Minimal and Reduced Reversible Automata

Giovanna J. Lavado, Giovanni Pighizzini, Luca Prigioniero

Minimal and Reduced Reversible Automata

Giovanna J. Lavado, Giovanni Pighizzini, and Luca Prigioniero

Dipartimento di Informatica, Università degli Studi di Milano, Italy
{lavado,pighizzini}@di.unimi.it, luca.prigioniero@studenti.unimi.it

Abstract. A condition characterizing the class of regular languages which have several nonisomorphic minimal reversible automata is presented. The condition concerns the structure of the minimum automaton accepting the language under consideration. It is also observed that there exist reduced reversible automata which are not minimal, in the sense that all the automata obtained by merging some of their equivalent states are irreversible. Furthermore, it is proved that if the minimum deterministic automaton accepting a reversible language contains a loop in the "irreversible part" then it is always possible to construct infinitely many reduced reversible automata accepting such a language.

1 Introduction

A device is said to be *reversible* when each configuration has exactly one predecessor, thus implying that there is no loss of information during the computation. On the other hand, as observed by Landauer, logical irreversibility is associated with physical irreversibility and implies a certain amount of heat generation [8]. In order to avoid such a power dissipation and, hence, to reduce the overall power consumption of computational devices, the possibility of realizing reversible machines looks appealing.

A lot of work has been done to study reversibility in different computational devices. Just to give a few examples in the case of general devices as Turing machines, Bennet proved that each machine can be simulated by a reversible one [2], while Lange, McKenzie, and Tapp proved that each deterministic machine can be simulated by a reversible machine which uses the same amount of space [9]. As a corollary, in the case of a constant amount of space, this implies that each regular language is accepted by a *reversible two-way deterministic finite automaton*. Actually, this result was already proved by Kondacs and Watrous [5].

However, in the case of *one-way* automata, the situation is different. In fact, as shown by Pin, the regular language a^*b^* cannot be accepted by any reversible automaton [11].[1] So the class of languages accepted by *reversible automata* is a proper subclass of the class of regular languages. Actually, there are some different notions of reversible automata in literature. In 1982, Angluin introduced reversible automata in algorithmic learning theory, considering devices having

---

[1] From now on, we will consider only *one-way automata*. Hence we will omit to specify "one-way" all the times.

only one initial and only one final state [1]. On the other hand, the devices considered in [11], besides a set of final states, can have multiple initial states, hence they can take a nondeterministic decision at the beginning of the computation. An extension which allows to consider nondeterministic transitions, without changing the class of accepted languages, has been considered by Lombardy [10], introducing and investigating *quasi reversible automata*. Classical automata, namely automata with a single initial state and a set of final states, have been considered in the works by Holzer, Jakobi, and Kutrib [6, 3, 7]. In particular, in [3] the authors gave a characterization of regular languages which are accepted by reversible automata. This characterization is given in terms of the structure of the minimum deterministic automaton. Furthermore, they provide an algorithm that, in the case the language is acceptable by a reversible automaton, allows to transform the minimum automaton into an equivalent reversible automaton, which in the worst case is exponentially larger than the given minimum automaton. In spite of that, the resulting automaton is minimal, namely there are no reversible automata accepting the same language with a smaller number of states. However, it is not necessarily unique, in fact there could exist different reversible automata with the same number of states accepting the same language.

In this paper we continue the investigation of minimality in reversible automata. Our first result is a condition that characterizes languages having several different minimal reversible automata. Even this condition is on the structure of the transition graph of the minimum automaton accepting the language under consideration. As a special case, we show that each time the "irreversible part" of the minimum automaton contains a loop, it is possible to construct at least two different minimal reversible automata.

We also observe that there exist reversible automata which are not minimal but they are reduced, in the sense that when we try to merge some of their equivalent states we always obtain an irreversible automaton. Investigating this phenomenon more into details, we were able to find a language for which there exist arbitrary large, and hence infinitely many, reduced reversible automata. In the paper, we present a general construction that allows to obtain arbitrary large reversible automata for each language accepted by a minimum deterministic automaton satisfying the structural condition given in [3] and such that the "irreversible part" contains a loop. We know that this is also possible in other situations, namely that our condition is not necessary. We leave as an open problem, to find a characterization of the class of the languages having infinitely many reduced reversible automata.

2   Preliminaries

In this section we recall some basic definitions and results useful in the paper. We assume the reader is familiar with standard notions from automata and formal language theory (see, e.g., [4]). Given a set S, let us denote by $\#S$ its cardinality

and by 2^S the family of all its subsets. Given an alphabet Σ, $|w|$ denotes the length of a string $w \in \Sigma^*$ and ε the empty string.

