The transformation monoid of a partially lossy queue

Chris Köcher⋆ and Dietrich Kuske⋆

Technische Universität Ilmenau, Institut für Theoretische Informatik

Abstract. We model the behavior of a lossy fifo-queue as a monoid of transformations that are induced by sequences of writing and reading. To have a common model for reliable and lossy queues, we split the alphabet of the queue into two parts: the forgettable letters and the letters that are transmitted reliably.
We describe this monoid by means of a confluent and terminating semi-Thue system and then study some of the monoids algebraic properties. In particular, we characterize completely when one such monoid can be embedded into another as well as which trace monoids occur as submonoids. The resulting picture is far more diverse than in the case of reliable queues studied before.

1 Introduction

Queues (alternatively: fifo queues or channels) form a basic storage mechanism that allows to append items at the end and to read the left-most item from the queue. Providing a finite state automaton with access to a queue results in a Turing complete computation model [2] such that virtually all decision problems on such devices become undecidable.

Situation changes to the better if one replaces the reliable queue by some forgetful version. The most studied version are lossy queues that can nondeterministically lose any item at any moment [7,3,1,13]: in that case reachability, safety properties over traces, inevitability properties over states, and fair termination are decidable (although of prohibitive complexity, see, e.g., [4]). A practically more realistic version are priority queues where items of high priority can erase any previous item of low priority. If items of priority i can be erased by subsequent items of priority *at least* i, then safety and inevitability properties are decidable, if items of priority i can be erased by subsequent items of priority *strictly larger than* i, only, then these problems become undecidable [8].

In this paper, we study partially lossy queues that can be understood as a model between lossy and priority queues. Seen as a version of lossy queues, their alphabet is divided into two sets of reliable and forgettable letters where only items from the second set can be lost. Seen as a version of priority queues, partially lossy queues use only two priorities (0 and 1) where items of priority 0 can be erased by any item of priority at least 0 (i.e., by all items) and items of

⋆ Supported by the DFG-Project “Speichermechanismen als Monoide”, KU 1107/9-1.

priority 1 can only be erased by items of strictly larger priority (which do not exist).

We describe the behavior of such a partially lossy queue by a monoid as was done, e.g., for pushdowns in [10] and for reliable queues in [9,12]: A partially lossy queue is given by its alphabet A as well as the subset $X \subseteq A$ of letters that the queue will transmit reliably. Note that writing a symbol into a queue is always possible (resulting in a longer queue), but reading a symbol is possible only if the symbol is at the beginning of the queue (or is preceded by forgettable symbols, only). Thus, basic actions define partial functions on the possible queue contents. The generated transformation monoid is called *partially lossy queue monoid* or *plq monoid* $\mathcal{Q}(A, X)$. Then $\mathcal{Q}(A, A)$ models the behaviour of a reliable queue with alphabet A [9,12] and $\mathcal{Q}(A, \emptyset)$ the fully lossy queue that can forget any symbol [11].

The first part of this paper presents a complete infinite semi-Thue system for the monoid $\mathcal{Q}(A, X)$. The resulting normal forms imply that two sequences of actions are equivalent if their subsequences of write and of read actions, respectively, coincide and if the induced transformations agree on the shortest queue that they are defined on.

This result is rather similar, although technically more involved, than the corresponding result on the monoid $\mathcal{Q}(A, A)$ of the reliable queue from [9]. In that paper, it is also shown that $\mathcal{Q}(A, A)$ embeds into $\mathcal{Q}(B, B)$ provided B is not a singleton. This is an algebraic formulation of the wellknown fact that the reliable queue with two symbols can simulate any other reliable queue. The second part of the current paper is concerned with the embeddability relation between the monoids $\mathcal{Q}(A, X)$. Clearly, the monoid $\mathcal{Q}(A, \emptyset)$ of the fully lossy queue embeds into $\mathcal{Q}(B, \emptyset)$ whenever $|A| \leq |B|$ by looking at A as a subset of B. Joining this almost trivial idea with the (nontrivial) idea from [9], one obtains an embedding of $\mathcal{Q}(A, X)$ into $\mathcal{Q}(B, Y)$ provided the second queue has at least as many forgettable letters as the first and its number of unforgettable letters is at least the number of unforgettable letters of the first queue or at least two (i.e., $|A \setminus X| \leq |B \setminus Y|$ and $\mathsf{min}\{|X|, 2\} \leq |Y|$). We prove that, besides these cases, an embedding exists only in case the second queue has precisely one non-forgettable letter and properly more forgettable letters than the first queue (i.e., $|Y| = 1$ and $|A \setminus X| < |B \setminus Y|$). As for the reliable queue, this algebraically mirrors the intuion that a partially lossy queue can simulate another partially lossy queue in these cases, only. In particular, a reliable queue does not simulate a fully lossy queue and vice versa and a fully lossy queue cannot simulate another fully lossy queue with more (forgettable) letters.

These results show that the class of submonoids of a plq monoid $\mathcal{Q}(A, X)$ depends heavily on the number of forgettable and non-forgettable letters. In [9], it is shown that the direct product of two free monoids embeds into the monoid of the reliable queue $\mathcal{Q}(A, A)$ (with $|A| \geq 2$). The paper [12] elaborates on this and characterizes the class of trace monoids $\mathbb{M}(\Gamma, I)$ [6] that embed into $\mathcal{Q}(A, A)$. In particular, it shows that $\mathbb{N}^3$ is not a submonoid of $\mathcal{Q}(A, A)$. The final section of this paper studies this question for plq monoids. The – at least for the authors

– surprising answer is that, provided the queue has at least one non-forgettable or at least three forgettable letters, a trace monoid embeds into $\mathcal{Q}(A,X)$ if and only if it embeds into $\mathcal{Q}(A,A)$. By [12], this is the case if all letters in the independence alphabet (Γ,I) have degree at most 1 or the independence alphabet is a complete bipartite graph with some additional isolated vertices. We provide a similar characterization for trace monoids embedding into $\mathcal{Q}(\{a,b\},\emptyset)$: here, the complete bipartite component is replaced by a star graph. In any case, the direct product of $(\mathbb{N},+)$ and $\{a,b\}^*$ embeds into $\mathcal{Q}(A,X)$. Since in this direct product, the inclusion problem for rational sets is undecidable (cf. [15]), the same applies to $\mathcal{Q}(A,X)$.

In summary, we study properties of the transformation monoid of a partially lossy queue that were studied for the reliable queue in [9,12]. We find expected similarities (semi-Thue system), differences (embeddability relation) and surprising similarities (trace submonoids).

