

# Congruence Relations for Büchi Automata

Yong Li<sup>1</sup>, Yih-Kuen Tsay<sup>2</sup>, Moshe Y. Vardi<sup>3</sup>

<sup>1</sup> State Key Laboratory of Computer Science,  
Institute of Software, Chinese Academy of Sciences

<sup>2</sup> National Taiwan University

<sup>3</sup> Rice University

**Abstract.** We revisit here congruence relations for Büchi automata, which play a central role in the automata-based verification. The size of the classical congruence relation is in  $3^{\mathcal{O}(n^2)}$ , where  $n$  is the number of states of a given Büchi automaton  $\mathcal{A}$ . Here we present improved congruence relations that can be exponentially more compact than the classical one. We further give asymptotically *optimal* congruence relations of size  $2^{\mathcal{O}(n \log n)}$ . Based on these optimal congruence relations, we obtain an *optimal* translation from Büchi automata to a family of deterministic finite automata (FDFW) that accepts the complementary language. To the best of our knowledge, our construction is the *first direct* and optimal translation from Büchi automata to FDFWs.

## 1 Introduction

Congruence relations for nondeterministic Büchi automata on words (NBWs) [6] are fundamental for Büchi complementation, a key component in the automata-based model checking [21] framework. The model-checking problem of whether the behavior of the system  $A$  satisfies the specification  $B$  reduces to a language-containment problem between the NBWs  $A$  and  $B$ , which is reduced to the intersection of  $A$  and the complement of  $B$ . The first complementation construction for Büchi automata, widely known as Ramsey-based Büchi complementation (RBC) proposed by Büchi [6], relies on a congruence relation with a doubly exponential blow-up with respect to the input automaton. There is a relation between each state of the complementary automaton and an equivalence class of the congruence relation, in a manner similar to the Myhill-Nerode theorem for regular languages [20]. The congruence relation of RBC was later improved by Sistla, Vardi, and Wolper in 1987 with a blow-up of  $3^{\mathcal{O}(n^2)}$  [18]; this classical congruence relation was then formalized in [20].

Notably, current practical approaches to the containment checking for NBWs are all based on the classical congruence relation in [18, 20], even though it has a larger blow-up than other optimal complementation constructions ( $3^{\mathcal{O}(n^2)}$  vs.  $2^{\mathcal{O}(n \log n)}$ ), such as *rank-based* complementation [14]. In fact, RABIT, based on the classical congruence relation, is the state-of-the-art tool for checking

language-containment between NBWs, and has integrated various state-space pruning techniques proposed in [1, 2, 8] for RBC.

Recently, a new type of automata called *family of deterministic finite automata on words* (FDFWs) [3] has been proposed for representing  $\omega$ -regular languages, as an alternative to NBWs. If we model system and specifications as FDFWs, then the model-checking problem can reduce to a containment problem between two FDFWs, which is doable in polynomial time [3], in contrast to PSPACE-complete for NBWs [13]. It has been shown that FDFWs can be induced from the congruence relation defined over a given  $\omega$ -regular language, where each state of the FDFW corresponds to an equivalence class of the congruence relation [4]. Notably, the congruence relation also enables an algorithm to learn an FDFW by interacting with an oracle that knows the language of the target FDFW [4].

In this work, we demonstrate that RBC and FDFWs have an intimate connection and the underlying concept that connects them is the congruence relation for NBWs. This connection gives us a possibility to tighten the congruence relations for both RBC and FDFWs. In fact, the state-space pruning techniques developed in [1, 2] for RBC in [18] are inherently heuristics for identifying subsumptions and simulations between congruence relations of RBC. Therefore, in order to further theoretically or empirically improve model-checking algorithms based on RBC or FDFWs, it is important to understand the congruence relations for both FDFWs and RBC and, hopefully, make their congruence relations smaller.

*Contribution.* We focus here on an in-depth study of the congruence relation for NBWs and its connection to FDFWs. First, we show how to improve the classical congruence relation  $\sim$  with a blow-up of  $3^{\mathcal{O}(n^2)}$  defined by the classical RBC to congruence relations that can be exponentially tighter (Theorem 3), while the improved congruence relations can never be larger than the classical congruence relation  $\sim$  (Theorem 2). Notably, the improved congruence relations only have a blow-up of  $\mathcal{O}(n^2)$  when dealing with a deterministic Büchi automaton (Theorem 4). Second, we further propose congruence relations for NBWs with a blow-up of only  $2^{\mathcal{O}(n \log n)}$  (Lemma 9), which is then proved to be optimal (Theorem 5). Finally, we show that our congruence relations define an FDFW recognizing  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A})$  from an NBW  $\mathcal{A}$ . In particular, if  $\mathcal{A}$  has  $n$  states, then our optimal congruence relations yield an FDFW  $\mathcal{F}$  with an optimal complexity  $2^{\mathcal{O}(n \log n)}$ . Thus, to the best of our knowledge, we have presented the *first direct* translation from an NBW to an FDFW with *optimal* complexity.

We defer all missing proofs of the paper to Appendix A.

## 2 Preliminaries

We fix an *alphabet*  $\Sigma$ . A *word* is a finite or infinite sequence  $w$  of letters in  $\Sigma$ . We denote by  $\Sigma^*$  and  $\Sigma^\omega$  the set of all finite and infinite words (or  $\omega$ -words), respectively. A *finitary language* is a subset of  $\Sigma^*$  while an  $\omega$ -*language* is a subset of  $\Sigma^\omega$ . Let  $L$  be a finitary language (resp.,  $\omega$ -language); the complementary

language of  $L$  is  $\Sigma^* \setminus L$  (resp.,  $\Sigma^\omega \setminus L$ ). Let  $\rho$  be a sequence; we denote by  $\rho[i]$  the  $i$ -th element of  $\rho$  and we use  $\rho[i..k]$  to denote the subsequence of  $\rho$  starting at the  $i$ -th element and ending at the  $k$ -th element, inclusively, when  $i \leq k$ , and the empty sequence  $\epsilon$  when  $i > k$ . Given a finite word  $u$  and a word  $w$ , we denote by  $u \cdot w$  ( $uw$ , for short) the concatenation of  $u$  and  $w$ . Given a finitary language  $L_1$  and a finitary/ $\omega$ -language  $L_2$ , we denote by  $L_1 \cdot L_2$  ( $L_1 L_2$ , for short) the concatenation of  $L_1$  and  $L_2$ , i.e.,  $L_1 \cdot L_2 = \{uw \mid u \in L_1, w \in L_2\}$  and  $L_1^\omega$  the infinite concatenation of  $L_1$ .

## 2.1 NBWs

A (nondeterministic) automaton is a tuple  $\mathcal{A} = (Q, I, \delta, F)$ , where  $Q$  is a finite set of states,  $I \subseteq Q$  is a set of initial states,  $\delta: Q \times \Sigma \rightarrow 2^Q$  is a transition function, and  $F \subseteq Q$  is a set of accepting states. We extend  $\delta$  to sets of states, by letting  $\delta(S, a) = \bigcup_{q \in S} \delta(q, a)$ . We also extend  $\delta$  to words, by letting  $\delta(q, \epsilon) = \{q\}$  and  $\delta(q, a_1 a_2 \dots a_k) = \delta(\delta(q, a_1), \dots, a_k)$ , where we have  $k \geq 1$  and  $a_i \in \Sigma$  for  $i \in \{1, \dots, k\}$ . An automaton on finite words is called a *nondeterministic automaton on finite words* (NFW), while an automaton on  $\omega$ -words is called a *nondeterministic Büchi automaton on infinite words* (NBW). A *run* of an automaton  $\mathcal{A}$  on a finite word  $u$  of length  $n \geq 0$  is a sequence of states  $\rho = q_0 q_1 \dots q_n \in Q^+$ , such that for every  $0 < i \leq n$ ,  $q_i \in \delta(q_{i-1}, u[i])$ ; a finite word  $u \in \Sigma^*$  is *accepted* by an NFW  $\mathcal{A}$  if there is a run  $q_0 \dots q_n$  such that  $q_0 \in I$  and  $q_n \in F$ . Similarly, an  $\omega$ -run of  $\mathcal{A}$  on an  $\omega$ -word  $w$  is an infinite sequence of states  $\rho = q_0 q_1 \dots$  such that  $q_0 \in I$  and for every  $i > 0$ ,  $q_i \in \delta(q_{i-1}, a_i)$ . We denote by  $\text{Inf}(\rho)$  the set of states that occur infinitely often in the run  $\rho$ . An  $\omega$ -word  $w \in \Sigma^\omega$  is *accepted* by an NBW  $\mathcal{A}$  if there exists an  $\omega$ -run  $\rho$  of  $\mathcal{A}$  over  $w$  such that  $\text{Inf}(\rho) \cap F \neq \emptyset$ . The *finitary language* recognized by an NFW  $\mathcal{A}$ , denoted by  $\mathcal{L}_*(\mathcal{A})$ , is defined as the set of finite words accepted by it. Similarly, we denote by  $\mathcal{L}(\mathcal{A})$  the  $\omega$ -language recognized by an NBW  $\mathcal{A}$ , i.e., the set of  $\omega$ -words accepted by  $\mathcal{A}$ . The complementary automaton of an NBW  $\mathcal{A}$ , denoted as  $\mathcal{A}^c$ , accepts the complementary language of  $\mathcal{L}(\mathcal{A})$ , i.e.,  $\mathcal{L}(\mathcal{A}^c) = \Sigma^\omega \setminus \mathcal{L}(\mathcal{A})$ .

We denote by  $q \xrightarrow{u} r$  if there is a run of  $\mathcal{A}$  from  $q$  to  $r$  over  $u \in \Sigma^*$  where the path is constructed with the transition function  $\delta$  of  $\mathcal{A}$ . Obviously, we have  $q \xrightarrow{\epsilon} q$  for all  $q \in Q$ . Further, we denote by  $q \xRightarrow{u} r$  if there is a run of  $\mathcal{A}$  from  $q$  to  $r$  over  $u \in \Sigma^*$  that visits an accepting state of  $\mathcal{A}$ .

An NFW  $\mathcal{A}$  is said to be a *deterministic* finite-word automaton (DFW) if  $|I| = 1$  and for each  $q \in Q$  and  $a \in \Sigma$ ,  $|\delta(q, a)| \leq 1$ . DBW is defined similarly.

## 2.2 Congruence Relations

A *congruence relation* is an equivalence relation  $\sim$  over  $\Sigma^*$  such that  $x \sim y$  implies  $uxv \sim uyv$  for every  $x, y, u, v \in \Sigma^*$ . Specially, if  $x \sim y$  implies  $xv \sim yv$  for all  $v \in \Sigma^*$ ,  $\sim$  is a *right congruence*. We denote by  $|\sim|$  the index of  $\sim$ , i.e., the number of equivalence classes of  $\sim$ . A *finite congruence relation* is a congruence relation with a finite index. We use  $\Sigma^*/\sim$  to denote the set of equivalence classes

of  $\sim$ . For a word  $x \in \Sigma^*$ , we denote by  $[x]_{\sim}$  the equivalence class of  $\sim$  that  $x$  belongs to.

For a given right congruence  $\sim$  of a regular language  $L$ , it is well-known that Myhill-Nerode theorem [16, 17] defines a unique minimal DFW  $D$  of  $L$ , in which each state of  $D$  corresponds to an equivalence class defined by  $\sim$  over  $\Sigma^*$ . Therefore, we can construct a DFW  $\mathcal{D}[\sim]$  from  $\sim$  in a standard way.

**Definition 1 ([16, 17]).** *Let  $\sim$  be a right congruence of finite index. The DFW  $\mathcal{D}[\sim]$  without accepting states induced by  $\sim$  is a tuple  $(S, s_0, \delta_{\mathcal{D}}, \emptyset)$  where  $S = \Sigma^*/_{\sim}$ ,  $s_0 = [\epsilon]_{\sim}$ , and for each  $u \in \Sigma^*$  and  $a \in \Sigma$ ,  $\delta_{\mathcal{D}}([u]_{\sim}, a) = [ua]_{\sim}$ .*

The resulting DFW  $\mathcal{D}[\sim]$ , parameterized with  $\sim$ , indicates that the DFW is induced from the right congruence relation  $\sim$ . We may write  $\mathcal{D}$  instead of  $\mathcal{D}[\sim]$  if  $\sim$  is clear from the context.

### 2.3 FDFWs

The  $\omega$ -regular languages accepted by NBWs can also be recognized by FDFWs by means of their *ultimately periodic words* (UP-words) [3]. A UP-word  $w$  is an  $\omega$ -word of the form  $uv^\omega$ , where  $u \in \Sigma^*$  and  $v \in \Sigma^+$ . Thus  $w = uv^\omega$  can be represented as a pair of words  $(u, v)$  called a *decomposition* of  $w$ . A UP-word can have multiple decompositions: for instance  $(u, v)$ ,  $(uv, v)$ , and  $(u, vv)$  are all decompositions of  $uv^\omega$ . For an  $\omega$ -language  $L$ , let  $\text{UP}(L) = \{uv^\omega \in L \mid u \in \Sigma^*, v \in \Sigma^+\}$  denote the set of all UP-words in  $L$ . The set of UP-words of an  $\omega$ -regular language  $L$  can be seen as the fingerprint of  $L$ , as stated below.

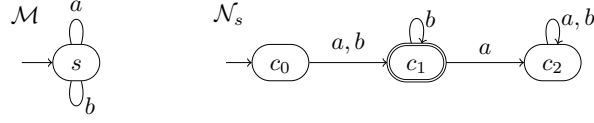
**Theorem 1 ([6]).** (1) *Every non-empty  $\omega$ -regular language  $L$  contains at least one UP-word.* (2) *Let  $L, L'$  be two  $\omega$ -regular languages. Then  $L = L'$  if and only if  $\text{UP}(L) = \text{UP}(L')$ .*

Based on Theorem 1, Angluin *et al.* introduced in [3] the notion of FDFWs as another type of automata to recognize  $\omega$ -regular languages.