A *deterministic finite automaton* (DFA for short) is a tuple $A = (Q, \Sigma, \delta, q_I, F)$, where Q is the finite set of *states*, Σ is the *input alphabet*, $q_I \in Q$ is the *initial state*, $F \subseteq Q$ is the set of *accepting states*, and $\delta : Q \times \Sigma \to Q$ is the partial *transition function*. The *language accepted* by A is $L(A) = \{w \in \Sigma^* \mid \delta(q_I, w) \in F\}$. The *reverse* transition function of A is a function $\delta^R : Q \times \Sigma \to 2^Q$, with $\delta^R(p, a) = \{q \in Q \mid \delta(q, a) = p\}$. A state $p \in Q$ is *useful* if p is *reachable*, i.e., there is $w \in \Sigma^*$ such that $\delta(q_I, w) = p$, and *productive*, i.e., if there is $w \in \Sigma^*$ such that $\delta(p, w) \in F$. In this paper we only consider automata with all useful states.

We say that two states $p, q \in Q$ are *equivalent* if and only if for all $w \in \Sigma^*$, $\delta(p, w) \in F$ exactly when $\delta(q, w) \in F$. When $p \neq q$ are equivalent states, we can reduce the size of the automaton by "merging" p and q. This would imply to merge all the states reachable from p and q by reading a same string, namely the states $\delta(p, w)$ and $\delta(q, w)$, for $w \in \Sigma^*$.

Let $A' = (Q', \Sigma, \delta', q_I', F')$ be another DFA. A *morphism* φ from A to A', in symbols $\varphi : A \to A'$, is a function $\varphi : Q \to Q'$ such that $\varphi(q_I) = q_I'$, for each $q \in Q$, $a \in \Sigma$, $\varphi(\delta(q, a)) = \delta'(\varphi(q), a)$, and $q \in F$ if and only if $\varphi(q) \in F'$. Notice that if there exists a morphism $\varphi : A \to A'$ then it is unique and, for $x, y \in \Sigma^*$, $\delta(q_I, x) = \delta(q_I, y)$ implies $\delta'(q_I', x) = \delta'(q_I', y)$. We can observe that since in all automata we are considering all the states are useful, there exists the morphism $\varphi : A \to A'$ if and only if the automaton A' can be obtained from A after merging all pairs of states p, q of A, with $\varphi(p) = \varphi(q)$ (and possibly renaming the states). Hence, the number of states of A' cannot exceed that of A. Hence $\varphi^{-1}(s)$ denotes the set of states of A which are merged in the state s of A'. Two automata A and A' are said to be *equivalent* if they accept the same language, i.e., $L(A) = L(A')$.

Let $\mathcal{C}$ be a family DFAs and $A \in \mathcal{C}$. We consider the following notions:

- The automaton A is *reduced* in $\mathcal{C}$ if for each morphism $\varphi : A \to A'$, the automaton A' does not belong to $\mathcal{C}$, i.e., every automaton obtained from A by merging some equivalent states does not belong to $\mathcal{C}$.
- The automaton A is *minimal* in $\mathcal{C}$ if and only if each automaton in $\mathcal{C}$ has at least as many states as A.
- The automaton A is the *minimum* in $\mathcal{C}$ if and only if it is the unique (up to an isomorphism, i.e., a renaming of the states) minimal automaton in $\mathcal{C}$.

Notice that each minimal automaton in a family $\mathcal{C}$ is reduced. Furthermore, if $\mathcal{C}$ contains a minimum automaton M, then M is also the only minimal and the only reduced automaton in $\mathcal{C}$. This happens, for instance, when $\mathcal{C}$ is the family of all DFAs accepting a given regular language L. However, a family $\mathcal{C}$ which does not have a minimum automaton, could contain reduced automata which are not minimal, as in the cases that will be presented in the paper.

A *strongly connected component* (SCC) C of a DFA $A = (Q, \Sigma, \delta, q_I, F)$ is a maximal subset of Q such that in the transition graph of A there exists a path between every pair of states in C. A SCC consisting of a single state q, *without* a

looping transition, is said to be *trivial*. Otherwise C is *nontrivial* and, for each state in $q \in C$, there is a string $w \in \Sigma^* \setminus \{\varepsilon\}$ such that $\delta(q, w) = q$.

We introduce a partial order $\preceq$ on the set of SCCs of M, such that, for two such components C_1 and C_2, $C_1 \preceq C_2$ when no state in C_1 can be reached from a state in C_2, but a state in C_2 is reachable from a state in C_1. We write $C_1 \npreceq C_2$ when $C_1 \preceq C_2$ is false, namely, $C_1 \neq C_2$ and either $C_2 \preceq C_1$ or C_1 and C_2 are incomparable.

Given a DFA $A = (Q, \Sigma, \delta, q_I, F)$, a state $r \in Q$ is said to be *irreversible* when $\#\delta^R(r, a) \geq 2$ for some $a \in \Sigma$, i.e., there are two transitions on the same letter entering r, otherwise r is said to be *reversible*. The DFA A is said to be *irreversible* if it contains at least one irreversible state, otherwise A is *reversible* (REV-DFA for short). As pointed out in [7], the notion of reversibility for a language is related to the computational model under consideration. In this paper we only consider DFAs. Hence, by saying that a language L is *reversible*, we refer to this model, namely we mean that there exists a REV-DFA accepting L.