2 Preliminaries

At first we need some basic definitions. So let A be an alphabet. A word $u \in A^*$ is a *prefix* of $v \in A^*$ iff $v \in uA^*$. Similarly, u is a *suffix* of v iff $v \in A^*u$. Furthermore u is a *subword* of v iff there are $k \in \mathbb{N}$, $a_1,\ldots,a_k \in A$ and $w_1,\ldots,w_{k+1} \in A^*$ such that $u = a_1\ldots a_k$ and $v = w_1a_1w_2a_2\ldots w_ka_kw_{k+1}$, i.e., we obtain u if we drop some letters from v. In this case we write $u \preceq v$. Note that $\preceq$ is a partial ordering on A^*. Let $X \subseteq A$. Then we define the *projection* $\pi_X\colon A^* \to X^*$ on X by

$$\pi_X(\varepsilon) = \varepsilon \text{ and } \pi_X(au) = \begin{cases} a\pi_X(u) & \text{if } a \in X \\ \pi_X(u) & \text{otherwise} \end{cases}$$

for each $a \in A$ and $u \in A^*$. Moreover, u is an X*-subword* of v (denoted $u \preceq_X v$) if $\pi_X(v) \preceq u \preceq v$, i.e., if we obtain u from v by dropping some letters not in X. Note that $\preceq_\emptyset$ is the subword relation $\preceq$ and $\preceq_A$ is the equality relation.

2.1 Definition of the Monoid

We want to model the behaviour of an unreliable queue that stores entries from the alphabet A. The unreliability of the queue stems from the fact that it can forget certain letters that we collect in the set $A\setminus X$. In other words, letters from $X \subseteq A$ are *non-forgettable* and those from $A \setminus X$ are *forgettable*. Note that this unreliability extends the approach from [9] where we considered reliable queues (i.e., $A = X$).

So let A be an alphabet of possible queue entries and let $X \subseteq A$ be the set of non-forgettable letters. The states of the queue are the words from A^*. Furthermore we have some basic controllable actions on these queues: writing of a symbol $a \in A$ (denoted by a) and reading of $a \in A$ (denoted by $\overline{a}$). Thereby we assume that the set $\overline{A}$ of all these reading operations $\overline{a}$ is a disjoint copy of A. So

$\Sigma := A \cup \overline{A}$ is the set of all controllable operations on the partially lossy queue. For a word $u = a_1 \dots a_n \in A^*$ we write $\overline{u}$ for the word $\overline{a_1}\,\overline{a_2}\,\dots\,\overline{a_n}$.

Formally, the action $a \in A$ appends the letter a to the state of the queue. The action $\overline{a} \in \overline{A}$ tries to cancel the letter a from the beginning of the current state of the queue. If this state does not start with a then the operation $\overline{a}$ is not defined. The lossiness of the queue is modeled by allowing it to forget arbitrary letters from $A \setminus X$ of its content at any moment.

This semantics is similar to the "standard semantics" from [4, Appendix A] where a lossy queue can lose any message at any time. The main part of that paper considers the "write-lossy semantics" where lossiness is modeled by the effect-less writing of messages into the queue. The authors show that these two semantics are equivalent [4, Appendix A] and similar remarks can be made about priority queues [8]. A third possible semantics could be termed "read-lossy semantics" where lossiness is modeled by the loss of any messages that reside in the queue before the one that shall be read. In that case, the queue forgets letters only when necessary and this necessity occurs when one wants to read a letter that is, in the queue, preceeded by some forgettable letters.

In the complete version of this paper, we define both, the "standard semantics" and the "read-lossy semantics" and prove that the resulting transformation monoids are isomorphic; here, we only define the "read-lossy semantics" as this semantics is more convenient for our further considerations.

Definition 2.1. *Let $X \subseteq A$ be two finite sets and $\bot \notin A$. Then the map $\circ_X : (A^* \cup \{\bot\}) \times \Sigma^* \to (A^* \cup \{\bot\})$ is defined for each $q \in A^*$, $a \in A$ and $u \in \Sigma^*$ as follows:*

(i) $q \circ_X \varepsilon = q$

(ii) $q \circ_X au = qa \circ_X u$

(iii) $q \circ_X \overline{a}u = \begin{cases} q' \circ_X u & \text{if } q \in (A \setminus (X \cup \{a\}))^* \, a \, q' \\ \bot & \text{otherwise} \end{cases}$

(iv) $\bot \circ_X u = \bot$

Consider the definition of $q \circ_X \overline{a}u$. There, the word aq' is the smallest suffix of q that contains all the occurrences of the letter a (it follows that the operation $\circ_X$ is welldefined) and the complementary prefix consists of forgettable entries, only. Hence, to apply $\overline{a}$, the queue first "forgets" the prefix and then "delivers" the letter a that is now at the first position.

Lemma 2.2. *Let $q, u \in A^*$ such that $q \circ_X \overline{u} \neq \bot$. Then $q \circ_X \overline{u}$ is the longest suffix of q with $\pi_X(p) \preceq u \preceq p$ where p is the complementary prefix.*

Example 2.3. Let $a \in A \setminus X$, $b \in X$, $q = aabaabba$ and $u = aba$. Then we have $q \circ_X \overline{u} = abaabba \circ_X \overline{ba} = aabba \circ_X \overline{a} = abba$.

On the other hand, the words $aaba$ and $aabaa$ are the only prefixes p' of q with $\pi_X(p') \preceq u \preceq p'$. Their complementary suffixes are $abba$ and bba, the longer one equals $q \circ_X \overline{u}$ as claimed by the lemma.

Two sequences of actions that behave the same on each and every queue will be identified:

Definition 2.4. *Let $X \subseteq A$ be two finite sets and $u, v \in \Sigma^*$. Then u and v act* equally *(denoted by $u \equiv_X v$) if $q \circ_X u = q \circ_X v$ holds for each $q \in A^*$.*

The resulting relation $\equiv_X$ is a congruence on the free monoid Σ^*. Hence, the quotient $\mathcal{Q}(A, X) := \Sigma^*/_{\equiv_X}$ is a monoid which we call partially lossy queue monoid *or* plq monoid *induced by (A, X).*

Example 2.5. Let $a, b \in A$ be distinct. Then we have $\varepsilon \circ_\emptyset ba\overline{a} = ba \circ_\emptyset \overline{a} = \varepsilon$ and $\varepsilon \circ_\emptyset b\overline{a}a = \bot$ implying $ba\overline{a} \not\equiv_\emptyset b\overline{a}a$.

On the other hand, $\varepsilon \circ_A ba\overline{a} = ba \circ_A \overline{a} = \bot = \varepsilon \circ_A b\overline{a}a$. It can be verified that, even more, $q \circ_A ba\overline{a} = q \circ_A b\overline{a}a$ holds for all $q \in A^*$ (since $a \neq b$) implying $ba\overline{a} \equiv_A b\overline{a}a$.