**Definition 2 (FDFWs [3]).** *An FDFW is a pair  $\mathcal{F} = (\mathcal{M}, \{\mathcal{N}_q\})$  consisting of a leading DFW  $\mathcal{M}$  and of a progress DFW  $\mathcal{N}_q$  for each state  $q$  in  $\mathcal{M}$ .*

Intuitively, the leading DFW  $\mathcal{M}$  of  $\mathcal{F} = (\mathcal{M}, \{\mathcal{N}_q\})$  for an  $\omega$ -regular language  $L$  recognizes the finite prefix  $u$  of a UP-word  $uv^\omega \in \text{UP}(L)$  and for each state  $q$  of  $\mathcal{M}$ , the progress DFW  $\mathcal{N}_q$  accepts the periodic part  $v$  of  $uv^\omega$ . An example of FDFW  $\mathcal{F}$  is depicted in Figure 1 where the leading DFW  $\mathcal{M}$  has only one state  $s$  and the progress DFW associated with  $s$  is  $\mathcal{N}_s$ . Note that the leading DFW  $\mathcal{M}$  of every FDFW has no accepting states.

Let  $\mathcal{D}$  be a DFW whose initial state and transition function are  $q_0$  and  $\delta$  respectively; for a word  $u \in \Sigma^*$ , we often use  $\mathcal{D}(u)$  as a shorthand for  $\delta(q_0, u)$ . Each FDFW  $\mathcal{F}$  characterizes a set of UP-words  $\text{UP}(\mathcal{F})$  by the acceptance condition defined as follows.



**Fig. 1.** An example of FDFW  $\mathcal{F} = (\mathcal{M}, \{\mathcal{N}_s\})$  which is not saturated.

**Definition 3 (FDFW Acceptance).** Let  $\mathcal{F} = (\mathcal{M}, \{\mathcal{N}_q\})$  be an FDFW and  $w$  a UP-word. A decomposition  $(u, v)$  of  $w$  is normalized with respect to  $\mathcal{F}$  if  $\mathcal{M}(u) = \mathcal{M}(uv)$ <sup>1</sup>. A decomposition  $(u, v)$  is captured by  $\mathcal{F}$  if we have  $v \in \mathcal{L}_*(\mathcal{N}_q)$  where  $q = \mathcal{M}(u)$ . A decomposition  $(u, v)$  is accepted by  $\mathcal{F}$  if  $(u, v)$  is normalized and is captured by  $\mathcal{F}$ .  $w$  is accepted by  $\mathcal{F}$  if there exists a decomposition  $(u, v)$  of  $w$  accepted by  $\mathcal{F}$ .

Note that the normalized decomposition  $(u, v)$  is defined with respect to  $\mathcal{F}$ . We usually omit  $\mathcal{F}$  and just say  $(u, v)$  is normalized when  $\mathcal{F}$  is clear from the context. Consider again the FDFW  $\mathcal{F}$  in Figure 1:  $(aba)^\omega$  is not accepted by  $\mathcal{F}$  since no decomposition of  $(aba)^\omega$  is accepted by  $\mathcal{F}$ , while  $(ab)^\omega$  is accepted by  $\mathcal{F}$  since there exists the decomposition  $(ab, ab)$  of  $(ab)^\omega$  such that  $\mathcal{M}(ab \cdot ab) = \mathcal{M}(ab) = s$  and  $ab \in \mathcal{L}_*(\mathcal{N}_s)$ .

One can observe that the normalized decomposition  $(ab, abab)$  of  $(ab)^\omega$  is not accepted by  $\mathcal{F}$ , despite that  $(ab, ab)$  is accepted by  $\mathcal{F}$ . In the following, we define a class of FDFWs that *saturates* each accepting normalized  $(ab, (ab)^k)$  with  $k \geq 1$  of  $(ab)^\omega$  if  $(ab, ab)$  is accepted, which is important for FDFWs to recognize  $\omega$ -regular languages [3, 15].

**Definition 4 (Saturation of FDFWs [3]).** Let  $\mathcal{F} = (\mathcal{M}, \{\mathcal{N}_q\})$  be an FDFW and  $w$  a UP-word in  $UP(\mathcal{F})$ . We say  $\mathcal{F}$  is saturated if for all normalized decompositions  $(u, v)$  and  $(u', v')$  of  $w$ , either both  $(u, v)$  and  $(u', v')$  are accepted by  $\mathcal{F}$  or both are not.

Intuitively, for a saturated FDFW  $\mathcal{F}$ , a UP-word  $w$  is accepted by  $\mathcal{F}$  if and only if all normalized decompositions  $(u, v)$  of  $w$  are accepted by  $\mathcal{F}$ . From a saturated FDFW  $\mathcal{F}$ , one can construct an equivalent NBW  $\mathcal{A}$  that recognizes  $UP(\mathcal{F})$  in polynomial time.

**Lemma 1 (Polynomial Equivalent Translation from FDFWs to NBWs [3, 15]).** Let  $\mathcal{F} = (\mathcal{M}, \{\mathcal{N}_q\})$  be a saturated FDFW with  $n$  states. Then, one can construct an NBW  $\mathcal{A}$  with  $\mathcal{O}(n^3)$  states such that  $UP(\mathcal{F}) = UP(\mathcal{L}(\mathcal{A}))$ .

We note, however, that an FDFW that is *not* saturated does not necessarily recognize an  $\omega$ -regular language according to [15], let alone permit an equivalent translation to NBWs.

<sup>1</sup> We use the normalized decomposition of UP-words defined in [15], which is different from the one given in [3]. Ours is a definition for a UP-word, while their definition is applied to a decomposition. However, this difference does not affect the definition of a saturated FDFW to be given later.

In the remainder of the paper, we fix an NBW  $\mathcal{A} = (Q, I, \delta, F)$ , unless explicitly stated otherwise, where  $\mathcal{A}$  has  $n$  states, i.e.,  $n = |Q|$ . We call a state in an FDFW a *macrostate* to distinguish it from states of  $\mathcal{A}$ .

### 3 Improved Congruence Relations for NBWs

We first review the classical congruence relations defined in [18, 20] in Sect. 3.1 and then give improved congruence relations in Sect. 3.2.

#### 3.1 Classical Congruence Relations

As mentioned in the introduction, the size of the congruence relation of RBC proposed by Büchi [6] is doubly exponential in the number of states of  $\mathcal{A}$ . Sistla, Vardi, and Wolper [18] showed how to improve RBC with a subset construction; the improved construction was later presented by Thomas [20] as the following congruence relation  $\sim_{\mathcal{A}}$ .

**Definition 5 ([18, 20]).** *In the RBC construction, for all  $u_1, u_2 \in \Sigma^*$ ,  $u_1 \sim_{\mathcal{A}} u_2$  if and only if for all pairs of states  $q, r \in Q$  of  $\mathcal{A}$ , (1)  $q \xrightarrow{u_1} r$  iff  $q \xrightarrow{u_2} r$ ; and (2)  $q \xRightarrow{u_1} r$  iff  $q \xRightarrow{u_2} r$ .*

One can easily verify that if  $u_1 \sim_{\mathcal{A}} u_2$ , then  $xu_1y \sim_{\mathcal{A}} xu_2y$  for all  $x, y \in \Sigma^*$ , so  $\sim_{\mathcal{A}}$  is a (right-)congruence relation. Since we fix an NBW  $\mathcal{A}$  throughout the paper, we write  $\sim$  instead of  $\sim_{\mathcal{A}}$  as  $\mathcal{A}$  is clear from the context. Next we cite the result in [18, 20] that the congruence relation  $\sim$  is of finite index.

**Lemma 2 ([18, 20]).** *Let  $\sim$  be the congruence relation in Definition 5. Then  $|\sim| \leq 3^{n^2}$ .*

The congruence relation  $\sim$  is defined by reachability between states. For an equivalence class  $[u]_{\sim}$ , there are at most  $n \times n$  distinct pairs of states and for each pair of states  $q$  and  $r$ , we either have both  $q \xRightarrow{u} r$  and  $q \xrightarrow{u} r$ , just  $q \xrightarrow{u} r$  or the pair is not present. Thus we have  $|\sim| = |\Sigma^*/_{\sim}| \leq 3^{n^2}$ .

We recall below some known results in [18], adapted to our notations. One can verify that  $\Sigma^{\omega} = \bigcup \{ [u]_{\sim} [v]_{\sim}^{\omega} \mid [u]_{\sim}, [v]_{\sim} \in \Sigma^*/_{\sim} \}$  since  $\Sigma^* = \bigcup \{ [u]_{\sim} \mid [u]_{\sim} \in \Sigma^*/_{\sim} \}$  and  $|\sim| \leq 3^{n^2}$ . The words  $uv^{\omega}$  with  $v = \epsilon$  are finite words and thus omitted in  $\Sigma^{\omega}$ . Moreover, if  $v \in [\epsilon]_{\sim}$ , we can also use  $v$  as the representative word of  $[\epsilon]_{\sim}$ , i.e.,  $[v]_{\sim} = [\epsilon]_{\sim}$ . We now describe a type of  $\omega$ -languages called *proper languages* described in [18].

**Definition 6 ([18, 20]).** *For two classes  $[u]_{\sim}, [v]_{\sim} \in \Sigma^*/_{\sim}$ , the language  $[u]_{\sim} [v]_{\sim}^{\omega}$  is proper if  $[u]_{\sim} [v]_{\sim} \subseteq [u]_{\sim}$  and  $[v]_{\sim} [v]_{\sim} \subseteq [v]_{\sim}$ .*

Moreover, a proper language  $[u]_{\sim} [v]_{\sim}^{\omega}$  is either completely inside  $\mathcal{L}(\mathcal{A})$  or outside  $\mathcal{L}(\mathcal{A})$ , which we call the *saturation lemma* of the congruence relation  $\sim$  as below.

**Lemma 3 (Saturation Lemma [18, 20]).**

1. For  $[u]_{\sim}, [v]_{\sim} \in \Sigma^*/_{\sim}$ , if  $[u]_{\sim}[v]_{\sim}^{\omega}$  is proper, either  $\mathcal{L}(\mathcal{A}) \cap [u]_{\sim}[v]_{\sim}^{\omega} = \emptyset$  or  $\mathcal{L}(\mathcal{A}) \cap [u]_{\sim}[v]_{\sim}^{\omega} = [u]_{\sim}[v]_{\sim}^{\omega}$ .
2.  $\Sigma^{\omega} = \bigcup \{ [u]_{\sim}[v]_{\sim}^{\omega} \mid [u]_{\sim}, [v]_{\sim} \in \Sigma^*/_{\sim}, [u]_{\sim}[v]_{\sim}^{\omega} \text{ is proper} \}$ .
3.  $\Sigma^{\omega} \setminus \mathcal{L}(\mathcal{A}) = \bigcup \{ [u]_{\sim}[v]_{\sim}^{\omega} \mid [u]_{\sim}, [v]_{\sim} \in \Sigma^*/_{\sim}, [u]_{\sim}[v]_{\sim}^{\omega} \cap \mathcal{L}(\mathcal{A}) = \emptyset, [u]_{\sim}[v]_{\sim}^{\omega} \text{ is proper} \}$ .

For the set equality in the saturation lemma to hold, we, again, exclude all finite words in Lemma 3 and in later Lemmas 6 and 10 when we have the suffix  $v^{\omega} = \epsilon$ , as aforementioned. Therefore, it suffices to consider only proper languages to get the languages  $\Sigma^{\omega}$  and  $\Sigma^{\omega} \setminus \mathcal{L}(\mathcal{A})$  according to [18]. By Lemma 3, the congruence relation  $\sim$  allows us to obtain  $\mathcal{L}(\mathcal{A})$  (resp., the complementary language of  $\mathcal{L}(\mathcal{A})$ ) by identifying the exact set of proper languages that are inside  $\mathcal{L}(\mathcal{A})$  (resp., outside  $\mathcal{L}(\mathcal{A})$ ), thus being called saturating  $\mathcal{L}(\mathcal{A})$  (resp., the complementary language of  $\mathcal{L}(\mathcal{A})$ ).

In the remainder of the paper, we show that we can obtain similar saturation lemmas as Lemma 3 (cf. Lemmas 6 and 10) for the proposed congruence relations to obtain  $\mathcal{L}(\mathcal{A})$  or the complementary language of  $\mathcal{L}(\mathcal{A})$ .

### 3.2 Improved Congruence Relations for NBWs

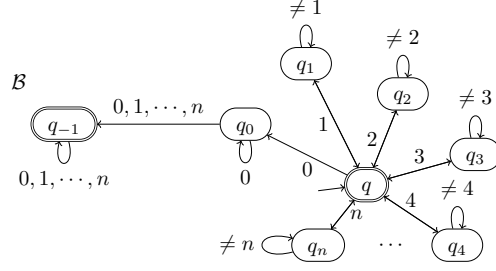
In this section, we present improved congruence relations that can be exponentially smaller than the classical congruence relation  $\sim$  (cf. Theorem 3), while our improved congruence relations can never be larger than  $\sim$  (cf. Theorem 2). Notably, our improved congruence relations only lead to a blow-up of  $\mathcal{O}(n^2)$  when  $\mathcal{A}$  is a DBW (cf. Theorem 4). We improve the classical congruence relation  $\sim$  in Sect. 3.1 based on following key observations: (1) We can use different congruence relations to process the finite prefix  $u$  and the periodic word  $v$  out of a UP-word  $uv^{\omega}$ , separately, in a manner similar to FDFWs. (2) The assumption  $[v]_{\sim}[v]_{\sim} \subseteq [v]_{\sim}$  in proper languages can further be left out in Lemma 3 according to [18]. (3) Inspired by [5], we can consider only reachable states from the initial states of  $\mathcal{A}$ , which allows us to relax the congruence relation  $\sim$  to *right congruence* relations. We defer the comparison of our work and [5] to Remark 2 in Sect. 4.