The following result presents a characterization of reversible languages:

Theorem 1. *[3] Let L be a regular language and $M = (Q, \Sigma, \delta, q_I, F)$ be the minimum DFA accepting a language L. L is accepted by a REV-DFA if and only if there do not exist useful states $p, q \in Q$, a letter $a \in \Sigma$, and a string $w \in \Sigma^*$ such that $p \neq q$, $\delta(p, a) = \delta(q, a)$, and $\delta(q, aw) = q$.*

According to Theorem 1, a language L is reversible exactly when the minimum DFA accepting it does not contain the "forbidden pattern" consisting of two transitions on a same letter a entering in a same state r, with one of these transitions arriving from a state in the same SCC as r. Notice that, since transitions entering the initial state q_I can only arrive from states in the same SCC of q_I, if the language L is reversible, then the initial state q_I of M should be reversible.

An algorithm to convert a minimum DFA M into an equivalent REV-DFA, if any, was obtained in [3]. Furthermore, the resulting REV-DFA is minimal. We present an outline of it. The algorithm builds a REV-DFA A in the following way. At the beginning A is a copy of M. Then, the algorithm considers a minimal (with respect to $\preceq$) SCC C that contains an irreversible state and replace it with a number of copies which is equal to the maximum number of transitions on a same letter incoming in a state of C. This process is iterated until all the states in A are reversible.

3   Minimal Reversible Automata

In [3] it has been observed that there are reversible languages having several nonisomorphic minimal REV-DFAs. In this section we deepen that investigation by presenting a characterization of the languages having a unique minimal REV-DFA. (Notice that it could be different from the minimum DFA accepting the language.) To prove it we make use of a series of preliminary results. Hence, from now on, let us fix a reversible language L and the minimum DFA $M = (Q, \Sigma, \delta, q_I, F)$ accepting it.

Lemma 2. *Let $A' = (Q', \Sigma, \delta', q'_I, F')$ be a* REV-DFA *and $A'' = (Q'', \Sigma, \delta'', q''_I, F'')$ be a minimal* REV-DFA *both accepting L. Given the morphisms $\varphi' : A' \to M$ and $\varphi'' : A'' \to M$, it holds that $\#{\varphi'}^{-1}(s) \geq \#{\varphi''}^{-1}(s)$, for each $s \in Q$.*

Proof. By contradiction, suppose $\#{\varphi'}^{-1}(q) < \#{\varphi''}^{-1}(q)$ for some state q.

Let us partition Q in the set $Q_L = \{p \mid \exists w \in \Sigma^* \ \delta(p, w) = q\}$ of the states from which q is reachable and the set Q_R of remaining states. The sets Q' and Q'' are partitioned in a similar way, by defining $Q'_L = {\varphi'}^{-1}(Q_L)$, $Q'_R = {\varphi'}^{-1}(Q_R)$, $Q''_L = {\varphi''}^{-1}(Q_L)$, $Q''_R = {\varphi''}^{-1}(Q_R)$.

First, let us suppose $\#{\varphi'}^{-1}(p) \leq \#{\varphi''}^{-1}(p)$ for each $p \in Q_L$. We build another automaton $A''' = (Q''', \Sigma, \delta''', q'''_I, F''')$, which starts the computation by simulating A' using the states in Q'_L and, at some point, continues by simulating A'' using the states in Q''_R. In particular:

- $Q''' = Q'_L \cup Q''_R$
- The transitions are defined as follows:
    - For $s \in Q''_R$, $a \in \Sigma$: $\delta'''(s, a) = \delta''(s, a)$;
    - For $s \in Q'_L$, $a \in \Sigma$, such that $\delta'(s, a) \in Q'_L$: $\delta'''(s, a) = \delta'(s, a)$;
    - The remaining transitions, i.e., $\delta'''(s, a)$, in the case $s \in Q'_L$, $a \in \Sigma$, and $\delta'(s, a) \in Q'_R$, are obtained in the following way:
    Let us consider set of states $\{s_1, s_2, \ldots, s_k\}$ which are equivalent to s in A', i.e., $\varphi'(s_i) = \varphi'(s)$ for $i = 1, \ldots, k$ (notice that $s = s_h$ for some $h \in \{1, \ldots, k\}$), and the set of states $\{r_1, r_2, \ldots, r_j\}$ which are equivalent to s in A'', i.e., $\varphi''(r_i) = \varphi'(s)$ for $i = 1, \ldots, j$. Since $j \geq k$ we can safely define $\delta'''(s_i, a) = \delta''(r_i, a)$, for $i = 1, \ldots, k$.

The resulting automaton A''' still recognizes the language L, it is reversible and it has $\#Q'_L + \#Q''_R$ states. From $\#{\varphi'}^{-1}(p) \leq \#{\varphi''}^{-1}(p)$, for each $p \in Q_L$, and $\#{\varphi'}^{-1}(q) < \#{\varphi''}^{-1}(q)$, it follows that $\#Q'_L < \#Q''_L$, thus implying that the number of states of A''' is smaller than the one of A'', which is a contradiction.