General assumption Suppose $A = \{a\}$ is a singleton. Then $a^{n+1} \circ_X \overline{a} = a^n$ for any $n \geq 0$ (independent of whether $X = A$ or $X = \emptyset$). Hence $\mathcal{Q}(A, A) = \mathcal{Q}(A, \emptyset)$ is the bicyclic semigroup. From now on, we exclude this case and assume $|A| \geq 2$.

2.2 A semi-Thue system for $\mathcal{Q}(A, X)$

Lemma 2.6. *Let $a, b \in A$, $x \in X$ and $w \in A^*$. Then the following hold:*

(i) $b\overline{a} \equiv_X \overline{a}b$ if $a \neq b$ *(iii)* $xwa\overline{a} \equiv_X xw\overline{a}a$
(ii) $a\overline{ab} \equiv_X \overline{a}a\overline{b}$ *(iv)* $awa\overline{a} \equiv_X aw\overline{a}a$

At first we take a look at equations (i)-(iii) (with $|w|_a = 0$ for simplicity). In order for a queue $q \in A^*$ to be defined after execution of the actions, the letter a must already be contained in q preceded by forgettable letters only. Since, in all cases, $\overline{a}$ is the first read operation, $\overline{a}$ reads this occurrence of a from q. Hence it does not matter whether we write b (a, resp.) before or after this reading of a. In equation (iv) we are in the same situation after execution of the leading write operation a. Therefore we can commute the read and write operations in all these situations.

In case of $X = A$, (iv) is a special case of (iii). Furthermore (i), (ii), and (iii) with $w = \varepsilon$ are exactly the equations that hold in $\mathcal{Q}(A, A)$ by [9, Lemma 3.5].

Ordering the equations from Lemma 2.6, the semi-Thue system $\mathcal{R}_X$ consists of the following rules for $a, b \in A$, $x \in X$ and $w \in A^*$:

(a) $b\overline{a} \to \overline{a}b$ if $a \neq b$ (c) $xwa\overline{a} \to xw\overline{a}a$
(b) $a\overline{ab} \to \overline{a}a\overline{b}$ (d) $awa\overline{a} \to aw\overline{a}a$

Since all the rules are length-preserving and move letters from $\overline{A}$ to the left, this semi-Thue system is terminating. Since it is also locally confluent, it is confluent. Hence for any word $u \in \Sigma^*$, there is a unique irreducible word $\mathsf{nf}_X(u)$ with $u \to^* \mathsf{nf}_X(u)$, the *normal form* of u. Let NF_X denote the set of words in normal form.

Proposition 2.7. *Let $u, v \in \Sigma^*$. Then $u \equiv_X v$ if, and only if, $\mathsf{nf}_X(u) = \mathsf{nf}_X(v)$.*

Recall that $a\overline{ab} \equiv_X \overline{a}a\overline{b}$ and $a\overline{a} \not\equiv_X \overline{a}a$, i.e., in general, we cannot cancel in the monoid $\mathcal{Q}(A,X)$. Since rules from $\mathcal{R}_X$ move letters from $\overline{A}$ to the left, we obtain the following restricted cancellation property.

Corollary 2.8. *Let $u, v \in \Sigma^*$ and $x, y \in A^*$ with $\overline{x}uy \equiv_X \overline{x}vy$. Then $u \equiv_X v$.*

To describe the shape of words from NF_X we use a special shuffle operation on two words $u, v \in A^*$: Each symbol $\overline{a}$ of $\overline{v}$ is placed directly behind the first occurrence of a such that we preserve the relative order of symbols in $\overline{v}$ and such that there is no symbol from X between the preceding reading symbol and $a\overline{a}$.

Example 2.9. Let $a, b \in A$ with $a \neq b$ and $q = aabb\overline{ab}$. If $a \notin X$ then we have

$$aabb\overline{ab} \to aab\overline{a}b\overline{b} \to aa\overline{a}bb\overline{b} \to a\overline{a}abb\overline{b} \to a\overline{a}ab\overline{b}b$$

and therefore $a\overline{a}ab\overline{b}b = \mathsf{nf}_X(aabb\overline{ab})$. Otherwise, i.e., if $a \in X$, we can apply rule (c) to $a\overline{a}ab\overline{b}b$ and hence obtain $\mathsf{nf}_X(aabb\overline{ab}) = \overline{ab}aabb$.

The "special shuffle" alluded to above in these cases is $\langle\!\langle aabb, \overline{ab} \rangle\!\rangle = a\overline{a}ab\overline{b}b$ if $a \notin X$ and $\langle\!\langle aabb, \overline{ab} \rangle\!\rangle = \overline{ab}aabb$ otherwise.

The inductive definition of the special shuffle looks as follows:

Definition 2.10. *Let $u, v \in A^*$ and $a \in A$. Then we set*

$$\langle\!\langle u, \overline{\varepsilon} \rangle\!\rangle := u$$

$$\langle\!\langle u, \overline{av} \rangle\!\rangle := \begin{cases} u_1 a\overline{a} \langle\!\langle u_2, \overline{v} \rangle\!\rangle & \text{if } u = u_1 a u_2 \text{ where } u_1 \in (A \setminus (X \cup \{a\}))^*, u_2 \in A^* \\ \text{undefined} & \text{otherwise.} \end{cases}$$

By induction on the length of the word v, one obtains that $\langle\!\langle u, \overline{v} \rangle\!\rangle$ is defined if, and only if, u has a prefix u' with $v \preceq_X u'$. We denote this property by $v \leq_X u$ and call v an *X-prefix of u*. Clearly, the binary relation $\leq_X$ is a partial order. Note that $\leq_\emptyset$ is the subword relation $\preceq$ and $\leq_A$ is the prefix relation on A^*.

Definition 2.11. *The projections $\pi, \overline{\pi} \colon \Sigma^* \to A^*$ on write and read operations are defined for any $u \in \Sigma^*$ by $\pi(u) = \pi_A(u)$ and $\overline{\overline{\pi}(u)} = \pi_{\overline{A}}(u)$.*

In a nutshell, the projection π deletes all letters from $\overline{A}$ from a word. Dually, the projection $\overline{\pi}$ deletes all letters from A from a word *and then surpresses the overlines*. For instance $\pi(a\overline{a}b) = ab$ and $\overline{\pi}(a\overline{a}b) = a$.