Instead of considering every pair of states  $(q, r)$  of  $\mathcal{A}$  to define congruence relation  $\sim$  in Definition 5, we can process the finite prefixes  $u$  by a simple subset construction over the states of  $\mathcal{A}$ , as shown with the improved right congruence relation  $\sim^i$  below.

**Definition 7.** For  $u_1, u_2 \in \Sigma^*$ , we say  $u_1 \sim^i u_2$  iff  $\delta(I, u_1) = \delta(I, u_2)$ .

**Lemma 4.** Let  $\sim^i$  be the right congruence in Definition 7. Then  $|\sim^i| \leq 2^n$ .

Thus  $\sim^i$  is a right congruence of finite index. We use  $\sim^i$  to process the finite prefix  $u$  of a UP-word  $uv^{\omega}$  and we define a new right congruence relation  $\approx_u$  for identifying a valid periodic finite word  $v$  for  $u$ , by considering only states reachable from  $\delta(I, u)$ .



**Fig. 2.** The family of NBWs  $B_n$  over the alphabet  $\{0, 1, \dots, n\}$  for which  $|\succ|$  is at least  $n!$  while  $|\approx_u|$  is at most  $2 \times (n + 3)$  for each  $u \in \Sigma^*$ .

**Definition 8.** For  $u \in \Sigma^*$ ,  $v_1, v_2 \in \Sigma^*$ , we say  $v_1 \approx_u v_2$  iff for all states  $q \in \delta(I, u)$  and  $r$  of  $\mathcal{A}$ , (1)  $q \xrightarrow{v_1} r$  iff  $q \xrightarrow{v_2} r$ ; and (2)  $q \xRightarrow{v_1} r$  iff  $q \xRightarrow{v_2} r$ .

Compared to Definition 5, we only take into account the states that can be reached from  $\delta(I, u)$ , as opposed to the whole set  $Q$ . In this way we can obtain a more compact right congruence relation than  $\sim$  for the periodic finite words, as proved later in Theorem 2.

**Lemma 5.** For each  $u \in \Sigma^*$ , let  $\approx_u$  be the right congruence defined in Definition 8. Then  $|\approx_u| \leq 3^{n^2}$ .

Although the right congruence relation  $\approx_u$  has the same worst-case complexity as its predecessor  $\sim$ , we show below that it is a more compact one.

**Theorem 2.** Let  $\sim$  be the congruence relation defined in Definition 5. For all  $u, v_1, v_2 \in \Sigma^*$ , we have that  $v_1 \sim v_2$  implies that  $v_1 \approx_u v_2$ .

Similarly,  $u_1 \sim u_2$  implies that  $u_1 \sim^i u_2$  for all  $u_1, u_2 \in \Sigma^*$ .

*Proof.* Assume that  $v_1 \sim v_2$ . For each pair of states  $q \in \delta(I, u)$  and  $r \in Q$ , if  $q \xrightarrow{v_1} r$ , we also have  $q \xrightarrow{v_2} r$  by Definition 5, since  $v_1 \sim v_2$ . Analogously,  $q \xRightarrow{v_1} r$  implies that  $q \xRightarrow{v_2} r$ . Similarly, we can prove that  $u_1 \sim u_2$  implies that  $u_1 \sim^i u_2$  for all  $u_1, u_2 \in \Sigma^*$ .

Further, we show that for each  $u \in \Sigma^*$ ,  $|\approx_u|$  can be exponentially smaller than  $|\sim|$  over the family of NBWs  $\mathcal{B}_n$  depicted in Figure 2.

**Theorem 3.** Let  $\sim$  be the congruence relation in Definition 5 and  $\approx_u$  the right congruence in Definition 8 with  $u \in \Sigma^*$ . There is a family of NBWs  $\mathcal{B}_1, \dots, \mathcal{B}_n$  with  $n + 3$  states for which  $|\sim| \geq n!$  and  $|\approx_u| \leq 2 \times (n + 3)$  for each  $u \in \Sigma^*$ .

*Proof (Proof sketch).* The family of NBWs  $\mathcal{B}_n$ , inspired from [4], is depicted in Figure 2 with  $n + 3$  states. Let  $\delta_n$  be the transition function of  $\mathcal{B}_n$ . We can see from Figure 2 that the initial state is  $q$  and  $F = \{q, q_{-1}\}$ .



Here we only show that  $|\approx_\epsilon|$  is at most  $2 \times (n+3)$ . The cases for other  $u \in \Sigma^*$  are analogous and we refer to Appendix A for detailed proof. By Definition 7, the equivalence class  $[e]_{\sim^i}$  can be encoded as  $\delta_n(q, \epsilon) = \{q\}$ . For each  $v \in \Sigma^+$  of  $[v]_{\approx_\epsilon}$ , if  $v$  does not contain 0, then there is at most one state  $r \in \{q, q_1, \dots, q_n\}$  such that  $q \xrightarrow{v} r$ . We may also have  $q \xRightarrow{v} r$ . Therefore, the number of possible  $[v]_{\approx_\epsilon}$  is at most 2. Otherwise if  $v$  contains 0, the number of possible  $r$  is at most 2, i.e.,  $r \in \{q_0, q_{-1}\}$ . Then we either have  $\{q \xrightarrow{v} q_{-1}, q \xrightarrow{v} q_{-1}, q \xrightarrow{v} q_0\}$  or just  $\{q \xrightarrow{v} q_0, q \xrightarrow{v} q_0\}$ . Therefore the number of possible  $[v]_{\approx_\epsilon}$  is also at most 2. Since the number of possible sets of reachable states from  $\{q\}$  over  $v$  is  $n+3$ , i.e.,  $\{q\}, \{q_1\}, \dots, \{q_n\}, \{q_0\}, \{q_0, q_{-1}\}$ , we then have  $|\approx_\epsilon| \leq 2 \times (n+3)$ .

To prove that  $|\sim| \geq n!$ , one can just show that for each pair of different permutation words  $u = i_1, \dots, i_n$  and  $u' = i'_1, \dots, i'_n$  of  $\{1, \dots, n\}$ ,  $u \not\sim u'$ . We refer the reader to Appendix A for detailed proof.

We further show that the size of the right congruence relations  $\bigcup_{u \in \Sigma^*} \{\approx_u\}$  can be improved exponentially if  $\mathcal{A}$  is a DBW.

**Theorem 4.** *Let  $\mathcal{A}$  be a DBW with  $n$  states. Then  $\Sigma_{[u]_{\sim^i} \in \Sigma^* / \sim^i} |\approx_u|$  is in  $\mathcal{O}(n^2)$ .*

*Proof.* Let  $\delta$ ,  $s$  and  $Q$  be the transition function, the initial state and the set of states of  $\mathcal{A}$ , respectively. Since an equivalence class  $[u]_{\sim^i}$  can be uniquely encoded as  $\delta(s, u)$ , the number of equivalence classes of  $\sim^i$  is  $n$  since  $\delta$  is deterministic. For an equivalence class  $[x]_{\sim^i}$ , an equivalence class  $[v]_{\approx_u}$  can be uniquely encoded as a set  $S$  of the reachability between states in  $\{q\}$  and in  $Q$  where  $q = \delta(s, u)$ . Since  $\delta$  is deterministic, we have that  $|\delta(q, v)| = 1$  and there are at most 2 pairs of states in  $S$ . This is because that there is at most one state  $r$  in  $q \xrightarrow{v} r$  once  $v$  is given. W.l.o.g, for an equivalence class  $[v]_{\approx_u}$ , we assume that  $r = \delta(q, v)$ . Then, we have that  $S = \{q \xrightarrow{v} r\}$  or  $S = \{q \xrightarrow{v} r, q \xRightarrow{v} r\}$ . It follows that the number of possible  $S$  for the equivalence class  $[v]_{\approx_u}$  is at most 2. Therefore, the number of equivalence classes of  $\approx_u$  is  $2n$ , as the number of possible  $q$  is  $n$ . Then  $\Sigma_{[u]_{\sim^i} \in \Sigma^* / \sim^i} |\approx_u| \leq n \times 2n \in \mathcal{O}(n^2)$ .

For congruence relations to recognize *exactly*  $\mathcal{L}(\mathcal{A})$  or its complement  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A})$ , we now propose a saturation lemma for  $\sim^i$  and  $\approx_u$  with  $u \in \Sigma^\omega$ , similarly to Lemma 3. The  $\omega$ -language  $[u]_{\sim^i} [v]_{\approx_u}^\omega$  with  $uv \sim^i u$  is either completely inside  $\mathcal{L}(\mathcal{A})$  or outside  $\mathcal{L}(\mathcal{A})$ ; the key difference between Lemma 6 and Lemma 3 is that we do not require  $[v]_{\approx_u} [v]_{\approx_u} \subseteq [v]_{\approx_u}$ .

**Lemma 6 (Saturation Lemma).**

1. For  $u \in \Sigma^*, v \in \Sigma^+$ , if  $uv \sim^i u$ , then either  $[u]_{\sim^i} [v]_{\approx_u}^\omega \cap \mathcal{L}(\mathcal{A}) = \emptyset$  or  $[u]_{\sim^i} [v]_{\approx_u}^\omega \subseteq \mathcal{L}(\mathcal{A})$ .
2.  $\Sigma^\omega = \bigcup \{ [u]_{\sim^i} [v]_{\approx_u}^\omega \mid u \in \Sigma^*, v \in \Sigma^+, uv \sim^i u \}$ .
3.  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}) = \bigcup \{ [u]_{\sim^i} [v]_{\approx_u}^\omega \mid u \in \Sigma^*, v \in \Sigma^+, uv \sim^i u, [u]_{\sim^i} [v]_{\approx_u}^\omega \cap \mathcal{L}(\mathcal{A}) = \emptyset \}$ .

*Proof.* We first prove Item (1). Assume that  $uv \sim^i u$ . By Definition 7,  $\delta(I, u) = \delta(I, uv)$ . By Definition 8, for each word  $v' \in [v]_{\approx_u}$ , all states  $q \in \delta(I, u)$  and  $r \in Q$ ,  $q \xrightarrow{v} r$  implies  $q \xrightarrow{v'} r$ . Then,  $\delta(I, uv') = \delta(I, uv) = \delta(I, u)$  for each  $v' \in [v]_{\approx_u}$ . It follows that  $[u]_{\sim^i} [v]_{\approx_u} = [u]_{\sim^i}$ . Assume that there exists a word  $w \in [u]_{\sim^i} [v]_{\approx_u}^\omega \cap \mathcal{L}(\mathcal{A})$ . We can then decompose  $w$  as  $w = u_0 \cdot v_1 \cdot v_2 \cdots$  with  $u_0 \in [u]_{\sim^i}$  and  $v_i \in [v]_{\approx_u}$  for each  $i \geq 1$ . It follows that  $\delta(I, u_0) = \delta(I, u_0 v_i) = \delta(I, u_0 v_1 \cdots v_i)$  for each  $i \geq 1$  since  $u_0 v_i \sim^i u_0$  for each  $i \geq 1$ . Therefore, the set of states  $\delta(I, u_0) = \delta(I, u)$  has been visited for infinitely many times and they can be reachable by themselves. Since  $w \in \mathcal{L}(\mathcal{A})$ , there exists an *execution* of  $\mathcal{A}$  over  $w$  written as  $\rho_{u_0} = q \xrightarrow{u_0} q_0 \xrightarrow{v_1} q_1 \xrightarrow{v_2} q_2 \cdots$  where  $q \in I, q_0 \in \delta(q, u_0)$  and  $q_i \in \delta(I, u_0 v_i)$  for each  $i \geq 0$ . Moreover, the corresponding run  $\hat{\rho}_{u_0} = q \cdots q_0 \cdots q_1 \cdots q_2 \cdots$  of  $\rho_{u_0}$  is accepting since  $w \in \mathcal{L}(\mathcal{A})$ . By Definition 7, we then also have an execution  $\rho_u = p \xrightarrow{u} q_0 \xrightarrow{v_1} q_1 \xrightarrow{v_2} q_2 \cdots$  for some  $p \in I$  such that  $p \xrightarrow{u} q_0$ , since  $q_0 \in \delta(I, u)$ . Let  $w' = u'_0 v'_1 v'_2 \cdots$  with  $u'_0 \in [u]_{\sim^i}$  and  $v'_i \in [v]_{\approx_u}$  for each  $i \geq 1$ . By Definition 8, one execution of  $\mathcal{A}$  over  $u \cdot v'_1 \cdot v'_2 \cdots$  is also  $\rho_u$ , since  $q_{i-1} \xrightarrow{v'_i} q_i$ , as  $v_i \approx_u v'_i$  for each  $k \geq 1$  and  $q_i \in \delta(I, u)$  for each  $k \geq 0$ . Since  $u'_0 \in [u]_{\sim^i}$ , there exists a state  $q' \in I$  such that  $q' \xrightarrow{u'_0} q_0$ . It follows that  $\hat{\rho}_{u'_0} = q' q_0 q_1 \cdots$  is a run of  $\mathcal{A}$  over  $w'$ . Clearly,  $\hat{\rho}_{u'_0}$  is an accepting run of  $\mathcal{A}$ . Therefore,  $w'$  is also accepted by  $\mathcal{A}$ . It follows that  $[u]_{\sim^i} [v]_{\approx_u}^\omega \subseteq \mathcal{L}(\mathcal{A})$  if  $[u]_{\sim^i} [v]_{\approx_u}^\omega \cap \mathcal{L}(\mathcal{A}) \neq \emptyset$ .