In case $\#{\varphi'}^{-1}(p) > \#{\varphi''}^{-1}(p)$ for some $p \in Q_L$, we can apply the same construction, after switching the role of A' and A'', so producing an equivalent REV-DFA $\hat{A}'$ which is smaller than A' and still verifies $\#{\hat{\varphi}'}^{-1}(q) < \#{\varphi''}^{-1}(q)$, for the morphism $\hat{\varphi}' : \hat{A}' \to M$. Then, we iterate the proof on the two REV-DFAs $\hat{A}'$ and A''.

Hence, we can conclude that $\#{\varphi'}^{-1}(s) \geq \#{\varphi''}^{-1}(s)$, for each $s \in Q$. $\qquad \square$

Lemma 2 allows to associate with each reversible language L and the minimum DFA $M = (Q, \Sigma, \delta, q_I, F)$ accepting it, the function $c : Q \to \mathbf{N}$ such that, for $q \in Q$, $c(q)$ is the number of states equivalent to q in any minimal REV-DFA A equivalent to M, i.e., $c(q) = \#\varphi^{-1}(q)$ for the morphism $\varphi : A \to M$. Notice that $c(q_I) = 1$. Furthermore, each REV-DFA accepting L should contain at least $c(q)$ states equivalent to q. These facts are summarized in the following result, where we also show that $c(q)$ has the same value for all states belonging to the same SCC of M.
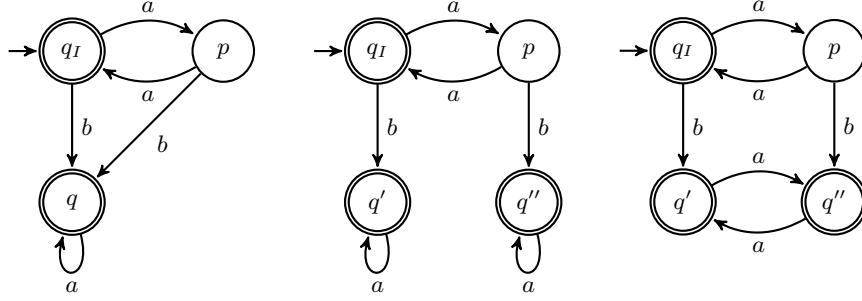
Fig. 1. A minimum DFA accepting the language $L = (aa)^* + a^*ba^*$, with two minimal nonisomorphic REV-DFAs

Lemma 3. *Let A be a* REV-DFA *accepting L, with the morphism $\varphi : A \to M$. If two states p, q of M belong to the same* SCC *of M then $\#\varphi^{-1}(p) = \#\varphi^{-1}(q) \geq c(p)$. Furthermore, if A is minimal then $c(p) = c(q) = \#\varphi^{-1}(p)$.*

Proof. Observe that since p, q belong to the same SCC there exists $x \in \Sigma^*$ such that $\delta(q, x) = p$. Let $\{q_1, q_2, \ldots, q_k\} = \varphi^{-1}(q)$ and $\{p_1, p_2, \ldots, p_j\} = \varphi^{-1}(p)$ be the sets of states in A which are equivalent to q and p, respectively. We are going to prove that $k = j$.

For each q_i, there exists p_{h_i} such that $\delta(q_i, x) = p_{h_i}$. Suppose $j < k$. In this case there are two indices i', i'' such that $p_{h_{i'}} = p_{h_{i''}}$ and then $\delta(q_{i'}, x) = \delta(q_{i''}, x) = p_{h_{i'}}$, implying that the state $p_{h_{i'}}$ is irreversible, which is a contradiction. This means that $j \geq k$. In the same way, by interchanging the roles of p and q, we can prove that $k \geq j$, which leads to the conclusion $j = k$.

The facts that $\#\varphi^{-1}(p) \geq c(p)$ and, for A minimal, $\#\varphi^{-1}(p) = c(p)$, follow from Lemma 2.   □

In the following, for each SCC C of the transition graph of M, we use $c(C)$ to denote the value $c(q)$, for $q \in C$. Considering the algorithm outlined at the end of Section 2, we can observe that if C' is another SCC, then $C \preceq C'$ implies $c(C) \leq c(C')$.

As a consequence of Lemma 3, all the minimal REV-DFAs accepting L have the same "state structure", in the sense that they should contain exactly $c(q)$ states equivalent to the state q of M. However, they could differ in the transitions (see Figure 1 for an example).

Lemma 4. *Let $A' = (Q', \Sigma, \delta', q_I', F')$ and $A'' = (Q'', \Sigma, \delta'', q_I'', F'')$ be two* REV-DFA*s accepting L. If there are no morphisms $\varphi : A' \to A''$ then there exists a state $p \in Q$ with $\#\varphi''^{-1}(p) \geq 2$ such that either $p = q_I$, or*

$$\delta^R(p, a) \neq \emptyset \text{ and } \delta^R(p, b) \neq \emptyset$$

for two symbols $a, b \in \Sigma$, with $a \neq b$, and the morphism $\varphi'' : A'' \to M$.