Remark 2.12. Since a word is in normal form if no rule from the semi-Thue system $\mathcal{R}_X$ can be applied to it, we get

$$\mathsf{NF}_X = \{\overline{u}\langle\!\langle v, \overline{w} \rangle\!\rangle \mid u, v, w \in A^*, v \leq_X w\} = \overline{A}^* \left(\bigcup_{a \in A} (A \setminus (X \cup \{a\}))^* a\overline{a} \right)^* .$$

Thus, for $u \in \Sigma^*$, there are unique words $u_1, u_2, u_3 \in A^*$ with $\mathsf{nf}_X(u) = \overline{u_1}\langle\!\langle u_2, \overline{u_3} \rangle\!\rangle$; we set $\overline{\pi}_1(u) = u_1$ and $\overline{\pi}_2(u) = u_3$. As a consequence, we get $\mathsf{nf}_X(u) = \overline{\overline{\pi}_1(u)} \left\langle\!\!\left\langle \pi(u), \overline{\overline{\pi}_2(u)} \right\rangle\!\!\right\rangle$.

While $\overline{\pi}_1(u)$ is defined using the semi-Thue system $\mathcal{R}_X$, it also has a natural meaning in terms of the function $\circ_X$: $\overline{\pi}_1(u) \circ_X u$ is defined and, if $q \circ_X u$ is defined, then $|\overline{\pi}_1(u)| \leq |q|$. Hence $\overline{\pi}_1(u)$ is the shortest queue such that execution of u does not end up in the error state.

Example 2.13. Recall Example 2.9. In case of $a \notin X$ we have $\overline{\pi}_1(q) = \varepsilon$ and $\overline{\pi}_2(q) = ab$. Otherwise we have $\overline{\pi}_1(q) = ab$ and $\overline{\pi}_2(q) = \varepsilon$.

For words $u, v \in A^*$ with $\mathsf{nf}_X(u\overline{v}) = \overline{w_1}\langle\!\langle w_2, \overline{w_3}\rangle\!\rangle$, we have $w_2 = \pi(u\overline{v}) = u$ and $w_1 w_3 = \overline{\pi}(u\overline{v}) = v$. Hence, to describe the normal form of $u\overline{v}$, we have to determine $w_3 = \overline{\pi}_2(u\overline{v})$ which is accomplished by the following lemma.

Lemma 2.14. *Let $u, v \in A^*$. Then $\overline{\pi}_2(u\overline{v})$ is the longest suffix v' of v that satisfies $v' \leq_X u$, i.e., such that $\langle\!\langle u, \overline{v'}\rangle\!\rangle$ is defined.*

3 Fully Lossy Queues

The main result of this section is Theorem 3.4 that provides a necessary condition on a homomorphism into $\mathcal{Q}(A, X)$ to be injective. We derive this condition by considering first queue monoids where all letters are forgettable, i.e., monoids of the form $\mathcal{Q}(A, \emptyset)$. Note that the relations $\preceq_\emptyset$ and $\leq_\emptyset$ are both equal to the subword relation $\preceq$. Hence we will use this in the following statements.

The first result of this section (Theorem 3.2) describes the normal form of the product of two elements from $\mathcal{Q}(A, \emptyset)$ in terms of their normal forms (Lemma 2.14 solves this problem in case the first factor belongs to $[A^*]$ and the second to $[\overline{A}^*]$ for arbitrary sets $X \subseteq A$.)

Definition 3.1. *Let $u, v \in A^*$. The* overlap *of u and v is the longest suffix $\mathsf{ol}(u, v)$ of v that is a subword of u.*

Assuming $X = \emptyset$ the relation $\leq_X$ equals the subword relation $\preceq$. Hence, in this situation, Lemma 2.14 implies $\overline{\pi}_2(u\overline{v}) = \mathsf{ol}(u, v)$ for any words $u, v \in A^*$.

Recall that Lemma 2.14 describes the shape of $\mathsf{nf}_X(uv)$ for arbitrary X, $u \in A^*$ and $v \in \overline{A}^*$. The following Theorem describes this normal form for $X = \emptyset$, but arbitrary $u, v \in \Sigma^*$.

Theorem 3.2. *Let $X = \emptyset$, $u, v \in \Sigma^*$, and $w = \mathsf{ol}(\pi(u), \overline{\pi}_2(u)\overline{\pi}_1(v))$. Then*

$$\begin{aligned} \overline{\pi}_2(uv) &= \quad\; w\ \overline{\pi}_2(v) \text{ and} \\ \overline{\pi}(u)\overline{\pi}_1(v) &= \overline{\pi}_1(uv)\ w. \end{aligned}$$

We next infer that if u and v agree in their subsequences of read and write operations, respectively, then they can be equated by multiplication with a large power of one of them.

Proposition 3.3. *Let $u, v \in \Sigma^*$ with $\pi(u) = \pi(v)$, $\overline{\pi}(u) = \overline{\pi}(v)$, and $\overline{\pi}_1(u) \in \overline{\pi}_1(v)A^*$. Then there is a number $i \in \mathbb{N}$ with $\mathsf{nf}_\emptyset(u^i v u^i) = \mathsf{nf}_\emptyset(u^i u u^i)$.*

Proof. If there is $i \geq 1$ with $|\pi(v)| \leq |\overline{\pi}_1(u^i)|$, then $\overline{\pi}_2(vu^i) = \overline{\pi}_2(u\,u^i)$ can be derived from Theorem 3.2 by inductively proving a similar statement for powers of u. Otherwise, let $i \geq 1$ such that $|\overline{\pi}_1(u^i)|$ is maximal (this maximum exists since $|\overline{\pi}_1(u^j)| < |\pi(v)|$ for any $j \in \mathbb{N}$). Again by Theorem 3.2, one obtains $\overline{\pi}_2(u^i v) = \overline{\pi}_2(u^i u)$. Hence, in any case, $\overline{\pi}_2(u^i v u^i) = \overline{\pi}_2(u^i u\,u^i)$. Note that $\pi(u^i v u^i) = \pi(u^i u u^i)$ follows from $\pi(u) = \pi(v)$ and similarly for $\overline{\pi}(u^i v u^i) = \overline{\pi}(u^i u u^i)$. Consequently $\mathsf{nf}_\emptyset(u^i v u^i) = \mathsf{nf}_\emptyset(u^i u u^i)$. □

From this proposition, we can infer the announced necessary condition for a homomorphism into $\mathcal{Q}(A, X)$ to be injective. This condition will prove immensely useful in our investigation of submonoids of $\mathcal{Q}(A, X)$ in the following two sections. It states that if the images of x and y under an embedding ϕ perform the same sequences of read and write operations, respectively, then x and y can be equated by putting them into a certain context.

Theorem 3.4. *Let $\mathcal{M}$ be a monoid, $\phi\colon \mathcal{M} \hookrightarrow \mathcal{Q}(A, X)$ an embedding, and $x, y \in \mathcal{M}$ such that $\pi(\phi(x)) = \pi(\phi(y))$ and $\overline{\pi}(\phi(x)) = \overline{\pi}(\phi(y))$.*

Then there is $z \in \mathcal{M}$ with $zxz = zyz$.