To prove Item (2), we let  $w \in \text{UP}(\Sigma^\omega)$ . Let  $(u', v')$  be a decomposition of  $w$  and  $\mathcal{M}$  the DFW induced from  $\sim^i$  with Definition 1. According to [3], there exist  $u = u'v'^h$  and  $v = v'^k$ , where  $h, k \geq 1$  such that  $\mathcal{M}(u) = \mathcal{M}(uv)$ , i.e.,  $(u, v)$  is a normalized decomposition of  $w$ . Intuitively, since  $\mathcal{M}$  is deterministic and has finite number of states,  $\mathcal{M}$  visits a state, say  $\mathcal{M}(u)$ , from  $\mathcal{M}(u')$  twice after inputting enough number of the word  $v'$ . It follows that there exist integers  $h, k \geq 1$  such that  $\mathcal{M}(u) = \mathcal{M}(u'v'^h) = \mathcal{M}(u'v'^{h+k})$ . W.l.o.g., let  $[u]_{\sim^i} = \mathcal{M}(u)$ . Since  $\mathcal{M}(u) = \mathcal{M}(uv)$ , we have that  $u \sim^i uv$ . It follows that we have an  $\omega$ -language  $[u]_{\sim^i} [v]_{\approx_u}^\omega$  with  $uv \sim^i u$  such that  $w \in [u]_{\sim^i} [v]_{\approx_u}^\omega$  for each  $\omega$ -word  $w \in \text{UP}(\Sigma^\omega)$ . Therefore  $\Sigma^\omega = \bigcup \{ [u]_{\sim^i} [v]_{\approx_u}^\omega \mid u \in \Sigma^*, v \in \Sigma^+, uv \sim^i u \}$ .

Item (3) is directly follows from Items (1) and (2).

*Discussion on theory vs. practice.* In Section 4, we will introduce congruence relations with a blow-up of only  $2^{\mathcal{O}(n \log n)}$ , which is tighter than  $3^{\mathcal{O}(n^2)}$  obtained here (cf. Lemma 5). We argue that a lower complexity of an algorithm, however, does *not* necessarily indicate a better performance on all cases in practice, as witnessed in [5, 11, 19]. We believe that the improved congruence relations can still be a good complement to the right congruences proposed in Section 4. Thus we keep them in this paper.

## 4 Optimal Congruence Relations for NBWs

In Sect. 3.2, the improved congruence relations still lead to a blow-up of  $3^{\mathcal{O}(n^2)}$ . Inspired by [9, 10], we reduce the blow-up to  $2^{\mathcal{O}(n \log n)}$  by introducing a *preorder* on the states based on the transition structure of  $\mathcal{A}$ . Breuers *et al.* also proposed

in [5] a preorder-based optimization to improve RBC and we defer the comparison of this work with it to Remark 2.

Let  $R$  be a set: a preorder  $\preceq$  over  $R$  is a *binary* relation that is reflexive ( $r \preceq r$  for all  $r \in R$ ) and transitive (if  $a \preceq b$  and  $b \preceq c$  then  $a \preceq c$  for all  $a, b, c \in R$ ). In particular,  $r \prec r'$  if  $r \preceq r'$  and  $r' \not\preceq r$ . The equivalence class where  $r \in R$  belongs to under  $\preceq$ , denoted as  $[r]_{\preceq}$ , is  $\{r' \in R \mid r' \preceq r, r \preceq r'\}$ . Let  $S \subseteq R$  be a *nonempty* set and  $\preceq$  a preorder over  $R$ : we denote by  $S/\preceq$  the set of equivalence classes of  $S$  under the preorder  $\preceq$ ; we then define the maximal equivalence class of  $S$  under  $\preceq$  as  $\max_{\preceq}(S) = \max(S/\preceq) = \{r' \in S \mid r \preceq r' \text{ for all } r \in S\}$ . As usual, the equivalence classes in  $S/\preceq$  can be ordered by  $[r]_{\preceq} \preceq [r']_{\preceq} \iff r \preceq r'$  and we have  $[r]_{\preceq} \prec [r']_{\preceq}$  if  $[r]_{\preceq} \preceq [r']_{\preceq}$  and  $[r']_{\preceq} \not\preceq [r]_{\preceq}$ . We remark that  $\preceq$  is a preorder not an equivalence relation.

Let  $\mathcal{R}$  be the set of possible preordered subsets of  $Q$ . According to [10], the number of possible preordered subsets of  $Q$  under a preorder  $\preceq$  is  $\mathcal{O}((\frac{n}{\epsilon \ln n})^n) \approx (0.53n)^n \leq n^n$ . Inspired by [9, 10], one can define a preorder  $\preceq$  and a transition function  $\phi$  between two preordered subsets of  $Q$  based on the transition structure of  $\mathcal{A}$  as below.

**Definition 9.** For the set of initial states  $I \subseteq Q$ , we define the ordered sets  $I/\preceq$  such that for all states  $p, q \in I$ ,  $[p]_{\preceq} \prec [q]_{\preceq}$  if  $p \notin F$  and  $q \in F$ , otherwise we have  $[q]_{\preceq} \preceq [p]_{\preceq}$ .

Let  $P/\preceq \in \mathcal{R}$  be ordered sets of  $P \subseteq Q$ ,  $a \in \Sigma$  a letter and  $P' = \delta(P, a)$ . We define the ordered  $a$ -successor  $P'/\preceq$  of  $P/\preceq$ , denoted by  $\phi(P/\preceq, a)$ , as follows. For all states  $p', q' \in P'$ ,

- $[p']_{\preceq} \prec [q']_{\preceq}$  if  $\max_{\preceq}\{p \in P \mid p \xrightarrow{a} p'\} \prec \max_{\preceq}\{q \in P \mid q \xrightarrow{a} q'\}$ ;
- Assume that  $\max_{\preceq}\{p \in P \mid p \xrightarrow{a} p'\} = \max_{\preceq}\{q \in P \mid q \xrightarrow{a} q'\}$ :
  - $[p']_{\preceq} \preceq [q']_{\preceq}$  if  $p' \notin F$ , and
  - $[p']_{\preceq} \prec [q']_{\preceq}$  if  $q' \in F$  and  $p' \notin F$ .

We extend  $\phi$  to words, by letting  $\phi(P/\preceq, \epsilon) = P/\preceq$  and  $\phi(P/\preceq, a_1 \cdots a_k) = \phi(\phi(P/\preceq, a_1), \cdots, a_k)$  where  $k \geq 1$  and  $a_i \in \Sigma$  for  $i \in \{1, \cdots, k\}$ .

In particular, we have  $[p']_{\preceq} = [q']_{\preceq}$  if  $p' \notin F$  under the assumption that  $\max_{\preceq}\{p \in P \mid p \xrightarrow{a} p'\} = \max_{\preceq}\{q \in P \mid q \xrightarrow{a} q'\}$  and  $q' \notin F$ . Intuitively, accepting states have a higher priority than nonaccepting states and so do their respective successors. We denote by  $[p]_{\preceq} \in P/\preceq$  when the equivalence class  $[p]_{\preceq}$  belongs to the ordered set  $P/\preceq$  with  $p \in P$ .

*Example 1.* Take the NBW  $B_n$  depicted in Figure 2 for example. At first we have only one equivalence class  $\{q\}$  as there is only one initial state. So  $I/\preceq = \langle \{q\} \rangle$ . The position of an equivalence class in an preordered set indicates its order and the rightmost equivalence class is the maximal equivalence class. After inputting a word  $00$ , we first reach the preordered set  $\langle \{q_0\} \rangle$  and then  $\langle \{q_0\}, \{q_{-1}\} \rangle$ . Both  $q_0$  and  $q_{-1}$  are 0-successors of the equivalence class  $\{q_0\}$ . Since by definition accepting states have higher priority than nonaccepting states, we have  $\phi(\langle \{q_0\} \rangle) = \langle \{q_0\}, \{q_{-1}\} \rangle$ , i.e., we have  $q_0 \prec q_{-1}$  in  $\delta(\{q_0\}, a)$ .

Further, one can see that  $\phi(\langle\{q_0\}, \{q_{-1}\}\rangle, a) = \langle\{q_0\}, \{q_{-1}\}\rangle$ . Here  $q_{-1}$  has two 0-predecessors, namely  $q_0$  and  $q_{-1}$ . By definition,  $q_{-1}$  should inherit the order of the predecessor in the maximal equivalence class compared to others. Since  $\{q_0\} \prec \{q_{-1}\}$ , we then put  $q_{-1}$  in the rightmost equivalence class when computing  $\phi(\langle\{q_0\}, \{q_{-1}\}\rangle, a)$ .

*Remark 1.* The preorder  $\preceq$  in Definition 9 is closely related to the *lexicographical order* of vertices at the same level of the run direct acyclic graph (DAG) over an  $\omega$ -word  $w$  in [10], which is constructed such that it has a path visiting vertices marked with accepting states infinitely often iff  $w$  is accepted by  $\mathcal{A}$ . In a run DAG of [10] which we call *ordered run DAG*, each vertex at a level is marked with a nonempty set of states of  $\mathcal{A}$ , which corresponds to an equivalence class under  $\preceq$ . For instance, if  $I \cap F \neq \emptyset$  and  $I \setminus F \neq \emptyset$ , then there are two ordered vertices at level 0 marked with  $I \setminus F$  and  $I \cap F$ , respectively. This order is also reflected by the fact that  $I \setminus F \prec I \cap F$  in Definition 9. Constructing a next level there corresponds to computing successors of ordered sets under  $\preceq$ . Detailed explanation on the mapping between Definition 9 and the ordered run DAG can be found in Appendix C.2.

Instead of only tracing the set of reachable states with the congruence relation  $\sim^i$ , we also equip the reachable states with the preorder  $\preceq$ , which is formalized below with the right congruence  $\sim^o$ .

**Definition 10.** For  $u_1, u_2 \in \Sigma^*$ , we say  $u_1 \sim^o u_2$  iff  $\phi(I/\preceq, u_1) = \phi(I/\preceq, u_2)$ .

Since each equivalence class  $[u]_{\sim^o}$  with  $u \in \Sigma^*$  can be uniquely encoded as the ordered sets  $\phi(I/\preceq, u)$ , we have the following upper bound of  $\sim^o$  according to [10].

**Lemma 7.** Let  $\sim^o$  be the right congruence in Definition 10. Then  $|\sim^o| \leq n^n$ .

**Lemma 8.** For all  $u_1, u_2 \in \Sigma^*$ , if  $u_1 \sim^o u_2$ , then  $u_1 \sim^i u_2$ .

According to Lemma 8, one can see that  $\sim^i$  is more compact than  $\sim^o$ . Nonetheless, we show that the right congruence  $\sim^o$  allows us to define a novel right congruence relation  $\approx_u^o$  of size  $2^{\mathcal{O}(n \log n)}$  for a given  $u \in \Sigma^*$ .

**Definition 11.** For  $u \in \Sigma^*$ ,  $v_1, v_2 \in \Sigma^*$ , we say  $v_1 \approx_u^o v_2$  if and only if (1)  $uv_1 \sim^o uv_2$  and then (2) for all states  $q \in P$  where  $P = \delta(I, uv_1) = \delta(I, uv_2)$ ,  $\max_{\preceq} \{ [p]_{\preceq} \in \phi(I/\preceq, u) \mid p \xrightarrow{v_1} q \} = \max_{\preceq} \{ [p]_{\preceq} \in \phi(I/\preceq, u) \mid p \xrightarrow{v_2} q \}$ .

If  $uv_1 \sim^o uv_2$ , we have  $\delta(I, uv_1) = \delta(I, uv_2)$  by Definition 10. Moreover, in addition to  $uv_1 \sim^o uv_2$ , the maximal equivalence class in  $\phi(I/\preceq, u)$  that reaches a state  $q \in P$  over  $v_1$  has to be the same maximal equivalence class in  $\phi(I/\preceq, u)$  that reaches  $q$  over  $v_2$ . Below we prove that the index of  $\approx_u^o$  is in  $2^{\mathcal{O}(n \log n)}$ .

**Lemma 9.** For each  $u \in \Sigma^*$ , let  $\approx_u^o$  be the right congruence defined in Definition 11. Then  $|\approx_u^o| \leq n^n \times (n+1)^n$ .

*Proof.* First,  $u$  is fixed. Let  $v \in \Sigma^*$ . Each equivalence class of  $[v]_{\approx_u^o}$  can be uniquely encoded as a pair  $(\phi(I/\preceq, uv), f)$  where  $f$  is a function mapping a state in  $Q$  to an equivalence class in  $\phi(I/\preceq, u)$  or an empty set that indicates the state is not present in  $\delta(I, uv)$ . Since  $\phi(I/\preceq, u)$  is given and there are at most  $n$  equivalence classes in  $\phi(I/\preceq, u)$ , the number of possible functions  $f$  is at most  $(n+1)^n$ . It is then clear that there are at most  $n^n \times (n+1)^n$  equivalence classes defined by  $\approx_u^o$  since the number of possible  $\phi(I/\preceq, uv)$  is  $n^n$ .

Similarly to Lemmas 3 and 6, we give following saturation lemma to recognize exactly  $\mathcal{L}(\mathcal{A})$  and  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A})$ .

**Lemma 10 (Saturation Lemma).**

1. For  $u \in \Sigma^*, v \in \Sigma^+$ , if  $uv \sim^o u$ , then either  $[u]_{\sim^o} [v]_{\approx_u^o} \cap \mathcal{L}(\mathcal{A}) = \emptyset$  or  $[u]_{\sim^o} [v]_{\approx_u^o} \subseteq \mathcal{L}(\mathcal{A})$ .
2.  $\Sigma^\omega = \bigcup \{ [u]_{\sim^o} [v]_{\approx_u^o} \mid u \in \Sigma^*, v \in \Sigma^+, uv \sim^o u \}$ .
3.  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}) = \bigcup \{ [u]_{\sim^o} [v]_{\approx_u^o} \mid u \in \Sigma^*, v \in \Sigma^+, uv \sim^o u, [u]_{\sim^o} [v]_{\approx_u^o} \cap \mathcal{L}(\mathcal{A}) = \emptyset \}$ .