Proof. Since there are no morphisms $\varphi : A' \to A''$, there exist $x, y \in \Sigma^*$ such that $\delta'(q_I', x) = \delta'(q_I', y)$ and $\delta''(q_I'', x) \neq \delta''(q_I'', y)$. Among all couples of strings with this property we choose one with $|xy|$ minimal. Furthermore, we observe that it cannot be possible that $x = y = \varepsilon$.

When $x = \varepsilon$, we have $\delta'(q_I', \varepsilon) = \delta'(q_I', y) = q_I'$ and, since M is minimum, $\delta(q_I, y) = q_I$. Hence, $\varphi''(\delta''(q_I'', y)) = \varphi''(q_I'') = q_I$. From $\delta''(q_I'', y) \neq q_I'' = \delta''(q_I'', \varepsilon)$, we conclude that $\#\varphi''^{-1}(q_I) \geq 2$. The case $y = \varepsilon$ is similar.

We now consider $x \neq \varepsilon$ and $y \neq \varepsilon$, i.e., $x = ua$, $y = vb$ for some $u, v \in \Sigma^*$ and $a, b \in \Sigma$. Let $\delta'(q_I', u) = q'$, $\delta'(q_I', v) = r'$, $\delta'(q', a) = \delta'(r', b) = \bar{p}$, $\delta''(q_I'', u) = q''$, $\delta''(q_I'', v) = r''$, $\delta''(q'', a) = s$, and $\delta''(r'', b) = t$, for states $q', r', \bar{p} \in Q'$, $q'', r'', s, t \in Q''$, with $s \neq t$.

Suppose $a = b$. Since A' is reversible from $\delta'(q', a) = \delta'(r', a) = \bar{p}$ we get $q' = r'$. Furthermore $q'' \neq r''$, otherwise A'' would be nondeterministic. Hence, on the strings u, v the automaton A' reaches the same state, while A'' reaches two different states, against the minimality of $|xy|$. Thus $a \neq b$.

Given the morphism $\varphi' : A' \to M$, let $p = \varphi'(\bar{p})$. Since M is minimum, it turns out that $\varphi''(s) = \varphi''(t) = \varphi'(\bar{p}) = p$. From $s \neq t$, we conclude that $\#\varphi''^{-1}(p) \geq 2$. Furthermore, from the previous discussion, the reader can observe that there are transitions on symbols a and b entering in p. $\square$

We are now able to prove the following:

Theorem 5. *Let $M = (Q, \Sigma, \delta, q_I, F)$ be the minimum* DFA *accepting a reversible language L. The following statements are equivalent:*

1. *There exists a state $p \in Q$ such that $c(p) \geq 2$, $\delta^R(p, a) \neq \emptyset$, $\delta^R(p, b) \neq \emptyset$, for two symbols $a, b \in \Sigma$, with $a \neq b$.*
2. *There exist at least two minimal nonisomorphic* REV-DFA*s accepting L.*

Proof. (2) implies (1): By Lemma 4, given two minimal nonisomorphic REV-DFAs A' and A'' accepting L, there is a state p such that $c(p) = \#\varphi''^{-1}(p) \geq 2$. Furthermore, since $c(q_I) = 1$, $p \neq q_I$. Hence, $\delta^R(p, a) \neq \emptyset$, $\delta^R(p, b) \neq \emptyset$, for two symbols $a, b \in \Sigma$, with $a \neq b$.

(1) implies (2): Let $w \in \Sigma^*$ be a string of minimal length such that $\delta(q_I, w) = p$, $a \in \Sigma$ be its last symbol, i.e., $w = xa$, with $x \in \Sigma^*$. Let $b \in \Sigma$ be a symbol with $b \neq a$ and $\delta^R(p, b) \neq \emptyset$. Given a minimal REV-DFA $A' = (Q', \Sigma, \delta', q_I', F')$ accepting L and the morphism $\varphi : A' \to M$, we consider the state $\hat{p} = \delta'(q_I', w)$. Then $\varphi'(\hat{p}) = p$.

We show how to build a minimal REV-DFA A'' nonisomorphic to A'. The idea is to use the set of states Q' as in A' and to modify only the transitions which simulates the transitions that in M enter the state p with the letter b. There are different cases.

When $\delta'^R(\hat{p}, b) = \emptyset$, it should exist $\tilde{p} \in \varphi'^{-1}(p)$ such that $\tilde{p} \neq \hat{p}$ and $\delta'(\tilde{q}, b) = \tilde{p}$, for some $\tilde{q} \in Q'$. The automaton A'' is defined as A', with the only difference that the transition $\delta'(\tilde{q}, b) = \tilde{p}$ is replaced by $\delta''(\tilde{q}, b) = \hat{p}$. To prove that it is nonisomorphic to A', we consider a string $y \in \Sigma^*$ of minimal length such

that $\delta'(q_I', y) = \tilde{q}$. Then $\delta'(q_I', yb) = \tilde{p} \neq \delta'(q_I', w) = \hat{p}$, while $\delta''(q_I', yb) = \hat{p} = \delta''(q_I', w)$.

