IN SEARCH OF MOST COMPLEX REGULAR LANGUAGES

JANUSZ BRZOZOWSKI
David R. Cheriton School of Computer Science,
University of Waterloo, Waterloo, ON, Canada N2L 3G1
brzozo@uwaterloo.ca

Received (Day Month Year)
Accepted (Day Month Year)
Communicated by (xxxxxxxxxx)

Sequences $(L_n \mid n \geqslant k)$, called *streams*, of regular languages L_n are considered, where k is some small positive integer, n is the state complexity of L_n, and the languages in a stream differ only in the parameter n, but otherwise, have the same properties. The following measures of complexity are proposed for any stream: 1) the state complexity n of L_n, that is, the number of left quotients of L_n (used as a reference); 2) the state complexities of the left quotients of L_n; 3) the number of atoms of L_n; 4) the state complexities of the atoms of L_n; 5) the size of the syntactic semigroup of L_n; and the state complexities of the following operations: 6) the reverse of L_n; 7) the star of L_n; 8) union, intersection, difference and symmetric difference of L_m and L_n; and 9) the concatenation of L_m and L_n. A stream that has the highest possible complexity with respect to these measures is then viewed as a most complex stream. The language stream $(U_n(a,b,c) \mid n \geqslant 3)$ is defined by the deterministic finite automaton with state set $\{0,1,\dots,n-1\}$, initial state 0, set $\{n-1\}$ of final states, and input alphabet $\{a,b,c\}$, where a performs a cyclic permutation of the n states, b transposes states 0 and 1, and c maps state $n-1$ to state 0. This stream achieves the highest possible complexities with the exception of boolean operations where $m = n$. In the latter case, one can use $U_n(a,b,c)$ and $U_n(b,a,c)$, where the roles of a and b are interchanged in the second language. In this sense, $U_n(a,b,c)$ is a universal witness. This witness and its extensions also apply to a large number of combined regular operations.

Keywords: atom, complexity of operation, finite automaton, quotient complexity, regular language, state complexity, syntactic semigroup, witness

2000 Mathematics Subject Classification: 68Q45, 68Q19, 68Q70

I dedicate this work to the memory of Sheng Yu whose extensive research on state complexity led to many questions studied in this paper.

1. Introduction

State complexity is currently an active area of research in the theory of formal languages; for references, see the surveys in [1, 26] and the bibliography at the end of this paper. The *state complexity of a regular language* [26] L over a finite alphabet Σ is the number of states in the minimal complete deterministic finite automaton recognizing the language. An equivalent notion is that of *quotient complexity* [1] of L, which is the number of distinct left quotients of L. This paper uses *complexity* for both of these equivalent notions, and not for any other property.

The *(state/quotient) complexity of an operation* on regular languages is the maximal complexity of the language resulting from the operation as a function of the complexities of the arguments. For example, if the complexities of K and L are m and n, respectively, then the complexity of $K \cup L$ is at most mn and, for every m and n, there exist languages with complexities m and n meeting this bound. Thus the complexity of union is mn.

There are two parts to the process of establishing the complexity of an operation. First, one must find an *upper bound* on the complexity of the result of the operation by using quotient computations or automaton constructions. Second, one must find *witnesseses* that meet this upper bound. For the witnesses, one usually defines a sequence $(L_n \mid n \geqslant k)$ of languages, where k is some small positive integer. This sequence will be called a *stream*. The languages in a stream differ only in the parameter n. For example, one might study unary languages $(\{a^n\}^* \mid n \geqslant 1)$ that have zero a's modulo n.

A unary operation takes its argument from a stream $(L_n \mid n \geqslant k)$. For a binary operation, one adds as the second argument a stream $(K_m \mid m \geqslant k)$, normally different from the first. In the past, the witness streams used for different operations have usually been different. The following question is posed in this paper: Is it possible to use the *same* stream of languages for all the operations? In other words, is there a *universal witness* over some small fixed alphabet? The answer is "yes" for all of the basic operations and many combined operations.

Section 2 introduces the terminology and notation used in this paper. Section 3 describes common conditions that make a language complex. Section 4 introduces the main witness stream $(U_n(a,b,c) \mid n \geqslant 3)$ (U for "universal"), and states the main theorem. Properties of a single language and unary operations are treated in Section 5, whereas binary operations are discussed in Section 6. It is pointed out in Section 7 how the bounds for several combined operations are also met by $U_n(a,b,c)$ or by other streams closely related to $U_n(a,b,c)$. Section 8 concludes the paper.

2. Terminology and Notation

For background material on regular languages and finite automata see [20, 21, 25].

Let Σ be a finite non-empty set called an *alphabet*. The free semigroup generated by Σ is denoted by Σ^+; this is the set of all *non-empty* words over Σ. The free monoid generated by Σ is Σ^*; this is the set of *all* words over Σ, including the empty word ε.

Any subset of Σ^* is a *language*. The *left quotient*, or simply *quotient* of $L \subseteq \Sigma^*$ by a word $w \in \Sigma^*$ is the language $w^{-1}L = \{x \in \Sigma^* \mid wx \in L\}$. A language is regular if and only if it has a finite number of distinct quotients.