*Proof (Proof Sketch).* Here we only prove Item (1). The Items (2) and (3) can be proved similarly as in the proof of Lemma 6.

Assume that  $uv \sim^o u$ . Similarly to the proof of Lemma 6,  $[u]_{\sim^o} [v]_{\approx_u^o} = [u]_{\sim^o}$ . Let  $w \in [u]_{\sim^o} [v]_{\approx_u^o} \cap \mathcal{L}(\mathcal{A})$  such that  $w = u_0 \cdot v_1 \cdot v_2 \cdots$  with  $u_0 \in [u]_{\sim^o}$  and  $v_i \in [v]_{\approx_u^o}$  for each  $i \geq 1$ . It follows that  $u \sim^o u_0 v_i \sim^o u_0 v_1 \cdots v_i$  for each  $i \geq 1$  since  $u_0 v_i \sim^o u_0 \sim^o u$  for each  $i \geq 1$ . Then  $\phi(I/\preceq, u) = \phi(I/\preceq, uv_i) = \phi(I/\preceq, uv_1 \cdots v_i)$  for every  $i \geq 1$  since  $u \sim^o u_0 v_i \sim^o u_0 v_1 \cdots v_i$  for each  $i \geq 1$  and  $\phi$  is deterministic. Therefore, the ordered sets  $\phi(I/\preceq, u_0) = \phi(I/\preceq, u) = \phi(I/\preceq, uv_i)$  have been visited for infinitely many times and they are reachable by themselves in the same way on  $v' \in [v]_{\approx_u^o}$  (cf. Item (2) in Definition 11). One can construct an ordered run DAG  $G_{w, \mathcal{A}}$  of  $\mathcal{A}$  over  $w = u_0 \cdot v_1 \cdot v_2 \cdots$  with Definition 9 (cf. Appendix C.2 for details). Therefore, from the ordered run DAG  $G_{w, \mathcal{A}}$ , there exists an *execution* of  $\mathcal{A}$  over  $w$  written as  $\rho_{u_0} = q \xrightarrow{u_0} q_0 \xrightarrow{v_1} q_1 \xrightarrow{v_2} q_2 \cdots$  where  $q \in I, [q_0]_{\preceq} \in \phi(I/\preceq, u_0), [q_i]_{\preceq} \in \phi(I/\preceq, u_0 v_i)$ , and  $[q_i]_{\preceq}$  is the maximal equivalence class in  $\phi(I/\preceq, u_0)$  that reaches the state  $q_{i+1}$  for each  $i \geq 0$ . Moreover, there exists an execution visiting infinitely many accepting states when  $w \in \mathcal{L}(\mathcal{A})$ . Let  $w' = u'_0 v'_1 v'_2 \cdots$  with  $u'_0 \in [u]_{\sim^o}$  and  $v'_i \in [v]_{\approx_u^o}$  for each  $i \geq 1$ . Similarly to the proof of Lemma 6, adapted to notions in Definition 11, we can also construct such an execution that visits accepting states infinitely many times for  $w'$ . Therefore,  $w$  is accepting if and only if  $w'$  is accepting. It follows that  $[u]_{\sim^o} [v]_{\approx_u^o} \subseteq \mathcal{L}(\mathcal{A})$  if  $[u]_{\sim^o} [v]_{\approx_u^o} \cap \mathcal{L}(\mathcal{A}) \neq \emptyset$ . We note that detailed construction of the ordered run DAG is provided in Appendix C.2.

*Remark 2.* Sistla *et al.* in [18] construct an NBW  $\mathcal{B}_{u,v}$ , for each proper language  $Y_{u,v} = [u]_{\sim} [v]_{\approx}$  such that  $Y_{u,v} \cap \mathcal{L}(\mathcal{A}) = \emptyset$ . Each  $\mathcal{B}_{u,v}$  can be constructed with two copies of DFWs  $\mathcal{M}[\sim]$  induced with  $\sim$  (cf. Definition 1) where the first  $\mathcal{M}$  processes the finite prefix  $u$  and the second  $\mathcal{M}$  is modified to accept the word  $v^\omega$ . According to [18], we obtain an NBW  $\mathcal{A}^c$  with  $3^{\mathcal{O}(n^2)}$  states. Further,

Breuers *et al.* in [5] also proposed a subset construction for improving the RBC for complementing NBWs to process the finite prefix  $u$  of a UP-word  $uv^\omega$  in  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A})$ ; nevertheless, they still used the classical congruence relation  $\sim$  for recognizing the periodic word  $v$  of  $uv^\omega$ . Different from the algorithms in [5, 18], we exploit the right congruence  $\approx_u$  or  $\approx_u^o$  instead of  $\sim$  for accepting the periodic word  $v$  of  $uv^\omega$ . With a preorder-based optimization, the part for accepting  $v$  in [5] has also been reduced to  $2^{\mathcal{O}(n \log n)}$ . However, this optimization uses more than one automaton for recognizing a periodic word  $v$  for each  $u$ ; in contrast, we only need one as each equivalence class  $[u]_{\sim^o}$  of  $\sim^o$  only relates with one right congruence relation  $\approx_u^o$ , thus yielding an FDFW in Section 5.

We formalize the main result of this section as Theorem 5 without proof since Theorem 5 is a direct consequence of Theorem 8 given later in Section 5.

**Theorem 5.** *The right congruence relations  $(\sim^o, \bigcup_{u \in \Sigma^*} \{\approx_u^o\})$  are asymptotically optimal.*

## 5 Connection to FDFWs

In this section, we reveal a deep connection between the congruence relations of NBWs and FDFWs, i.e., one can construct an FDFW  $\mathcal{F}$  with *optimal* complexity that accepts  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A})$  from the congruence relations  $(\sim^o, \bigcup_{u \in \Sigma^*} \{\approx_u^o\})$  proposed in this work. As a byproduct of this connection, we are able to prove Theorem 5 in Section 4; in other words, one cannot find congruence relations for NBWs of size less than  $2^{\mathcal{O}(n \log n)}$ .

We now introduce the construction of FDFWs from the right congruences. Since  $\sim^o$  (resp.,  $\sim^i$ ) and  $\approx_u^o$  (resp.,  $\approx_u$ ) with  $u \in \Sigma^*$  are right congruences of finite index, they can be used to define the transition structures of an FDFW  $\mathcal{F}$  with Definition 1 recognizing  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A})$ . Moreover, by Lemma 10 (resp., Lemma 6), we can identify the accepting macrostates of the progress DFWs. We now give a construction of an FDFW  $\mathcal{F}$  with  $\sim^o$  and  $\approx_u^o$ . We note that the construction of an FDFW with  $\sim^i$  and  $\approx_u$  is similar.

**Definition 12.** *The FDFW  $\mathcal{F}$  is a tuple  $(\mathcal{M}[\sim^o], \{\mathcal{N}_u[\approx_u^o]\})$  where*

- $\mathcal{M}[\sim^o]$  is the DFW without accepting macrostates induced by  $\sim^o$  of Definition 10 with Definition 1.
- For each macrostate  $[u]_{\sim^o}$  of  $\mathcal{M}[\sim^o]$ , the progress DFW  $\mathcal{N}_u[\approx_u^o]$  is induced by  $\approx_u^o$  of Definition 11 with Definition 1 augmented with the accepting macrostates of  $\mathcal{N}_u[\approx_u^o]$  are the equivalence classes  $[v]_{\approx_u^o}$  of  $\approx_u^o$  for which  $uv^\omega \notin \mathcal{L}(\mathcal{A})$ ,  $uv \sim^o u$ .

**Theorem 6.** *Let  $\mathcal{F}$  be the FDFW constructed from  $\mathcal{A}$  in Definition 12. Then (1)  $UP(\mathcal{F}) = UP(\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}))$ ; (2)  $\mathcal{F}$  is saturated; and (3)  $\mathcal{F}$  has  $2^{\mathcal{O}(n \log n)}$  macrostates.*

*Proof.* Here we only prove item (2); the complete proof for this theorem can be found in Appendix A. For two normalized decompositions  $(u_1, v_1)$  and  $(u_2, v_2)$  of  $w$ , we assume that  $(u_1, v_1)$  is accepted by  $\mathcal{F}$ . It follows that  $w = u_1 v_1^\omega = u_2 v_2^\omega$  is not accepted by  $\mathcal{A}$ . By Definition 12, let  $[u]_{\sim^\circ} = \mathcal{M}(u_2)$  and  $[v]_{\approx_u^\circ} = \mathcal{N}_u(v_2)$  where  $\mathcal{N}_u$  is the progress DFW of  $[u]_{\sim^\circ}$  of  $\mathcal{M}$ . It follows that  $u_2 \in [u]_{\sim^\circ}$  and  $v_2 \in [v]_{\approx_u^\circ}$ . According to the proof of Lemma 10, we have that  $uv \sim^\circ u$  if  $u_2 v_2 \sim^\circ u_2$ . By Lemma 10, we have  $[u]_{\sim^\circ} [v]_{\approx_u^\circ} \cap \mathcal{L}(\mathcal{A}) = \emptyset$  since  $u_2 v_2^\omega \in [u]_{\sim^\circ} [v]_{\approx_u^\circ}$  and  $uv \sim^\circ u$ . According to Definition 12,  $[v]_{\approx_u^\circ}$  is thus an accepting macrostate of  $\mathcal{N}_u$ . It follows that  $(u_2, v_2)$  is also accepted by  $\mathcal{F}$ . Thus  $\mathcal{F}$  is saturated.

We remark that saturated FDFWs can be complemented in linear time [3]. So we also can obtain an FDFW accepting  $\mathcal{L}(\mathcal{A})$  from  $\mathcal{F}$  easily.

Below we give the lower bound of the size of FDFWs constructed in Definition 12 from  $\mathcal{A}$ ; the proof idea of Theorem 7 is similar to the one for [3, Theorem 14] that concerns with UP-words of  $\mathcal{L}(\mathcal{A})$ .

**Theorem 7.** *There exists a family of NBWs  $\mathcal{A}_1, \dots, \mathcal{A}_n$  with  $n$  states for which an FDFW  $\mathcal{F}_n$  accepting  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}_n)$  has  $2^{\Omega(n \log n)}$  macrostates.*

The result of Theorem 7 is not surprising as there exists an NBW  $\mathcal{A}_n$  whose complementary NBW  $\mathcal{A}_n^c$  has  $2^{\Omega(n \log n)}$  states [22]. Thus an FDFW  $\mathcal{F}_n$  accepting  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}_n)$  with less than  $2^{n \log n}$  macrostates would lead to a contradiction to the result in [22] since one can obtain  $\mathcal{A}_n^c$  from  $\mathcal{F}_n$  with a polynomial blow-up (cf. Lemma 1).

Finally, as the main result of this section, we show below that the construction of FDFWs from  $\mathcal{A}$  via congruence relations  $(\sim^\circ, \bigcup_{u \in \Sigma^*} \{\approx_u^\circ\})$  in Definition 12 is asymptotically optimal, which directly follows from Theorem 6 and Theorem 7.

**Theorem 8.** *The construction of FDFWs in Definition 12 with the right congruence relations  $(\sim^\circ, \bigcup_{u \in \Sigma^*} \{\approx_u^\circ\})$  from  $\mathcal{A}$  is asymptotically optimal.*

Theorem 5 follows directly from Theorem 8 since the constructed FDFW has the same size of congruence relations  $(\sim^\circ, \bigcup_{u \in \Sigma^*} \{\approx_u^\circ\})$  according to Definition 12.

*Remark 3.* Here we discuss related works on FDFWs. There are polynomial-time translations from FDFWs to NBWs [3, 7]. The other direction of the translation is more difficult. The direct translations from an  $n$ -state NBW, proposed in [7] and in [12], produce an FDFW with  $\mathcal{O}(4^{n^2+n})$  states and an FDFW with  $\mathcal{O}(3^{n^2+n})$  states, respectively. Our construction in Definition 12 replaced with  $(\sim^i, \bigcup_{u \in \Sigma^*} \{\approx_u\})$  can even be exponentially better than the two translations above; due to lack of space, detailed reasoning can be found in Appendix B. The translation based on an intermediate determinization of NBWs to deterministic Parity automata in [3] yields an FDFW with the optimal complexity  $2^{\mathcal{O}(n \log n)}$ . Our construction in Definition 12, however, is the *first direct* and *optimal* translation from an NBW to an FDFW without involving determinization of NBWs. Given an  $\omega$ -regular language  $L$ , Angluin and Fisman in [4] directly operate on

the language  $L$  and give congruence relations for constructing FDFWs of  $L$ . In contrast, our work takes an NBW  $\mathcal{A}$  as input and defines congruence relations for recognizing  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A})$  based on the transition structure of  $\mathcal{A}$ .

## 6 Concluding Remarks

In this work, we have proposed more compact congruence relations than the classical congruence relation and further given asymptotically *optimal* right congruences for NBWs. Moreover, to the best of our knowledge, we give the *first direct* translation from an NBW to an FDFW with *optimal* complexity, based on the optimal right congruences for NBWs. We showed that congruence relations relate tightly the classical RBC and FDFWs. Congruence relations are known to be able to yield the minimal DFWs for given regular languages by the Myhill-Nerode Theorem, by identifying equivalent states. We conjecture that the resulting congruence relations above may enable the reduction of state space in the complementary automaton of  $\mathcal{A}$ . That is, similar subsumption and simulation techniques developed in [1, 2] for those congruence relations can be exploited to avoid exploration of redundant states in containment checking between NBWs. We leave this conjecture to future work.