When $\delta'^R(\hat{p}, b) \neq \emptyset$ we can use one of the following possibilities:

- If there exists $\tilde{p} \neq \hat{p}$ such that $\delta'^R(\tilde{p}, b) \neq \emptyset$, then it should also exist $\tilde{q}, \hat{q} \in Q'$ with $\tilde{q} \neq \hat{q}$ such that $\delta'(\tilde{q}, b) = \tilde{p}$ and $\delta'(\hat{q}, b) = \hat{p}$. The automaton A'' is defined by switching the destinations of these two transitions, namely by replacing them by $\delta''(\tilde{q}, b) = \hat{p}$ and $\delta''(\hat{q}, b) = \tilde{p}$. The proof that A' and A'' are non isomorphic is exactly the same as in the previous case.
- If there exists $\tilde{p} \neq \hat{p}$ such that $\delta'^R(\tilde{p}, b) = \emptyset$, then we can consider $\hat{q}$ such that $\delta'(\hat{q}, b) = \hat{p}$, and define A'' by replacing this transition by $\delta''(\hat{q}, b) = \tilde{p}$. Let $y \in \Sigma^*$ be a string of minimal length such that $\delta'(q_I', y) = \hat{q}$. Then $\delta'(q_I', yb) = \hat{p} = \delta'(q_I', w)$. On the other hand $\delta''(q_I', yb) = \tilde{p} \neq \hat{p} = \delta''(q_I', w)$. Hence, A' and A'' are nonisomorphic.

Finally, we observe that in all cases, the automaton A'' has the same number of states as A'. Furthermore, the construction preserves reversibility.   □

As a consequence of Theorem 5 we obtain the following characterization of reversible languages having a unique minimal (hence a minimum) REV-DFA:

Corollary 6. *Let L be a reversible language and $M = (Q, \Sigma, \delta, q_I, F)$ be the minimum DFA accepting it. There exists a unique (up to isomorphism) minimal REV-DFA accepting L if and only if for each state $p \in Q$ with $c(p) \geq 2$, all the transitions entering in $p \in Q$ are on the same symbol.*

When the minimum DFA accepting a reversible language contains a loop in the irreversible part, i.e., in the part "after" an irreversible state, the condition in Corollary 6 is always false, hence there exist at least two minimal nonisomorphic REV-DFAs. This is proved in the following result:

Theorem 7. *Let $M = (Q, \Sigma, \delta, q_I, F)$ be the minimum DFA accepting a reversible language L. If there exists an irreversible state $q \in Q$ such that the language accepted by computations starting in q is infinite, then there exists a state $p \in Q$ such that $c(p) \geq 2$, $\delta^R(p, a) \neq \emptyset$ and $\delta^R(p, b) \neq \emptyset$, for two symbols $a, b \in \Sigma$, with $a \neq b$.*

Proof. Let $p \in Q$ be a state reachable from q which belongs to a nontrivial SCC C. Hence $c(p) \geq 2$. Among all possibilities, we choose p in such a way that all the other states on a fixed path from q to p does not belong to C. Since C is nontrivial, it should exist a transition from a state of C, which enters in p. Let $a \in \Sigma$ be the symbol of such transition. Furthermore, it should exist another transition which enters in p from a state which does not belong to C. (If $p \neq q$ then we can take the last transition on the fixed path. Otherwise, since the initial state is always reversible, we have $q \neq q_I$, and so we can take the last transition entering in q on a path from q_I.) Let b the symbol of such transition. If $a = b$ the automaton M would contain the forbidden pattern (cfr. Theorem 1), thus implying that L is not reversible. Hence, we conclude $a \neq b$.   □
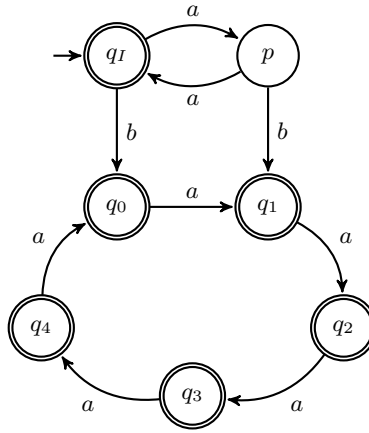
Fig. 2. A reduced REV-DFA

As a consequence of Theorem 7, considering Corollary 6 we can observe that when a reversible language has a unique minimal REV-DFA, all the loops in the minimum DFA accepting it should be in the reversible part. However, the converse does not hold, namely there are languages whose minimum DFA does not contain any loop in the irreversible part, which does not have a unique minimal REV-DFA. Indeed, in [3] an example with a finite language is presented.

4   Reduced Reversible Automata

In the section we show that there exist REV-DFAs which are reduced but not minimal, namely they have more states than equivalent minimal REV-DFAs, but merging some of their equivalent states would produce an irreversible automaton. Furthermore, we will prove that there exist reversible languages having arbitrarly large reduced REV-DFAs and, hence, infinitely many reduced REV-DFAs.