Proof. For notational simplicity, let $\phi(x) = [u]$ and $\phi(y) = [v]$.

We can, without loss of generality, assume that $|\overline{\pi}_1(u)| \leq |\overline{\pi}_1(v)|$. Since $\overline{\pi}(u) = \overline{\pi}(v)$, the word $\overline{\pi}_1(u)$ is a prefix of the word $\overline{\pi}_1(v)$. By Proposition 3.3, there is $i \in \mathbb{N}$ such that $\mathsf{nf}_\emptyset(u^i v u^i) = \mathsf{nf}_\emptyset(u^i u u^i)$. As the semi-Thue system $\mathcal{R}_X$ contains all the rules from $\mathcal{R}_\emptyset$ we get $\mathsf{nf}_X(u^i v u^i) = \mathsf{nf}_X(u^i u u^i)$ and therefore $u^i v u^i \equiv_X u^i u u^i$. In other words, $\phi(x^i y x^i) = \phi(x^i x x^i)$. The injectivity of ϕ now implies $x^i y x^i = x^i x x^i$. Setting $z = x^i$ yields $zxz = zyz$ as claimed. □

4 Embeddings between PLQ Monoids

We now characterize when the plq monoid $\mathcal{Q}(A, X)$ embeds into $\mathcal{Q}(B, Y)$.

Theorem 4.1. *Let A, B be alphabets with $|A|, |B| \geq 2$, $X \subseteq A$ and $Y \subseteq B$. Then $\mathcal{Q}(A, X) \hookrightarrow \mathcal{Q}(B, Y)$ holds iff all of the following properties hold:*

- *(A) $|A \setminus X| \leq |B \setminus Y|$, i.e., (B, Y) has at least as many forgettable letters as (A, X).*
- *(B) If $Y = \emptyset$, then also $X = \emptyset$, i.e., if (B, Y) consists of forgettable letters only, then so does (A, X).*
- *(C) If $|Y| = 1$, then $|A \setminus X| < |B \setminus Y|$ or $|X| \leq 1$, i.e., if (B, Y) has exactly one non-forgettable letter and exactly as many forgettable letters as (A, X), then A contains at most one non-forgettable letter.*

In particular, $\mathcal{Q}(A, A)$ embeds into $\mathcal{Q}(B, B)$ whenever $|B| \geq 2$, i.e., this theorem generalizes [9, Corollary 5.4]. We prove it in Propositions 4.2 and 4.5.

4.1 Preorder of Embeddability

The embeddability of monoids is reflexive and transitive, i.e., a preorder. Before diving into the proof of Theorem 4.1, we derive from it an order-theoretic description of this preorder on the class of all plq monoids (see the reflexive and transitive closure of the graph on the right). The plq monoid $\mathcal{Q}(A, X)$ is, up to isomorphism, completely given by the numbers $m = |X|$ and $n = |A \setminus X|$ of unforgettable and of forgettable letters, respectively. Therefore, we describe this preorder in terms of pairs of natural numbers. We write $(m, n) \to (m', n')$ if

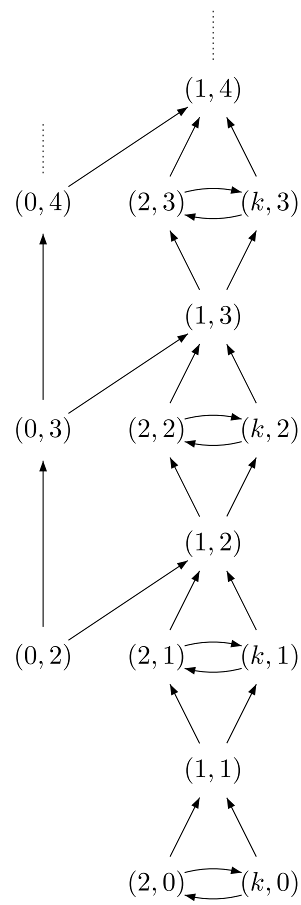
$$\mathcal{Q}([m+n], [m]) \hookrightarrow \mathcal{Q}([m'+n'], [m'])$$

where, as usual, $[n] = \{1, 2, \ldots, n\}$. Then Theorem 4.1 reads as follows: If $m, n, m', n' \in \mathbb{N}$ with $m+n, m'+n' \geq 2$, then $(m, n) \to (m', n')$ iff all of the following properties hold:

(A) $n \leq n'$
(B) If $m' = 0$, then $m = 0$
(C) If $m' = 1$, then $m \leq 1$ or $n < n'$

Then we get immediately for all appropriate natural numbers $m, n, n' \in \mathbb{N}$:

- if $k \geq 2$, then $(2, n) \to (k, n) \to (2, n)$
- $(2, n) \to (2, n')$ iff $n \leq n'$
- $(1, n) \to (2, n')$ iff $n \leq n'$
- $(0, n) \to (2, n')$ iff $n \leq n'$
- $(2, n) \to (1, n')$ iff $n < n'$
- $(1, n) \to (1, n')$ iff $n \leq n'$
- $(0, n) \to (1, n')$ iff $n \leq n'$
- $(2, n) \not\to (0, n')$
- $(1, n) \not\to (0, n')$ iff $n \leq n'$
- $(0, n) \to (0, n')$ iff $n \leq n'$



These facts allow to derive the above graph (where k stands for an arbitrary number at least 3).

First look at the nodes not of the form $(0, n)$. They form an alternating chain of infinite equivalence classes $\{(k, n) \mid k \geq 2\}$ and single nodes $(1, n)$. The infinite equivalence class at the bottom corresponds to the monoids of fully reliable queues considered in [9].

The nodes of the form $(0, n)$ also form a chain of single nodes (these nodes depict the fully lossy queue monoids from [11]). The single node number n (i.e.,

$(0, 2+n)$) from this chain is directly below the single node number $2+n$ (i.e., $(1, 2+n)$) of the alternating chain.

4.2 Sufficiency in Theorem 4.1

Proposition 4.2. *Let A, B be non-singleton alphabets, $X \subseteq A$, $Y \subseteq B$ satisfying Conditions (A)-(C) from Theorem 4.1. Then $\mathcal{Q}(A, X)$ embeds into $\mathcal{Q}(B, Y)$.*

Proof. First suppose $|X| \leq |Y|$. By Condition (A), we can assume $A \setminus X \subseteq B \setminus Y$ and $X \subseteq Y$. Then Proposition 2.7 implies that $\mathcal{Q}(A, X)$ is a submonoid of $\mathcal{Q}(B, Y)$ since the rules of the semi-Thue system only permute letters in words.