## References

1. Abdulla, P.A., Chen, Y., Clemente, L., Holík, L., Hong, C., Mayr, R., Vojnar, T.: Simulation subsumption in Ramsey-based Büchi automata universality and inclusion testing. In: CAV. LNCS, vol. 6174, pp. 132–147. Springer (2010)
2. Abdulla, P.A., Chen, Y., Clemente, L., Holík, L., Hong, C., Mayr, R., Vojnar, T.: Advanced Ramsey-based Büchi automata inclusion testing. In: CONCUR. LNCS, vol. 6901, pp. 187–202. Springer (2011)
3. Angluin, D., Boker, U., Fisman, D.: Families of DFAs as Acceptors of  $\omega$ -Regular Languages. Logical Methods in Computer Science **14**(1) (2018)
4. Angluin, D., Fisman, D.: Learning regular omega languages. Theoretical Computer Science **650**, 57–72 (2016)
5. Breuers, S., Löding, C., Olschewski, J.: Improved Ramsey-based Büchi complementation. In: FOSSACS. pp. 150–164 (2012)
6. Büchi, J.R.: On a decision method in restricted second order arithmetic. In: Proc. Int. Congress on Logic, Method, and Philosophy of Science. 1960. pp. 1–12. Stanford University Press (1962)
7. Calbrix, H., Nivat, M., Podelski, A.: Ultimately periodic words of rational  $\omega$ -languages. In: MFPS. pp. 554–566. Springer (1993)
8. Clemente, L., Mayr, R.: Efficient reduction of nondeterministic automata with application to language inclusion testing. Logical Methods in Computer Science **15**(1) (2019)
9. Fogarty, S., Kupferman, O., Vardi, M.Y., Wilke, T.: Profile trees for Büchi word automata, with application to determinization. Inf. Comput. **245**, 136–151 (2015)
10. Fogarty, S., Kupferman, O., Wilke, T., Vardi, M.Y.: Unifying Büchi complementation constructions. Log. Methods Comput. Sci. **9**(1) (2013)



11. Fogarty, S., Vardi, M.Y.: Efficient büchi universality checking. In: Esparza, J., Majumdar, R. (eds.) TACAS. Lecture Notes in Computer Science, vol. 6015, pp. 205–220. Springer (2010)
12. Kuperberg, D., Pinault, L., Pous, D.: Coinductive algorithms for Büchi automata. In: DLT. pp. 206–220. Springer International Publishing, Cham (2019)
13. Kupferman, O., Vardi, M.Y.: Verification of fair transisiton systems. In: Alur, R., Henzinger, T.A. (eds.) CAV. LNCS, vol. 1102, pp. 372–382. Springer (1996)
14. Kupferman, O., Vardi, M.Y.: Model checking of safety properties. *Formal Methods in System Design* **19**(3), 291–314 (2001)
15. Li, Y., Chen, Y.F., Zhang, L., Liu, D.: A novel learning algorithm for Büchi automata based on family of DFAs and classification trees. In: TACAS. pp. 208–226. Springer (2017)
16. Myhill, J.: Finite automata and the representation of events. In: Technical Report WADD TR-57-624. p. 112–137 (1957)
17. Nerode, A.: Linear automaton transformations. In: American Mathematical Society. p. 541–544 (1958)
18. Sistla, A.P., Vardi, M.Y., Wolper, P.: The complementation problem for Büchi automata with applications to temporal logic. *Theoretical Computer Science* **49**(2–3), 217–237 (1987)
19. Tabajara, L.M., Vardi, M.Y.: LTLf synthesis under partial observability: From theory to practice. In: Raskin, J., Bresolin, D. (eds.) GandALF. EPTCS, vol. 326, pp. 1–17 (2020)
20. Thomas, W.: Automata on infinite objects. In: Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics, pp. 133–191. Elsevier and MIT Press (1990)
21. Vardi, M.Y., Wolper, P.: An automata-theoretic approach to automatic program verification (preliminary report). In: LICS. pp. 332–344. IEEE (1986)
22. Yan, Q.: Lower bounds for complementation of  $\omega$ -automata via the full automata technique. *Logical Methods in Computer Science* **4**(1:5) (2008)

## A Proofs

**Lemma 2** ([18, 20]). *Let  $\sim$  be the congruence relation in Definition 5. Then  $|\sim| \leq 3^{n^2}$ .*

*Proof.* An equivalence class  $[u]_{\sim}$  can be encoded as a set  $S \subseteq \mathbf{T} = (Q \times \{0, 1\} \times Q)$  of labelled pairs of states. Intuitively, a tuple  $(q, 0, r)$  indicates that  $q \xrightarrow{u} r$ , while  $(q, 1, r)$  means  $q \xrightarrow{u} r$ . In particular, the equivalence class  $[\epsilon]_{\sim}$  is encoded as the set  $\{(q, 0, q) \mid q \in Q\} \cup \{(q, 1, q) \mid q \in F\}$ . Assume that  $u_1 \sim u_2$ . Let  $S_1 = \{(q, 0, r) \in \mathbf{T} \mid q \xrightarrow{u_1} r, q \in Q, r \in Q\} \cup \{(q, 1, r) \in \mathbf{T} \mid q \xrightarrow{u_1} r, q \in Q, r \in Q\}$  and  $S_2 = \{(q, 0, r) \in \mathbf{T} \mid q \xrightarrow{u_2} r, q \in Q, r \in Q\} \cup \{(q, 1, r) \in \mathbf{T} \mid q \xrightarrow{u_2} r, q \in Q, r \in Q\}$ . It is easy to verify that  $S_1 = S_2$ . Thus an equivalence class  $[u]_{\sim}$  can be uniquely represented as a set  $S \subseteq \mathbf{T}$ .

Note the set  $\mathbf{T}$  is finite, thus, the congruence relation  $\sim$  is of finite index. We thus can assume that  $m = |\sim| = |\Sigma^* / \sim|$ . Since there can be multiple valid pairs of states for a word  $u \in \Sigma^*$ , the class  $[u]_{\sim}$  corresponds to a subset of  $\mathbf{T}$ . There are at most  $n \times n$  distinct pairs of states and for each pair  $(q, r)$ , we either have both  $(q, 1, r) \in S$  and  $(q, 0, r) \in S$ , just  $(q, 0, r) \in S$  or the pair is not present in  $S$ . Thus we have  $m = |\sim| = |\Sigma^* / \sim| \leq 3^{n^2}$ .

**Lemma 4.** *Let  $\sim^i$  be the right congruence in Definition 7. Then  $|\sim^i| \leq 2^n$ .*

*Proof.* One can uniquely encode each equivalence class  $[u]_{\sim^i}$  as a set of states  $\delta(I, u)$ . Therefore, the number of equivalence classes of  $\sim^i$  is at most  $2^n$ .

**Lemma 5.** *For each  $u \in \Sigma^*$ , let  $\approx_u$  be the right congruence defined in Definition 8. Then  $|\approx_u| \leq 3^{n^2}$ .*

*Proof.* Similarly to Lemma 2, we can uniquely encode an equivalence class  $[v]_{\approx_u}$  as a set  $S \subseteq \mathbf{T} = (\delta(I, x) \times \{0, 1\} \times Q)$ . Thus,  $|\approx_u| \leq 3^{n^2}$ .

**Theorem 3.** *Let  $\sim$  be the congruence relation in Definition 5 and  $\approx_u$  the right congruence in Definition 8 with  $u \in \Sigma^*$ . There is a family of NBWs  $\mathcal{B}_1, \dots, \mathcal{B}_n$  with  $n + 3$  states for which  $|\sim| \geq n!$  and  $|\approx_u| \leq 2 \times (n + 3)$  for each  $u \in \Sigma^*$ .*

*Proof.* The family of NBWs  $\mathcal{B}_n$ , inspired from [4], is depicted in Figure 2 with  $n + 3$  states. Let  $\delta_n$  and  $Q$  be the transition function and the set of states of  $\mathcal{B}_n$ , respectively. We can see from Figure 2 that the initial state is  $q$  and  $F = \{q, q_{-1}\}$ .

We first show that for each  $u \in \Sigma^*$ ,  $|[u]_{\sim^i}| \leq 2 \times (n + 3)$ , which also indicates that the FDFW  $\mathcal{F}$  constructed from  $\mathcal{B}_n$  has  $\mathcal{O}(n^2)$  states. By Definition 7, each equivalence class  $[u]_{\sim^i}$  can be encoded as  $\delta(q, u)$ . Therefore, the number of equivalence classes of  $\sim^i$  is  $n + 3$ , namely  $\{q_0\}, \{q_1\}, \dots, \{q_n\}, \{q\}, \{q_0, q_{-1}\}$ . Next we show that for an equivalence class  $[u]_{\sim^i}$ , the number of equivalence classes of  $\approx_u$  is at most  $2 \times (n + 3)$ .

Following we fix a word  $v \in \Sigma^+$ . For an equivalence class, say  $[u]_{\sim^i} = \{p\}$  where  $p \in Q \setminus \{q_{-1}, q_0\}$ , if  $v$  does not contain 0, then there is at most one state  $r$

such that  $p \xrightarrow{v} r$  for the fixed word  $v$ . We may also have  $p \xRightarrow{v} r$ . Therefore, the number of possible  $[v]_{\approx_u}$  is at most 2. Otherwise if  $v$  contains 0, the number of possible  $r$  is at most 2, i.e.,  $r \in \{q_0, q_{-1}\}$ . Then we either have  $\{p \xRightarrow{v} q_{-1}, p \xrightarrow{v} q_{-1}, p \xrightarrow{v} q_0\}$  or just  $\{p \xRightarrow{v} q_0, p \xrightarrow{v} q_0\}$ . Therefore, the number of possible  $[v]_{\approx_u}$  is also at most 2. It follows that for equivalence class  $[u]_{\sim^i} = \{p\}$  where  $p \in Q \setminus \{q_{-1}, q_0\}$ , the number of possible  $[v]_{\approx_u}$  is at most 2 for a given  $v$ .

For  $[u]_{\sim^i} = \{q_0\}$ ,  $[v]_{\approx_u}$  can be encoded as either  $\{q_0 \xrightarrow{v} q_0\}$ , or  $\{q_0 \xrightarrow{v} q_0, q_0 \xrightarrow{v} q_{-1}, q_0 \xRightarrow{v} q_{-1}\}$ . So, the number of possible  $[v]_{\approx_u}$  is also at most 2.

Similarly, for  $[u]_{\sim^i} = \{q_0, q_{-1}\}$ , the number of possible  $[v]_{\approx_u}$  is at most 2.

Since each  $v \in \Sigma^+$  can be uniquely mapped to a set in  $\{q_0\}, \{q_1\}, \dots, \{q_n\}, \{q\}, \{q_0, q_{-1}\}$  (reachable states from the set of states of  $[u]_{\sim^i}$  over  $v$ ), then the number of equivalence classes of  $\approx_u$  with  $u \in \Sigma^*$  is at most  $2 \times (n+3)$ . There are  $n+3$  equivalence classes for  $\sim^i$ . It also follows that the FDFW constructed in Definition 12 has  $(n+3) + 2 \times (n+3)^2 \in \mathcal{O}(n^2)$  macrostates.

Now we show that the number of equivalence classes of  $\sim$  in Definition 5 is at least  $n!$ .

*Short version of the proof.* To prove that  $|\sim| \geq n!$ , one can just show that for each pair of different permutation words  $u = i_1, \dots, i_n$  and  $u' = i'_1, \dots, i'_n$  of  $\{1, \dots, n\}$ ,  $u \not\sim u'$ . We denote by  $k$  the smallest position such that  $i_k \neq i'_k$  where  $1 \leq k < n$ . If  $k = 1$ , i.e.,  $i_1 \neq i'_1$ , then  $u \not\sim u'$  by Definition 5, since we have  $q \xrightarrow{u} q_{i_1}$  and  $q \xrightarrow{u'} q_{i'_1}$  with  $q_{i_1} \neq q_{i'_1}$ . Note that  $\delta_n$  is deterministic on words without the letter 0. Thus  $q$  transitions to  $q_{i_1}$  after reading  $i_1$  and stays there for the remaining  $i_2, \dots, i_n$ , i.e.,  $q \xrightarrow{u} q_{i_1}$ . Similarly, we have  $q \xrightarrow{u'} q_{i'_1}$ . When  $1 < k < n$ , we have  $i_{k-1} = i'_{k-1}$ . Analogously,  $u \not\sim u'$  since we have  $q_{i_{k-1}} \xrightarrow{u} q_{i_k}$  while  $q_{i_{k-1}} \xrightarrow{u'} q_{i'_k}$  with  $i_k \neq i'_k$ . Therefore, we conclude that  $u \not\sim u'$ . Thus, the number of equivalence classes of  $\sim$  is at least  $n!$ .

*Long version of the proof.* Consider the NBW  $\mathcal{B}_n$  depicted in Figure 2: to prove above claim, we can just show that for each pair of different permutation words  $u = i_1, \dots, i_n$  and  $u' = i'_1, \dots, i'_n$  of  $\{1, \dots, n\}$ ,  $u \not\sim u'$ . We first define an index function  $g_u : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$  for each  $u = i_1, i_2, \dots, i_n$  such that  $g(i_k) = k$  for each  $k \geq 1$ . That is,  $g_u$  obtains the index  $k$  of a letter  $i_k$  in  $u$ . Obviously,  $g_u \neq g_{u'}$  if  $u \neq u'$ . For instance, when  $u = 1, 2, \dots, n$ , we have  $g_u(k) = k$  for each  $k \geq 1$ . Now we define another function  $f_u : \{q, q_1, \dots, q_n\} \rightarrow \{q, q_1, \dots, q_n\}$  to represent the reachability between states over the word  $u$ . When  $u = 1, 2, \dots, n$ , we have  $q_{k-1} \xrightarrow{u} q_k$  for each  $2 \leq k \leq n$ ,  $q \xrightarrow{u} q_1$ ,  $q_0 \xrightarrow{u} q_{-1}$  and  $q_{-1} \xrightarrow{u} q_{-1}$ . Here we can just omit  $q_{-1}$  and  $q_0$  since we have only  $q_{-1} \xrightarrow{u} q_{-1}$  and  $q_0 \xrightarrow{u} q_{-1}$  for each permutation word  $u$ . Moreover, we can omit the reachability path that visits accepting states, as this does not affect the lower bound of the complexity. Therefore, for  $u = 1, \dots, n$ , we have  $f_u(q_{k-1}) = q_k$  for each  $2 \leq k \leq n$ ,  $f_u(q_n) = q$  and  $f_u(q) = q_1$ . In fact, according to the transition

function  $\delta_n$ , the function  $f_u$  can be defined with  $g_u$  on the input  $u = i_1, \dots, i_n$  as follows:

- First,  $f_u(q) = q_{i_1}$ . From Figure 2, it is easy to verify that from state  $q$ ,  $\mathcal{B}_n$  first goes to state  $q_{i_1}$  after inputting  $i_1$  and then stays there for the remaining letters.
- Second,  $f_u(q_j) = q_{i_{g_u(j)+1}}$  if  $g_u(j)+1 \leq n$ , otherwise  $f_u(q_j) = q$ , as  $g_u(j) = n$ . Intuitively,  $\mathcal{B}_n$  will stay at  $q_j$  until seeing the letter  $j$ . Once received the letter  $j$ ,  $\mathcal{B}_n$  will go from  $q_j$  to  $q$  and then go on reading the letter  $k = i_{g_u(j)+1}$  if  $g_u(j) < n$  and transitions to state  $q_k$  and stay there. If  $g_u(j) = n$ , then  $\mathcal{B}_n$  will stay at  $q$ .