In Figure 2 a reduced REV-DFA equivalent to the DFAs in Figure 1 is depicted. If we try to merge two states in the loop, then the loop collapses to unique state, so producing the minimum DFA, which is irreversible. Actually, this example can be modified by using a loop of N states: if (and only if) N is prime, we get a reduced automaton. This is a special case of the construction which we are now going to present:

Theorem 8. *Let* $M = (Q, \Sigma, \delta, q_I, F)$ *be the minimum* DFA *accepting a reversible language* L. *If* M *contains a state* q *such that* $c(q) \geq 2$ *and the language accepted by computations starting in* q *is infinite, then there exist infinitely many nonisomorphic reduced* REV-DFA*s accepting* L.

Proof. Without loss of generality, we assume that the SCC C_q containing q is nontrivial. In fact, if this is not the case, we can find a state $\bar{q}$ which is reachable

from q, and so $c(\bar{q}) \geq c(q) \geq 2$, and which belongs to a nontrivial SCC. Then, we can give the proof replacing q by $\bar{q}$.

Let A be a minimal REV-DFA A accepting L, obtained applying the algorithm outlined in Section 2, and $N \geq c(q)$ an integer. The idea is to modify A by replacing the part corresponding to the SCC C_q, with N copies of each state in C_q and arranging the transitions in such a way that all the states in these N copies form one SCC, without changing the accepted language. Furthermore, all the SCC that follow C_q will be replicated a certain number of times. More precisely, we build a DFA $A_N = (Q_N, \Sigma, \delta_N, q_{IN}, F_N)$ using the following steps:

(i) We put in A_N all the states of A which correspond to SCCs C of M with $C_q \not\preceq C$ and all the transitions between these states.
(ii) We add N copies of the states in C_q to the set of states of A_N. Given a state $r \in C_q$, let us denote its copies as $r_0, r_1, \ldots, r_{N-1}$.
(iii) We fix a transition $\delta(q, a) = q'$ of M, with $q, q' \in C_q$. For $i = 0, \ldots, N-1$, we define $\delta_N(q_i, a) = q'_{(i+1) \bmod N}$, and for the remaining transitions, namely $\delta(r, b) = r'$ with $(r, b) \neq (q, a)$, we define $\delta_N(r_i, b) = r'_i$. In this way in A_N we have N copies of the SCC C_q, modified in such a way that the transition from s_i on a in copy i leads to the state $q'_{(i+1) \bmod N}$ in copy $(i+1) \bmod N$.
(iv) We add to A_N each transition that in A leads from a state added in (i) to one state in the first $c(q)$ copies of C_q added in (iii). (We remind the reader that A should contain $c(q)$ copies of the SCC C_q. Hence, in A_N we keep exactly the same connections as in A from the states at point (i) to the states in these copies.)
(v) We complete the construction of A_N by adding a suitable number of copies of the remaining SCCs of M and suitable transitions, in order to derive a REV-DFA. This can be done just following the steps of the algorithm described in Section 2.

By construction, the automaton A_N so obtained is reversible and it accepts L. We are going to show that when N is a prime number then A_N is reduced. To this aim we shall prove that if we try to merge two equivalent states p', p'' of A_N then we obtain an irreversible automaton. The proof is divided in three cases:

- p', p'' are equivalent to a state p of M with $C_q \not\preceq C_p$, where C_p denotes the SCC containing p.
  These states have been added at step (i), copying them from the minimal REV-DFA A. By Lemma 3, A contains exactly $c(p)$ states equivalent to p. Hence, merging p' and p'', the resulting automaton would contain less than $c(p)$ states equivalent to p and, hence, it cannot be reversible.
- p', p'' are equivalent to a state p of M belonging to C_q.
  First, suppose $p' = q_0$ and $p'' = q_j$, $0 < j < n$. Considering step (iii), we observe that there is a string z such that $\delta(q', z) = q$, then $\delta(q, w) = q$ and $\delta_N(q_i, w) = q_{(i+1) \bmod N}$, where $w = az$. Thus, for each $k \geq 0$, $\delta(q_0, w^{k(N-j)}) = q_{k(N-j) \bmod N}$ and $\delta(q_j, w^{k(N-j)}) = q_{j+k(N-j) \bmod N} = q_{(k-1)(N-j) \bmod N}$. Hence, merging q_0 and q_j would imply merging all the

states whose indices are in the set $\{k(N - j) \bmod N \mid k \geq 0\}$, which, be-
ing N prime, coincides with $\{0, \ldots, N-1\}$. As a consequence, all the states q_i,
should collapse in a unique state. However, since $c(q) \geq 2$, by Lemma 3 this
implies that the resulting automaton is not reversible.
If $p' \neq q_0$, then we can always find a string y such that $\delta_N(p', y) = q_0$.
Using the transitions introduced at step (iii), we get that $\delta_N(p'', y) = q_j$, for
some $0 < j < N$. Hence, merging p' and p'' would imply merging q_0 and q_j,
so reducing to the previous case.

- p', p'' are equivalent to a state p of M, such that $C_p \neq C_q$ and $C_q \preceq C_p$.
  Let $w \in \Sigma^*$ be such that $\delta(q, w) = p$ and $p' = \delta_N(q', w)$, $p'' = \delta_N(q'', w)$.
  From $p' \neq p''$, using the fact that A_N is reversible, we obtain $q' \neq q''$. So, to
  keep reversibility, merging p' and p'' would imply merging q' and q'', which
  are equivalent to q, so reducing to the previous case.

