $\mathrm{wrt}(x) = wv$ has distance $> 2^r$ from every q_i. We choose to q to be such an element x. It remains to specify $\mu(x)$. if $|\mu(p)| \leqslant 2^{r+m+2}$ then choose $\mu(q) = \mu(p)$. Otherwise let $(v_0, v_1, \ldots, v_m)$ be the complete border-decomposition of 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 $\mathcal{S}_{3 \cdot 2^{r+m+1}}(p') \equiv_{r+1} W$ there is a $j \in \{0, \ldots, n\}$ such that $(\mathcal{S}_{3 \cdot 2^{r+m+1}}(p'), i) \equiv_r (W, j)$. Now choose $\mu(q) = w_j$. Finally extend the partition by $X_{k+1} = \{m+1\}$.

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 $3 \cdot 2^{r+m+1}$-skeleton of $\mathrm{rc}_{2^{r+m+2}}(p_j)$ and V be the $3 \cdot 2^{r+m+1}$-skeleton of $\mathrm{rc}_{2^{r+m+2}}(q_j)$. Since $|p_i \Delta p_j| \leqslant \sum_{s=r+1}^{r+m} 2^s$ and $|p \Delta p_i| \leqslant 2^r$ we conclude that $|p \Delta p_j| \leqslant \sum_{s=r}^{r+m} 2^s \leqslant 2^{r+m+1}$. Hence, 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 (3d) 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 $(\sum_{s=r+m}^{r} 2^s)$-neighborhood of q_j. The construction of q ensures that Properties (3b)
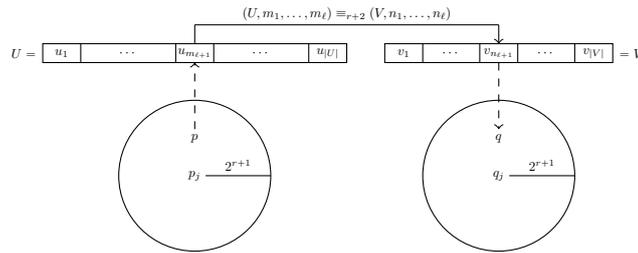
Figure 5 Construction of q from p using U and V.

to (3) are fulfilled for $(\vec{p}, p)$ and $(\vec{q}, q)$ by adding $\ell + 1$ to Y. Hence $(\vec{p}, p) \mathcal{E}_m^r (\vec{q}, q)$. ◀

The Lemmata 5.5 and 5.6 ensure that $\mathcal{E}_m^r$-equivalent tuples are also r-equivalent.

▶ **Corollary 5.7.** *For all $\vec{p} \in \mathcal{Q}^m$, $p \in \mathcal{Q}$, and $r \in \mathbb{N}$ there exists an element $q \in \mathcal{N}_{\exp_{r+3}(g(r+m))}(\vec{p})$ with $(\mathfrak{C}, \vec{p}, p) \equiv_r (\mathfrak{C}, \vec{p}, q)$ for some polynomial f.*

▶ **Lemma 5.8.** *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 $p\Delta q = (u, v, w, x) \in (A^{\leq r})^4$ and $\mu(q)$. Once we have fixed $p\Delta q \in (A^{\leq r})^4$ (and therefore fixed $\mathsf{rd}(q)$ and $\mathsf{wrt}(q)$) there are at most $\min\{|\mathsf{rd}(q)|, |\mathsf{wrt}(q)|\} \leq \min\{|\mathsf{rd}(p)|, |\mathsf{wrt}(p)|\} + r$ possible values for $\mu(q)$. ◄

With this lemma we obtain our main result.

▶ **Theorem 5.9.** $\mathsf{FOTh}(\mathfrak{C})$ *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 5.7. We see that the values $|\mathsf{rd}(p)|$ and $|\mathsf{wrt}(p)|$ are bounded by $\exp_{r+3}(g(r + m))$ Hence, by Lemma 5.8 the algorithm needs to consider at most $|A|^{4r}(\exp_{r+3}(g(r + m)) + 1)$ many Elements, which leads to a runtime of $|\varphi| \cdot (|A|^{4r}(\exp_{r+3}(g(r + m)) + 1))^r$, which is obviously a primitive recursive function. ◄

6 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 [13].

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 [10]	PSPACE [10]
counter	$(\mathbb{Z}, +)$	2EXPSPACE [21]	decidable [20]
stack	polycyclic monoid	2EXPSPACE [21]	decidable [6, 20]
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 a 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 Khoussainov and Nerode [16] nor automatic in the sense of Thurston et al. [4] due to [13], 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. [17, 18]) 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, 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*, volume 2747 of *Lecture Notes in Computer Science*, pages 378–387. Springer, 2003. `doi:10.1007/978-3-540-45138-9_32`.

7 Andrzej Ehrenfeucht. An application of games to the completeness problem for formalized theories. *Fundamenta Mathematicae*, 49(129-141):13, 1961.

8 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`.

9 Roland Fraïssé. Sur quelques classifications des systèmes de relations. *Université d'Alger, Publications Scientifiques, Série A*, 1:35–182, 1954.

10 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, 2007. `doi:10.1007/3-540-68804-8_3`.

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

12 Christoph Haase, Sylvain Schmitz, and Philippe Schnoebelen. The power of priority channel systems. *Logical Methods in Computer Science*, 10(4:4), 2014. `doi:10.2168/LMCS-10(4:4)2014`.

13 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`.

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

15 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`.

16 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`.

17 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`.

18 Chris Köcher, Dietrich Kuske, and Olena Prianychnykova. The Inclusion Structure of Partially Lossy Queue Monoids and their Trace Submonoids. *RAIRO - Theoretical Informatics and Applications*, 52(1):55–86, 2018. `doi:10.1051/ita/2018003`.

19 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`.

20 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`.

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

22   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`.

23   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`.

24   Paliath Narendran and Friedrich Otto. Some results on equational unification. In *10th International Conference on Automated Deduction*, volume 449 of *Lecture Notes in Computer Science*, pages 276–291. Springer, 1990. `doi:10.1007/3-540-52885-7_94`.

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

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

27   Jacques Sakarovitch. Kleene's theorem revisited. In *Trends, Techniques, and Problems in Theoretical Computer Science*, volume 281 of *Lecture Notes in Computer Science*, pages 39–50. Springer, 1986. `doi:10.1007/3540185356_29`.

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

29   Wolfgang Thomas. Languages, Automata, and Logic. In Grzegorz Rozenberg and Arto Salomaa, editors, *Beyond Words*, volume 3 of *Handbook of Formal Languages*, pages 389–455. Springer, 1997. `doi:10.1007/978-3-642-59126-6_7`.