Now assume $|X| > |Y|$. By Condition (A), there exists an injective mapping $\phi_1 \colon A \setminus X \hookrightarrow B \setminus Y$. Since $|X| > |Y|$, Condition (B) implies $Y \neq \emptyset$. Let $b_1 \in Y$ be arbitrary. If $|Y| > 1$, then choose $b_2 \in Y \setminus \{b_1\}$. Otherwise, we have $1 = |Y| < |X|$. Hence, by Condition (C), the mapping ϕ_1 is not surjective. So we can choose $b_2 \in B \setminus (Y \cup \{\phi_1(a) \mid a \in A \setminus X\})$. With $X = \{x_1, x_2, \ldots, x_n\}$, we set (for $a \in A$)

$$\phi'(a) = \begin{cases} \phi_1(a) & \text{if } a \in A \setminus X \\ b_1^{|A|+i} b_2 b_1^{|A|-i} b_2 & \text{if } a = x_i \end{cases} \quad \text{and } \phi'(\overline{a}) = \overline{\phi'(a)}\,.$$

Then ϕ' maps $(A \cup \overline{A})^*$, A^*, and $\overline{A}^*$ injectively into $(B \cup \overline{B})^*$, B^*, and $\overline{B}^*$, respectively.

We prove that ϕ' induces an embedding $\phi \colon \mathcal{Q}(A, X) \hookrightarrow \mathcal{Q}(B, Y)$ by $\phi([u]) = [\phi'(u)]$.

First let $u \equiv_X v$ with $u, v \in (A \cup \overline{A})^*$ be any of the equations in Lemma 2.6. In each of the four cases, one obtains $\phi'(u) \equiv_Y \phi'(v)$. Consequently, $u \equiv_X v$ implies $\phi'(u) \equiv_Y \phi'(v)$ for any $u, v \in (A \cup \overline{A})^*$ by Proposition 2.7. Hence ϕ is welldefined.

To prove its injectivity, let $u, v \in (A \cup \overline{A})^*$ with $\phi'(u) \equiv_Y \phi'(v)$.

Set $\mathsf{nf}_X(u) = \overline{u_1} \langle\!\langle u_2, \overline{u_3} \rangle\!\rangle$ and similarly $\mathsf{nf}_X(v) = \overline{v_1} \langle\!\langle v_2, \overline{v_3} \rangle\!\rangle$. The crucial part of the proof demonstrates that ϕ' commutes with the shuffle operation, more precisely, $\phi'(\langle\!\langle u_2, \overline{u_3} \rangle\!\rangle) \equiv_Y \left\langle\!\!\left\langle \phi'(u_2), \overline{\phi'(u_3)} \right\rangle\!\!\right\rangle$ and similarly for v.

Since ϕ' is a homomorphism, we then get

$$\phi'(\overline{u_1}) \left\langle\!\!\left\langle \phi'(u_2), \overline{\phi'(u_3)} \right\rangle\!\!\right\rangle \equiv_Y \phi'(\overline{u_1}\, \langle\!\langle u_2, \overline{u_3} \rangle\!\rangle) \equiv_Y \phi'(u)$$

and similarly $\phi'(v) \equiv_Y \phi'(\overline{v_1}) \left\langle\!\!\left\langle \phi'(v_2), \overline{\phi'(v_3)} \right\rangle\!\!\right\rangle$.

Thus the words $\phi'(\overline{u_1}) \left\langle\!\!\left\langle \phi'(u_2), \overline{\phi'(u_3)} \right\rangle\!\!\right\rangle$ and $\phi'(\overline{v_1}) \left\langle\!\!\left\langle \phi'(v_2), \overline{\phi'(v_3)} \right\rangle\!\!\right\rangle$ in normal form are equivalent and therefore equal. Hence we get

$$\phi'(\overline{u_1}) = \phi'(\overline{v_1})\,, \quad \phi'(u_2) = \phi'(v_2) \text{ and } \phi'(\overline{u_3}) = \phi'(\overline{v_3})\,.$$

Since $\phi' \colon (A \cup \overline{A})^* \to (B \cup \overline{B})^*$ is injective, this implies $u_1 = v_1$, $u_2 = v_2$, and $u_3 = v_3$ and therefore $u \equiv_X \mathsf{nf}(u) = \mathsf{nf}(v) \equiv_X v$. Thus, indeed, ϕ is an embedding of $\mathcal{Q}(A, X)$ into $\mathcal{Q}(B, Y)$. □

4.3 Necessity in Theorem 4.1

Now we have to prove the other implication of the equivalence in Theorem 4.1. Recall the embedding ϕ we constructed in the proof of Proposition 4.2. In particular, it has the following properties:

(1) If $a \in A$, then $\phi(a) \in [B^+]$ and $\phi(\overline{a}) = \overline{\phi(a)}$. In particular, the image of every write operation a performs write operations, only, and the image of every read operation $\overline{a}$ is the "overlined version of the image of the corresponding read operation" and therefore performs read operations, only.
(2) If $a \in A \setminus X$, then $\phi(a) \in B \setminus Y$. In particular, the image of every write operation of a forgettable letter writes only forgettable letters.
(3) If $x \in X$, then $\phi(x) \in [B^*YB^*]$. In particular, the image of every write operation of a non-forgettable letter writes at least one non-forgettable letter.

The proof of the necessity in Theorem 4.1 first shows that any embedding satisfies slightly weaker properties. We start with our weakenings of properties (1) and (2). The first statement of the following lemma is a weakening of (1) since it only says something about the letters in $\phi(a)$ and $\phi(\overline{a})$ but not that these two elements are dual. Similarly the second statement is a weakening of (2) since it does not say anything about the length of $\phi(a)$ but only something about the letters occurring in $\phi(a)$.

Lemma 4.3. *Let A, B be non-singleton alphabets, $X \subseteq A$, $Y \subseteq B$, and ϕ an embedding of $\mathcal{Q}(A, X)$ into $\mathcal{Q}(B, Y)$. Then the following holds:*

(i) $\phi(a) \in [B^+]$ *and* $\phi(\overline{a}) \in [\overline{B}^+]$ *for each* $a \in A$.
(ii) $\phi(a) \in [(B \setminus Y)^*]$ *for each* $a \in A \setminus X$.

Proof. To prove (i), let $a \in A$ and suppose $\phi(a) \notin [B^*]$. One first shows that $\phi(a)$ performs at least one write operation, i.e., $\phi(a) \notin [\overline{B}^*]$. Let $p, q \in B^+$ be the primitive roots of the nonempty words $\pi(\phi(a))$ and $\overline{\pi}(\phi(a))$, respectively.