In summary, for each prime number $N \geq c(q)$ we obtained a reduced REV-
DFA A_N with more than N states accepting the language L. Hence, we can
conclude that there exist infinitely many nonisomorphic reduced REV-DFA ac-
cepting L.   □

In Theorem 8 we gave a sufficient condition for the existence of infinitely
many reduced REV-DFAs accepting a given language. This condition is not nec-
essary. In fact, even if minimum DFA does not contain any loop in the irreversible
part, it could be possible to construct infinitely many reduced REV-DFAs. For
instance, by modifying the construction given to prove Theorem 8, we can show
that if the minimum DFA for a language L has a state p in the irreversible part,
which is entered by transitions on at least two different letters (cfr. Thm. 7) and
those transitions are used to recognize infinitely many strings, then there are
infinitely many reduced REV-DFAs accepting L.

5   Conclusion

In this paper we studied the existence of minimal and reduced REV-DFAs. In some
cases the minimum DFA accepting a language is already reversible, so assuring
that the language is reversible. However, in general a minimum DFA does not
need to be reversible, although the accepted language could be reversible. Using
Theorem 1 and the construction from [3] outlined in Section 2, in the case the
language is reversible, from a given minimum DFA we can obtain a minimal REV-
DFA. Minimal REV-DFAs are not necessarily unique (see Figure 1 for an example,
while Figure 3 shows a case with a unique minimal, and hence minimum, REV-
DFA). In Section 3 we gave a characterization of the languages having a unique
minimum REV-DFA, in terms of the structure of the minimum DFA.

Here we wanted to go beyond the investigation of minimal REV-DFAs studying
reduced REV-DFAs. We observed the existence of reduced REV-DFAs which are
not minimal and we gave a sufficient condition for the existence of infinitely
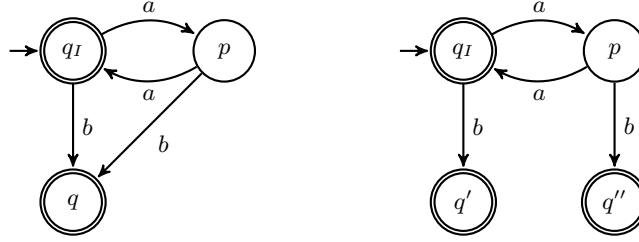many reduced REV-DFAs accepting a same reversible language.

Fig. 3. The minimum DFA and the minimum REV-DFA accepting the language $L = (aa)^* + a^*b$

References

1. Angluin, D.: Inference of reversible languages. J. ACM 29(3), 741–765 (1982)
2. Bennett, C.: Logical reversibility of computation. IBM Journal of Research and Development 17(6), 525–532 (1973)
3. Holzer, M., Jakobi, S., Kutrib, M.: Minimal reversible deterministic finite automata. In: Potapov, I. (ed.) Developments in Language Theory - 19th International Conference, DLT 2015, Liverpool, UK, July 27-30, 2015, Proceedings. Lecture Notes in Computer Science, vol. 9168, pp. 276–287. Springer (2015)
4. Hopcroft, J.E., Ullman, J.D.: Introduction to Automata Theory, Languages and Computation. Addison-Wesley (1979)
5. Kondacs, A., Watrous, J.: On the power of quantum finite state automata. In: FOCS. pp. 66–75. IEEE Computer Society (1997)
6. Kutrib, M.: Aspects of reversibility for classical automata. In: Calude, C.S., Freivalds, R., Kazuo, I. (eds.) Computing with New Resources - Essays Dedicated to Jozef Gruska on the Occasion of His 80th Birthday. Lecture Notes in Computer Science, vol. 8808, pp. 83–98. Springer (2014)
7. Kutrib, M.: Reversible and irreversible computations of deterministic finite-state devices. In: Italiano, G.F., Pighizzini, G., Sannella, D. (eds.) Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24-28, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9234, pp. 38–52. Springer (2015)
8. Landauer, R.: Irreversibility and heat generation in the computing process. IBM Journal of Research and Development 5(3), 183–191 (July 1961)
9. Lange, K., McKenzie, P., Tapp, A.: Reversible space equals deterministic space. J. Comput. Syst. Sci. 60(2), 354–367 (2000)
10. Lombardy, S.: On the construction of reversible automata for reversible languages. In: Widmayer, P., Ruiz, F.T., Bueno, R.M., Hennessy, M., Eidenbenz, S., Conejo, R. (eds.) Automata, Languages and Programming, 29th International Colloquium, ICALP 2002, Malaga, Spain, July 8-13, 2002, Proceedings. Lecture Notes in Computer Science, vol. 2380, pp. 170–182. Springer (2002)
11. Pin, J.: On reversible automata. In: Simon, I. (ed.) LATIN '92, 1st Latin American Symposium on Theoretical Informatics, São Paulo, Brazil, April 6-10, 1992, Proceedings. Lecture Notes in Computer Science, vol. 583, pp. 401–416. Springer (1992)