Given two different permutation words  $u$  and  $u'$  of  $\{1, \dots, n\}$ , we denote by  $k$  the smallest position such that  $i_k \neq i'_k$  where  $1 \leq k < n$ . If  $k = 1$ , i.e.,  $i_1 \neq i'_1$ , then  $f_u(q) = q_{i_1}$  while  $f_{u'}(q) = q_{i'_1}$ . Therefore,  $u \not\sim u'$  since we have  $q \xrightarrow{u} q_{i_1}$  and  $q \xrightarrow{u'} q_{i'_1}$  with  $q_{i_1} \neq q_{i'_1}$ . Note that  $\delta_n$  is deterministic on words without the letter 0.

Otherwise we have  $i_k \neq i'_k$  for  $1 < k < n$  and  $i_{k-1} = i'_{k-1}$ . Then we have  $f_u(q_{i_{k-1}}) = q_{i_k}$  while  $f_{u'}(q_{i_{k-1}}) = q_{i'_k}$ . Then  $u \not\sim u'$  since we have  $q_{i_{k-1}} \xrightarrow{u} q_{i_k}$  and  $q_{i_{k-1}} \xrightarrow{u'} q_{i'_k}$  with  $i_k \neq i'_k$ . We then conclude that  $u \not\sim u'$ . Thus, the number of equivalence classes of  $\sim$  is at least  $n!$ .

**Lemma 7.** *Let  $\sim^o$  be the right congruence in Definition 10. Then  $|\sim^o| \leq n^n$ .*

*Proof.* According to [10], the number of possible ordered sets of  $Q$  is approximately  $(0.53n)^n$ . Thus we have that  $|\sim^o| \leq n^n$ .

**Lemma 8.** *For all  $u_1, u_2 \in \Sigma^*$ , if  $u_1 \sim^o u_2$ , then  $u_1 \sim^i u_2$ .*

*Proof.* If  $u_1 \sim^o u_2$ , then  $\phi(I/\preceq, u_1) = \phi(I/\preceq, u_2)$ . By Definition 9, it is easy to see that  $\delta(I, u_1) = \delta(I, u_2)$ . Thus we have  $u_1 \sim u_2$ .

**Lemma 10 (Saturation Lemma).**

1. For  $u \in \Sigma^*, v \in \Sigma^+$ , if  $uv \sim^o u$ , then either  $[u]_{\sim^o} [v]_{\approx_u^o}^\omega \cap \mathcal{L}(\mathcal{A}) = \emptyset$  or  $[u]_{\sim^o} [v]_{\approx_u^o}^\omega \subseteq \mathcal{L}(\mathcal{A})$ .
2.  $\Sigma^\omega = \bigcup \{ [u]_{\sim^o} [v]_{\approx_u^o}^\omega \mid u \in \Sigma^*, v \in \Sigma^+, uv \sim^o u \}$ .
3.  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}) = \bigcup \{ [u]_{\sim^o} [v]_{\approx_u^o}^\omega \mid u \in \Sigma^*, v \in \Sigma^+, uv \sim^o u, [u]_{\sim^o} [v]_{\approx_u^o}^\omega \cap \mathcal{L}(\mathcal{A}) = \emptyset \}$ .

*Proof (Proof Sketch).* Here we only prove Item (1). The Items (2) and (3) can be proved similarly as in the proof of Lemma 6.

Assume that  $uv \sim^o u$ . By Definition 10,  $\phi(I/\preceq, u) = \phi(I/\preceq, uv)$ . By Definition 11, for each word  $v' \in [v]_{\approx_u^o}$ , we have  $uv \sim^o uv'$ , which implies  $uv \sim^o uv' \sim^o u$ . It follows that  $[u]_{\sim^o} [v]_{\approx_u^o}^\omega = [u]_{\sim^o}$ . Assume that there exists a word  $w \in [u]_{\sim^o} [v]_{\approx_u^o}^\omega \cap \mathcal{L}(\mathcal{A})$ . We can then decompose  $w$  as  $w = u_0 \cdot v_1 \cdot v_2 \cdots$  with  $u_0 \in [u]_{\sim^o}$  and  $v_i \in [v]_{\approx_u^o}$  for each  $i \geq 1$ . It follows that  $u \sim^o u_0 v_i \sim^o u_0 v_1 \cdots v_i$

for each  $i \geq 1$  since  $u_0 v_i \sim^o u_0$  for each  $i \geq 1$ . Then  $\phi(I/\preceq, u) = \phi(I/\preceq, uv_i) = \phi(I/\preceq, uv_1 \cdots v_i)$  for every  $i \geq 1$  since  $u \sim^o u_0 v_i \sim^o u_0 v_1 \cdots v_i$  for each  $i \geq 1$  and  $\phi$  is deterministic. Therefore, the ordered sets  $\phi(I/\preceq, u) = \phi(I/\preceq, uv_i)$  have been visited for infinitely many times and they can be reachable by themselves in the same way (cf. Item (2) in Definition 11). One can construct an ordered run DAG  $G_{w, \mathcal{A}}$  of  $\mathcal{A}$  over  $w = u_0 \cdot v_1 \cdot v_2 \cdots$  with Definition 9 (cf. Appendix C.2 for details). Therefore, from the ordered run DAG  $G_{w, \mathcal{A}}$ , there exists an *execution* of  $\mathcal{A}$  over  $w$  written as  $\rho_{u_0} = q \xrightarrow{u_0} q_0 \xrightarrow{v_1} q_1 \xrightarrow{v_2} q_2 \cdots$  where  $q \in I, [q]_{\preceq} \in \phi([q]_{\preceq}, u_0)$ ,  $[q_i]_{\preceq} \in \phi(I/\preceq, u_0 v_i)$ , and  $[q_i]_{\preceq}$  is the maximal equivalence class in  $\phi(I/\preceq, u_0)$  that reaches the state  $q_{i+1}$  for each  $i \geq 0$ . Moreover, there exists an execution visiting infinitely many accepting states when  $w \in \mathcal{L}(\mathcal{A})$ . Let  $w' = u'_0 v'_1 v'_2 \cdots$  with  $u'_0 \in [u]_{\sim^o}$  and  $v'_i \in [v]_{\approx_u^o}$  for each  $i \geq 1$ . Similarly to the proof of Lemma 6, adapted to notions in Definition 11, we can also construct such an execution that visits accepting states infinitely many times for  $w'$ . Therefore,  $w$  is accepting if and only if  $w'$  is accepting. It follows that  $[u]_{\sim^o} [v]_{\approx_u^o}^\omega \subseteq \mathcal{L}(\mathcal{A})$  if  $[u]_{\sim^o} [v]_{\approx_u^o}^\omega \cap \mathcal{L}(\mathcal{A}) \neq \emptyset$ . We note that detailed construction of the ordered run DAG is provided in Appendix C.2.

**Theorem 6.** *Let  $\mathcal{F}$  be the FDFW constructed from  $\mathcal{A}$  in Definition 12. Then (1)  $\text{UP}(\mathcal{F}) = \text{UP}(\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}))$ ; (2)  $\mathcal{F}$  is saturated; and (3)  $\mathcal{F}$  has  $2^{\mathcal{O}(n \log n)}$  macrostates.*

*Proof.* We first prove Item (3). First there are at most  $n^n$  macrostates in the leading DFW  $\mathcal{M}$ . For each progress DFW  $\mathcal{N}_u$  of  $[u]_{\sim^o}$ , there are at most  $n^n \times (n+1)^n$  macrostates, since the number of equivalence classes of  $\approx_u^o$  is at most  $n^n \times (n+1)^n$ . Therefore, there are  $n^n + n^n \times n^n \times (n+1)^n \in 2^{\mathcal{O}(n \log n)}$  macrostates in  $\mathcal{F}$ .

Now we prove Item (1). Assume that  $w \in \text{UP}(\mathcal{F})$ . Then there exists a normalized decomposition  $(u', v')$  of  $w$  such that  $(u', v')$  is accepted by  $\mathcal{F}$ . Let  $u' \in [u]_{\sim^o}$  and  $v' \in [v]_{\approx_u^o}$ . It follows that we have  $[u]_{\sim^o} = \mathcal{M}(u) = \mathcal{M}(u') = \mathcal{M}(uv) = \mathcal{M}(u'v')$  and  $[v]_{\approx_u^o} = \mathcal{N}_u(v')$  is an accepting macrostate. Thus  $u'v'^\omega \notin \mathcal{L}(\mathcal{A})$  by Definition 12, i.e.,  $\text{UP}(\mathcal{F}) \subseteq \text{UP}(\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}))$ . For the other direction, we assume that  $w \in \text{UP}(\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}))$ . By Lemma 10, there exists an  $\omega$ -language  $[u]_{\sim^o} [v]_{\approx_u^o}^\omega \cap \mathcal{L}(\mathcal{A})$  with  $uv \sim^o u$  such that  $w \in [u]_{\sim^o} [v]_{\approx_u^o}^\omega$ . W.l.o.g., let  $w = uv^\omega$ . It follows that  $\mathcal{N}_u(v)$  is an accepting macrostate, i.e.,  $w$  is accepted by  $\mathcal{F}$ . Therefore,  $\text{UP}(\Sigma^\omega \setminus \mathcal{L}(\mathcal{A})) \subseteq \text{UP}(\mathcal{F})$ . We then have that  $\text{UP}(\mathcal{F}) = \text{UP}(\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}))$ .

Now we prove item (2). For two normalized decompositions  $(u_1, v_1)$  and  $(u_2, v_2)$  of  $w$ , we assume that  $(u_1, v_1)$  is accepted by  $\mathcal{F}$ . It follows that  $w = u_1 v_1^\omega = u_2 v_2^\omega$  is not accepted by  $\mathcal{A}$ . By Definition 12, let  $[u]_{\sim^o} = \mathcal{M}(u_2)$  and  $[v]_{\approx_u^o} = \mathcal{N}_u(v_2)$  where  $\mathcal{N}_u$  is the progress DFW of  $[u]_{\sim^o}$  of  $\mathcal{M}$ . It follows that  $u_2 \in [u]_{\sim^o}$  and  $v_2 \in [v]_{\approx_u^o}$ . According to the proof of Lemma 10, we have that  $uv \sim^o u$  if  $u_2 v_2 \sim^o u_2$ . By Lemma 10, we have  $[u]_{\sim^o} [v]_{\approx_u^o}^\omega \cap \mathcal{L}(\mathcal{A}) = \emptyset$  since  $u_2 v_2^\omega \in [u]_{\sim^o} [v]_{\approx_u^o}^\omega$  and  $uv \sim^o u$ . According to Definition 12,  $[v]_{\approx_u^o}$  is thus an accepting macrostate of  $\mathcal{N}_u$ . It follows that  $(u_2, v_2)$  is also accepted by  $\mathcal{F}$ . Thus  $\mathcal{F}$  is saturated.

**Theorem 7.** *There exists a family of NBWs  $\mathcal{A}_1, \dots, \mathcal{A}_n$  with  $n$  states for which an FDFW  $\mathcal{F}_n$  accepting  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}_n)$  has  $2^{\Omega(n \log n)}$  macrostates.*

*Proof.* It is shown in [22] that there exists a family of  $\omega$ -languages  $L_1, \dots, L_n$  with  $n \geq 1$ , such that for each  $n$ , there exists an NBW  $\mathcal{A}_n$  of  $L_n$  with  $n$  states, while all NBWs accepting the complementary language  $L_n^c$  have  $2^{\Omega(n \log n)}$  states.

Assume that the FDFW  $\mathcal{F}_n$  accepting  $\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}_n)$  has  $m$  macrostates such that  $m < 2^{\Omega(n \log n)}$ . According to Lemma 1, we can construct a complementary NBW  $\mathcal{A}_n^c$  of  $\mathcal{B}_n$  with  $\mathcal{O}(m^3)$  states from  $\mathcal{F}$  such that  $\text{UP}(\mathcal{F}) = \text{UP}(\Sigma^\omega \setminus \mathcal{L}(\mathcal{A}_n)) = \text{UP}(\mathcal{L}(\mathcal{A}_n^c))$ . It follows that  $\mathcal{A}_n^c$  has  $\mathcal{O}(m^3) < 2^{\Omega(n \log n)}$  states, which contradicts with the results in [22]. Therefore,  $\mathcal{F}$  has  $2^{\Omega(n \log n)}$  macrostates.