Since $|A| \geq 2$, there exist distinct letters $a_1, a_2 \in A$. A crucial property of ϕ is that then $\pi(\phi(\overline{a_i})) \in p^*$ and $\overline{\pi}(\phi(\overline{a_i})) \in q^*$. Consequently, $\pi(\phi(\overline{a_1}\,\overline{a_2})) = \pi(\phi(\overline{a_1}))\,\pi(\phi(\overline{a_2})) = \pi(\phi(\overline{a_2}))\,\pi(\phi(\overline{a_1})) = \pi(\phi(\overline{a_2}\,\overline{a_1}))$ (the equality $\overline{\pi}(\phi(\overline{a_1}\,\overline{a_2})) = \overline{\pi}(\phi(\overline{a_2}\,\overline{a_1}))$ follows similarly). Since ϕ is an embedding, Theorem 3.4 implies the existence of $u \in (A \cup \overline{A})^*$ with $u\overline{a_1}\,\overline{a_2}u \equiv_X u\overline{a_2}\,\overline{a_1}u$. It follows that these two words have the same sequence of read operations and therefore in particular $a_1\,a_2 = a_2\,a_1$. But this implies $a_1 = a_2$ which contradicts our choice of these two letters. Hence, indeed, $\phi(a) \in [B^*]$ which proves the first claim from (i), the second follows similarly.

Statement (ii) is shown by contradiction. Let $a \in A \setminus X$ with $\phi(a) \notin [(B \setminus Y)^*]$. Since $|A| \geq 2$, there exists a distinct letter $b \in A \setminus \{a\}$. Using (i) and the assumption on $\phi(a)$, one obtains $\phi(a^n b\overline{b}) = \phi(a^n \overline{b} b)$ with n the length of $\phi(\overline{b})$. Injectivity of ϕ and Proposition 2.7 lead to a contradiction. □

We next come to property (3) that we prove for every embedding.

Lemma 4.4. *Let A, B be non-singleton alphabets, $X \subseteq A$, $Y \subseteq B$, and ϕ an embedding of $\mathcal{Q}(A,X)$ into $\mathcal{Q}(B,Y)$. Then we have $\phi(x) \in [B^*YB^*]$ for each $x \in X$.*

Proof. Let $x \in X$. Since $|A| \geq 2$, there is a distinct letter $a \in A \setminus \{x\}$. By Lemma 4.3, there are words $u, v, w \in B^+$ with $\phi(a) = [u]$, $\phi(\overline{a}) = [\overline{v}]$ and $\phi(x) = [w]$. One then shows $\overline{\pi}_2(wu\overline{v}) = \varepsilon \neq \overline{\pi}_2(u\overline{v})$.

By Lemma 2.14, $v' = \overline{\pi}_2(u\overline{v}) \neq \varepsilon$ is a suffix of v with $v' \leq_X u'$ for some prefix u' of u implying $v' \preceq wu$. Since $\overline{\pi}_2(wu\overline{v}) = \varepsilon$, Lemma 2.14 implies $\pi_Y(wu') \neq \pi_Y(v') = \pi_Y(u')$, i.e., w contains some letter from Y. ☐

Finally we obtain the remaining implication in Theorem 4.1.

Proposition 4.5. *Let A and B be non-singleton alphabets, $X \subseteq A$ and $Y \subseteq B$ such that $\mathcal{Q}(A,X) \hookrightarrow \mathcal{Q}(B,Y)$. Then the Conditions (A)-(C) from Theorem 4.1 hold.*

Proof. First suppose $X \neq \emptyset$. Then, $Y \neq \emptyset$ by Lemma 4.4, i.e., we have (B).

Condition (A) is trivial if $A \setminus X = \emptyset$. If $A \setminus X$ is a singleton, then Lemma 4.3(ii) implies $B \setminus Y \neq \emptyset$ and therefore $|A \setminus X| \leq |B \setminus Y|$. So it remains to consider the case that $A \setminus X$ contains at least two elements. One then shows that the last letters of the words $\overline{\pi}(\phi(\overline{a}))$ for $a \in A \setminus X$ are mutually distinct.

To prove Condition (C), suppose $Y = \{y\}$ and $|A \setminus X| = |B \setminus Y|$. One then proves $|X| \leq 1$ by considering the last letters of $\overline{\pi}(\phi(\overline{x}))$ for $x \in X$. ☐

5 Embeddings of Trace Monoids

Corollary 5.4 from [9] implies that all reliable queue monoids $\mathcal{Q}(A,A)$ for $|A| \geq 2$ have the same class of submonoids. Our Theorem 4.1 shows that this is not the case for all plq monoids $\mathcal{Q}(A,X)$ (e.g., $\mathcal{Q}(A,A)$ does not embed into $\mathcal{Q}(A,\emptyset)$ and vice versa). This final section demonstrates a surprising similarity among all these monoids, namely the trace monoids contained in them.

These trace (or free partially commutative) monoids are used for modeling concurrent systems where the concurrency is governed by the use of joint resources (cf. [14]). Formally such a system is a so called *independence alphabet*, i.e., a tuple (Γ, I) of a non-empty finite set Γ and a symmetric, irreflexive relation $I \subseteq \Gamma^2$, i.e., (Γ, I) can be thought of as an undirected graph. Given an independence alphabet (Γ, I), we define the relation $\equiv_I \; \subseteq (\Gamma^*)^2$ as the least congruence satisfying $ab \equiv_I ba$ for each $(a,b) \in I$. The induced *trace monoid* is $\mathbb{M}(\Gamma, I) := \Gamma^*/_{\equiv_I}$. See [14,5,6] for further information on trace monoids.

5.1 Large alphabets

Theorem 2.7 from [12] describes when the trace monoid $\mathbb{M}(\Gamma, I)$ embeds into the queue monoid $\mathcal{Q}(A,A)$ for $|A| \geq 2$. The following theorem shows that this is the case if, and only if, it embeds into $\mathcal{Q}(A,X)$ provided $|A| + |X| \geq 3$.

Theorem 5.1. *Let A be an alphabet and $X \subseteq A$ with $|A|+|X| \geq 3$. Furthermore let (Γ, I) be an independence alphabet. Then the following are equivalent:*

(A) $\mathbb{M}(\Gamma, I)$ *embeds into* $\mathcal{Q}(A, X)$.
(B) $\mathbb{M}(\Gamma, I)$ *embeds into* $\mathcal{Q}(A, A)$.
(C) $\mathbb{M}(\Gamma, I)$ *embeds into* $\{a, b\}^* \times \{c, d\}^*$.
(D) One of the following conditions holds:
  (D.a) All nodes in (Γ, I) have degree ≤ 1.
  (D.b) The only non-trivial connected component of (Γ, I) is complete bipartite.

Since $X \subseteq A$, the condition $|A| + |X| \geq 3$ implies in particular $|A| \geq 2$. Hence the equivalence between (B), (C), and (D) follows from [12, Theorem 2.7].

For the implication "(C)⇒(A)", one considers the two cases $|A| \geq 3$ and $|A| = 2$, $X \neq \emptyset$ separately. In the first case, one chooses pairwise distinct $a, b, c \in A$ and sets $\phi(a, \varepsilon) = a$, $\phi(b, \varepsilon) = b$, $\phi(\varepsilon, c) = \overline{ac}$, and $\phi(\varepsilon, d) = \overline{bc}$. In the second case, the embedding is similar to the one from [9, Proposition 8.3] (proving the implication "(C)⇒(B)").

The implication "(A)⇒(D)" is proved under the slightly more general assumption $|A| \geq 2$. It is, by far, more involved. We nevertheless only give an overview here since it follows the proof of the implication "(A)⇒(D)" from [12] rather closely:

Suppose ϕ embeds the trace monoid $\mathbb{M}(\Gamma, I)$ into the plq monoid $\mathcal{Q}(A, X)$ with $|A| \geq 2$. This defines a partition of the independence alphabet into the sets $\Gamma_+ := \{\alpha \in \Gamma \mid \phi(\alpha) \in [A^+]\}$, $\Gamma_- := \{\alpha \in \Gamma \mid \phi(\alpha) \in [\overline{A}^+]\}$, and $\Gamma_\pm := \Gamma \setminus (\Gamma_+ \cup \Gamma_-)$. The crucial steps are then to verify the following properties:

(i) $(\Gamma_+ \cup \Gamma_-, I)$ is complete bipartite with the partitions Γ_+ and Γ_-.
(ii) Let $a \in \Gamma_+ \cup \Gamma_-$ and $b, c \in \Gamma$ with $(b, c) \in I$. Then $(a, b) \in I$ or $(a, c) \in I$.
(iii) Let $a \in \Gamma_\pm$. Then a has degree ≤ 1 in the undirected graph (Γ, I).
(iv) (Γ, I) is P_4-free, i.e., the path on four vertices is no induced subgraph.

The proof of [12, Theorem 4.14] shows that any graph $(\Gamma_+ \uplus \Gamma_- \uplus \Gamma_\pm, I)$ satisfying these graph theoretic properties also satisfies (D.a) or (D.b).

5.2 The binary alphabet

In Theorem 5.1 we have only considered partially lossy queues with $|A| > 2$ or $|X| \neq 0$. For a complete picture, it remains to consider the case $|A| = 2$ and $|X| = 0$. The following theorem implies in particular that $\mathcal{Q}(\{\alpha, \beta\}, \emptyset)$ does not contain the direct product of two free monoids, i.e., it contains properly less trace monoids than $\mathcal{Q}(A, X)$ with $|A| + |X| \geq 3$.

Theorem 5.2. *Let A be an alphabet with $|A| = 2$ and (Γ, I) be an independence alphabet. Then the following are equivalent:*

(A) $\mathbb{M}(\Gamma, I)$ *embeds into* $\mathcal{Q}(A, \emptyset)$.
(B) One of the following conditions holds:
  (B.1) All nodes in (Γ, I) have degree ≤ 1.
  (B.2) The only non-trivial connected component of (Γ, I) is a star graph.

For the proof of the implication "(B)⇒(A)", one provides the embeddings as follows (with $A = \{\alpha, \beta\}$):

(B.1) It suffices to consider the case that (Γ, I) is the disjoint union of the edges (a_i, b_i) for $1 \leq i \leq n$. Then we define $w_i = \alpha^i \beta$ for $1 \leq i \leq n$ and the embedding ϕ is given by $\phi(a_i) = [\overline{w_i} w_i]$ and $\phi(b_i) = [\overline{w_i}\, \overline{w_i} w_i]$.

(B.2) Let c be the center of the star graph, s_i for $1 \leq i \leq m$ its neighbors, and r_i for $1 \leq i \leq n$ the isolated nodes of (Γ, I). Then the embedding ϕ is given by $\phi(c) = [\alpha]$, $\phi(s_i) = [\overline{w_i}]$ and $\phi(r_j) = [\overline{w_j \beta} \beta]$.

Note that these embeddings map letters to sequences containing both, read and write operations.

For the more involved implication "(A)⇒(B)", suppose Γ has a node of degree ≥ 2 and, towards a contradiction, (B) does not hold. Since we proved the implication "(A)⇒(D)" in Theorem 5.1 under the assumption $|A| \geq 2$, we obtain that (Γ, I) has a single nontrivial connected component $C \subseteq \Gamma_+ \cup \Gamma_-$. Furthermore, there are $a, b \in \Gamma_+$ distinct and $c \in \Gamma_-$ such that $(a, c), (c, b) \in I$. Using Lemma 2.14, one arrives at $ab \equiv_I ba$ which contradicts $a \neq b$.

References

1. P. A. Abdulla and B. Jonsson. Verifying programs with unreliable channels. *Information and Computation*, 127(2):91–101, 1996.
2. D. Brand and P. Zafiropulo. On communicating finite-state machines. *Journal of the ACM*, 30(2), 1983.
3. G. Cécé, A. Finkel, and S. Purushotaman Iyer. Unreliable channels are easier to verify than perfect channels. *Information and Computation*, 124(1):20–31, 1996.
4. P. Chambart and P. Schnoebelen. The ordinal recursive complexity of lossy channel systems. In *LICS'08*, pages 205–216. IEEE Computer Society Press, 2008.
5. V. Diekert. *Combinatorics on traces*, volume 454. Springer Science & Business Media, 1990.
6. V. Diekert and G. Rozenberg. *The Book of Traces*. World Scientific, 1995.
7. A. Finkel. Decidability of the termination problem for completely specified protocols. *Distributed Computing*, 7(3):129–135, 1994.
8. C. Haase, S. Schmitz, and P. Schnoebelen. The power of priority channel systems. *Logical Methods in Computer Science*, 10(4:4), 2014.
9. M. Huschenbett, D. Kuske, and G. Zetzsche. The monoid of queue actions. *Semigroup forum*, 2017. To appear.
10. M. Kambites. Formal languages and groups as memory. *Communications in Algebra*, 37(1):193–208, 2009.
11. C. Köcher. Einbettungen in das Transformationsmonoid einer vergesslichen Warteschlange. Master's thesis, TU Ilmenau, 2016.
12. D. Kuske and O. Prianychnykova. The trace monoids in the queue monoid and in the direct product of two free monoids. *arXiv preprint arXiv:1603.07217*, 2016.
13. B. Masson and P. Schnoebelen. On verifying fair lossy channel systems. In *MFCS'02*, Lecture Notes in Comp. Science vol. 2420, pages 543–555. Springer, 2002.
14. A. Mazurkiewicz. Concurrent program schemes and their interpretations. *DAIMI Report Series*, 6(78), 1977.
15. A. Muscholl and H. Petersen. A note on the commutative closure of star-free languages. *Inf. Process. Lett.*, 57(2):71–74, 1996.