## B Comparison with [7] and [12]

The FDFW  $\mathcal{F}$  obtained from [7] for  $\mathcal{A}$  is represented with a DFW  $\mathcal{D}_\S$  such that  $\mathcal{L}_*(\mathcal{D}_\S) = \{ u\$v \in \Sigma^*\Sigma^+ \mid uv^\omega = w, w \in \mathcal{L}(\mathcal{A}) \}$  where  $\$ \notin \Sigma$ .  $\mathcal{D}_\S$  has  $\mathcal{O}(4^{n^2+n})$  states by reusing shared macrostates of  $\mathcal{F}$ . This construction has been further improved in [12], which produces a DFW  $\mathcal{D}_\S$  with  $\mathcal{O}(3^{n^2+n})$  states. Note that the resulting FDFW  $\mathcal{F}$  from [7, 12] actually has more macrostates than  $\mathcal{D}_\S$ . All above two constructions yield an FDFW  $\mathcal{F}$  such that each decomposition  $(u, v)$  of  $w \in \mathcal{L}(\mathcal{A})$  is captured by  $\mathcal{F}$ , as  $\mathcal{D}_\S$  accepts each  $u\$v$  such that  $uv^\omega \in \mathcal{L}(\mathcal{A})$ . In contrast, ours only captures normalized decompositions of  $w$ , see the condition  $uv \sim^i u$  in Definition 12 replaced with  $\sim^i$  and  $\approx_u$ .

Our direct construction for FDFWs with Definition 12 replaced with  $\sim^i$  and  $\approx_u$  has the same worst-case complexity with the one in [12] since  $|\approx_u| \leq 3^{n^2}$ . Nonetheless, we show that the FDFW constructed in [7, 12] may be exponentially larger than the FDFWs by our construction even for DBWs, since the FDFWs constructed in [7, 12] try to capture each decomposition of desired UP-words.

**Theorem 9.** *There exists a family of DBWs  $\mathcal{B}'_1, \dots, \mathcal{B}'_n$  with  $n + 2$  states for which the FDFW obtained from [7, 12] has  $2^{\Omega(n \log n)}$  macrostates while the FDFW constructed in Definition 12 has  $\mathcal{O}(n^2)$  macrostates.*

*Proof.* We can obtain the DBW  $\mathcal{B}'_n$  from the NBW  $\mathcal{B}_n$  in Figure 2 by removing the accepting state  $q_{-1}$  and making  $q_0$  a sink nonaccepting state. Clearly, the resulting  $\mathcal{B}'_n$  is a DBW. By Lemma 4, the FDFW constructed in Definition 12 has  $\mathcal{O}(n^2)$  macrostates. According to [4], the minimal FDFW  $\mathcal{F}'$  capturing each decomposition  $(u, v)$  of  $w \in \mathcal{L}(\mathcal{B}'_n)$  has at least  $(n + 1)! \in 2^{\mathcal{O}(n \log n)}$  macrostates regardless of the transition structure of  $\mathcal{B}'_n$ . The proof idea is similar to the one for Theorem 3; one can prove that over each pair of different permutation words  $u = i_1, \dots, i_n$  and  $u' = i'_1, \dots, i'_n$  of  $\{1, \dots, n\}$ , the progress DFW for the equivalence class that  $\epsilon$  belongs to in  $\mathcal{F}'$  will reach different macrostates. We refer the reader to [4] for the detailed proof. Thus we complete the proof.

## C Ordered Run DAG in [10]

### C.1 Run DAGs

Let  $\mathcal{A} = (Q, I, \delta, F)$  be an NBW and  $w = a_0 a_1 \dots$  be an infinite word. The run DAG  $G_{w, \mathcal{A}} = \langle V, E \rangle$  of  $\mathcal{A}$  over  $w$  is defined as follows:

- Vertices:  $V \subseteq Q \times \mathbb{N}$  is the set of vertices  $\bigcup_{l \geq 0} V_l \times \{l\}$  where  $V_0 = I$  and  $V_{l+1} := \delta(V_l, a_l)$  for every  $l \geq 0$ .
- Edges: There is an edge from  $\langle q, l \rangle$  to  $\langle q', l' \rangle$  iff  $l' = l + 1$  and  $q' \in \delta(q, a_l)$ .

A vertex  $\langle q, l \rangle$  is said to be on level  $l$  and there are at most  $|Q|$  states on each level. A vertex  $\langle q, l \rangle$  is an *F-vertex* if  $q \in F$ . A sequence of vertices  $\hat{\rho} = \langle q_0, 0 \rangle \langle q_1, 1 \rangle \dots$  is called a *branch* of  $G_{w, \mathcal{A}}$  if  $q_0 \in I$  and for each  $l \geq 0$ , there is an edge from  $\langle q_l, l \rangle$  to  $\langle q_{l+1}, l+1 \rangle$ . An  $\omega$ -*branch* of  $G_{w, \mathcal{A}}$  is a branch of infinite length. A finite *fragment*  $\langle q_l, l \rangle \langle q_{l+1}, l+1 \rangle \dots$  of  $\hat{\rho}$  is said to be a branch from the vertex  $\langle q_l, l \rangle$ ; a fragment  $\langle q_l, l \rangle \dots \langle q_{l+k}, l+k \rangle$  of  $\hat{\rho}$  is said to be a *path* from  $\langle q_l, l \rangle$  to  $\langle q_{l+k}, l+k \rangle$ , where  $k \geq 1$ . A vertex  $\langle q_j, j \rangle$  is *reachable* from  $\langle q_l, l \rangle$  if there is a path from  $\langle q_l, l \rangle$  to  $\langle q_j, j \rangle$ . We call a vertex  $\langle q, l \rangle$  is *finite* in  $G_{w, \mathcal{A}}$  if there are no  $\omega$ -branches in  $G_{w, \mathcal{A}}$  starting from  $\langle q, l \rangle$ ; and we call a vertex  $\langle q, l \rangle$  *F-free* if it is not finite and no *F*-vertices are reachable from  $\langle q, l \rangle$  in  $G_{w, \mathcal{A}}$ .

There is a bijection between the set of runs of  $\mathcal{A}$  on  $w$  and the set of  $\omega$ -branches in  $G_{w, \mathcal{A}}$ . To a run  $\rho = q_0 q_1 \dots$  of  $\mathcal{A}$  over  $w$  corresponds an  $\omega$ -branch  $\hat{\rho} = \langle q_0, 0 \rangle \langle q_1, 1 \rangle \dots$ . Therefore,  $w$  is accepted by  $\mathcal{A}$  if and only if there exists an  $\omega$ -branch in  $G_{w, \mathcal{A}}$  that visits *F*-vertices infinitely often; we say that such an  $\omega$ -branch is *accepting*;  $G_{w, \mathcal{A}}$  is accepting if and only if there exists an accepting  $\omega$ -branch in  $G_{w, \mathcal{A}}$ .

### C.2 Ordered Run DAG and Preorder $\preceq$

In [10], every vertex is labelled with a set of states instead of one state. Moreover, the vertices at the same level are put in an order described below.

The ordered run DAG  $G_{w, \mathcal{A}}^o$  is constructed as we proceed along the word  $w$ . Here the superscript  $o$  is used to mark that  $G_{w, \mathcal{A}}^o$  is an ordered run DAG. At level 0, we may obtain at most two vertices of  $G_{w, \mathcal{A}}^o$ : a vertex  $\langle S_1, 0 \rangle = \langle I \setminus F, 0 \rangle$  and an *F*-vertex  $\langle S_2, 0 \rangle = \langle I \cap F, 0 \rangle$ . Recall that  $I$  and  $F$  are the set of initial states and the set of accepting states of  $\mathcal{A}$ , respectively. Here  $S_1$  and  $S_2$  are disjoint. A vertex  $\langle S_j, i \rangle$  is an *F*-vertex if  $S_j \subseteq F$ , where  $j \geq 1$  and  $i \geq 0$ . The vertices  $\langle S_j, i \rangle$  on level  $i$  in  $G_{w, \mathcal{A}}^o$  are ordered from left to right by their indices  $j$  where  $i \geq 0$  and  $1 \leq j \leq n$ . During the construction, empty sets  $S_j$  are removed and the indices of remaining sets are reset according to the increasing order of their original indices.

Assume that on level  $i$ , the sequence of vertices in  $G_{w, \mathcal{A}}^o$  is  $\langle S_1, i \rangle, \dots, \langle S_{k_i}, i \rangle$  where  $i \geq 0$  and  $1 \leq k_i \leq n$ . Intuitively, here the set of ordered sets  $\langle S_1, i \rangle, \dots, \langle S_{k_i}, i \rangle$  corresponds to the ordered sets  $(\bigcup_{1 \leq j \leq k_i} S_j) / \preceq$  such that  $S_1 \prec \dots \prec S_{k_i}$ . Specially, at level 0, if there are two vertices  $\langle I \setminus F, 0 \rangle$  and  $\langle I \cap F, 0 \rangle$ , we then have  $I \setminus F \prec I \cap F$ . This can be formalized in Definition 9

by the fact that for all states  $p, q \in I$ ,  $[p]_{\preceq} \prec [q]_{\preceq}$  if  $p \notin F$  and  $q \in F$ , otherwise we have  $[q]_{\preceq} \preceq [p]_{\preceq}$ .

We now describe how to construct the vertices on level  $i + 1$ . First, for a set  $S_j$  where  $1 \leq j \leq k_i$ , on reading the letter  $w[i]$ , the set of successors of  $S_j$  is partitioned into (1) a non- $F$  set  $S'_{2j-1} = \delta(S_j, w[i]) \setminus F$ , and (2) an  $F$ -set  $S'_{2j} = \delta(S_j, w[i]) \cap F$ , as a possible new  $F$ -vertex. This gives us a sequence of sets  $S'_1, S'_2, \dots, S'_{2k_i-1}, S'_{2k_i}$ . Note that there can be some states in  $\mathcal{A}$  present in multiple sets  $S'_j$  where  $j \geq 1$ . Here we only keep the rightmost occurrence of a state. Intuitively, different runs of  $\mathcal{A}$  may merge with each other at some level and we only need to keep the right most one and cut off others, as they share the same infinite suffix. This operation does not change whether the ordered DAG  $G_{w, \mathcal{A}}^o$  is accepting, since at least one accepting run of  $\mathcal{A}$  remains and will not be cut off. Formally, for each set  $S'_j$  where  $1 \leq j \leq 2k_i$ , we define a set  $S''_j = S'_j \setminus \bigcup_{p \leq 2k_i} S'_p$ . This yields a sequence of disjoint sets  $S''_1, S''_2, \dots, S''_{2k_i-1}, S''_{2k_i}$ . After removing the empty sets in this sequence and reassigning the index of each set according to their positions, we finally obtain the sequence of sets of vertices on level  $i + 1$ , denoted by  $\langle S_1, l + 1 \rangle, \dots, \langle S_{k_{i+1}}, l + 1 \rangle$ . Obviously, the resulting sets at the same level are again pairwise disjoint.

Keeping the rightmost occurrence of a state actually corresponds to a successor inheriting the maximal order of the predecessors in Definition 9. We recall the definition of  $a$ -successor of  $P/\preceq$  as below. Let  $P/\preceq \in \mathcal{R}$  be ordered sets of  $P \subseteq Q$ ,  $a \in \Sigma$  a letter and  $P' = \delta(P, a)$ . We can show that if  $P/\preceq$  is  $(\bigcup_{1 \leq j \leq k_i} S_j)/\preceq$  such that  $S_1 \prec \dots \prec S_{k_i}$  at level  $l$ , then  $P'/\preceq$  is  $(\bigcup_{1 \leq j \leq k_{i+1}} S_j)/\preceq$  such that  $S_1 \prec \dots \prec S_{k_{i+1}}$  at level  $l + 1$ . We define the ordered  $a$ -successor  $P'/\preceq$  of  $P/\preceq$ , denoted by  $\phi(P/\preceq, a)$ , as follows. For all states  $p', q' \in P'$ ,

- $[p']_{\preceq} \prec [q']_{\preceq}$  if  $\max_{\preceq} \{p \in P \mid p \xrightarrow{a} p'\} \prec \max_{\preceq} \{q \in P \mid q \xrightarrow{a} q'\}$ ; Intuitively, we only keep the rightmost occurrence (successors of maximal equivalence class) here if  $p'$  and  $q'$  have multiple occurrences.
- Assume that  $\max_{\preceq} \{p \in P \mid p \xrightarrow{a} p'\} = \max_{\preceq} \{q \in P \mid q \xrightarrow{a} q'\}$ :
  - $[p']_{\preceq} \preceq [q']_{\preceq}$  if  $p' \notin F$ , and
  - $[p']_{\preceq} \prec [q']_{\preceq}$  if  $q' \in F$  and  $p' \notin F$ .

Intuitively, here we divide the set of successors of  $\max_{\preceq} \{p \in P \mid p \xrightarrow{a} p'\}$  into non- $F$  states and  $F$ -states, so  $F$ -states will be put on the right side.

Therefore, one can also construct an ordered run DAG of  $\mathcal{A}$  over  $w$  with the preorder  $\preceq$ . Moreover, we have the following lemma that follows from the construction of ordered run DAG described above.

**Lemma 11.** *We can use the preorder  $\preceq$  in Definition 9 to construct an ordered run DAG  $G_{w, \mathcal{A}}$  as described in [10] and the ordered run DAG  $G_{w, \mathcal{A}}$  is accepting if and only if  $w$  is accepting. Moreover, one can extract from each  $\omega$ -branch in  $G_{w, \mathcal{A}}$  an execution  $q \xrightarrow{u_0} q_0 \xrightarrow{v_1} q_1 \xrightarrow{v_2} q_2 \dots$  where  $q \in I, [q_0]_{\preceq} \in \phi([q]_{\preceq}, u_0)$ ,  $[q_i]_{\preceq} = \max_{\preceq} \{ [p]_{\preceq} \in \phi(I/\preceq, u_0 v_1 \dots v_i) \mid p \xrightarrow{v_i} q_{i+1} \}$  for each  $i \geq 0$ . Here  $v_0$  means empty word.*