The following set operations are defined on languages K and L: complement ($\overline{L}$) of L with respect to Σ^*, union ($K \cup L$), intersection ($K \cap L$), symmetric difference ($K \oplus L$), and difference ($K \setminus L$).

The *reverse* w^R of a word w is defined inductively: $\varepsilon^R = \varepsilon$ and $(wa)^R = aw^R$ for $a \in \Sigma$, $w \in \Sigma^*$. The *reverse of a language* L is $L^R = \{w^R \mid w \in L\}$.

The *product* (also called *concatenation* or *catenation*) of languages K and L is $KL = \{uv \mid u \in K, v \in L\}$. Let $L^0 = \{\varepsilon\}$ and let $L^n = L^{n-1}L$ for $n \geqslant 1$. The *positive closure* of a language L is $L^+ = \bigcup_{n=1}^{\infty} L^n$, and the *Kleene closure* or *star* of L is $L^* = \bigcup_{n=0}^{\infty} L^n = L^+ \cup \{\varepsilon\}$.

An *atom*[a] [6,7] of a regular language L with quotients $K_0, \dots, K_{n-1}$ is a non-empty intersection of the form $\widetilde{K_0} \cap \cdots \cap \widetilde{K_{n-1}}$, where $\widetilde{K_i}$ is either K_i or $\overline{K_i}$. Thus the number of atoms is bounded from above by 2^n, and it was proved in [7] that this bound is tight. Since every quotient of L (including L itself) is a union of atoms, the atoms of L are its basic building blocks.

The *Myhill congruence* [19] $\approx_L$ of $L \subseteq \Sigma^*$ is defined as follows: For $x, y \in \Sigma^*$,

$$x \approx_L y \text{ if and only if } uxv \in L \Leftrightarrow uyv \in L \text{ for all } u, v \in \Sigma^*.$$

The *syntactic semigroup* [21] of L is the quotient semigroup $\Sigma^+ / \approx_L$.

A *deterministic finite automaton (DFA)* $\mathcal{D} = (Q, \Sigma, \delta, q_0, F)$ consists of a set Q of *states*, a finite non-empty *alphabet* Σ, a *transition function* $\delta : Q \times \Sigma \to Q$, *initial state* q_0, and set F of *final states*. The transition function is extended to functions $\delta' : Q \times \Sigma^* \to Q$ and $\delta'' : 2^Q \times \Sigma^* \to 2^Q$ as usual, but these extensions are also denoted by δ. A state q of a DFA is *reachable* if there is a word $w \in \Sigma^*$ such that $\delta(q_0, w) = q$. The *language accepted* by $\mathcal{D}$ is $L(\mathcal{D}) = \{w \in \Sigma^* \mid \delta(q_0, w) \in F\}$. Two DFAs are *equivalent* if their languages are the same. The *language of a state* q is the language accepted by the DFA $\mathcal{D}_q = (Q, \Sigma, \delta, q, F)$. Two states are *equivalent* if their languages are equal; otherwise, they are *distinguishable* by some word that is in the language of one of the states, but not of the other. A DFA is *minimal* if all of its states are reachable and no two states are equivalent. A state is *empty* if its language is empty.

A *nondeterministic finite automaton* (NFA) is a quintuple $\mathcal{D} = (Q, \Sigma, \eta, Q_0, F)$, where Q, Σ, and F are as in a DFA, $Q_0 \subseteq Q$ is the *set of initial states*, and $\eta : Q \times \Sigma \to 2^Q$ is the transition function. An ε-NFA has all the features of an NFA but its transition function $\eta : Q \times (\Sigma \cup \{\varepsilon\}) \to 2^Q$ allows also transitions under the empty word. The *language accepted* by an NFA or an ε-NFA is the set of words w for which there exists a sequence of transitions such that the concatenation of the symbols causing the transitions is w, and this sequence leads from a state in Q_0 to a state in F. Two NFAs are *equivalent* if they accept the same language.

A *transformation* of a set $Q = \{0, \dots, n-1\}$ is a mapping of Q into itself. If t is a transformation of Q and $i \in Q$, then it is the *image* of i under t. A *permutation* of Q is a mapping of Q *onto* itself. For $2 \leqslant k \leqslant n$, a permutation t is a *cycle* of length k, if there exist pairwise different elements $i_1, \dots, i_k$ such that $i_1 t = i_2, i_2 t = i_3, \dots, i_{k-1} t = i_k$, and $i_k t = i_1$, and t maps every other element to itself.

[a] Atoms of regular languages were introduced in 2011 by Brzozowski and Tamm [6], and the theory was slightly modified in 2012 [7]. The newer model, which admits up to 2^n atoms, is used here.

A cycle is denoted by $(i_1, i_2, \ldots, i_k)$. A *transposition* (i,j) is the cycle of length 2 that interchanges i and j. A *singular* transformation t, mapping i to $it = j \neq i$ is denoted by $(i \to j)$ and has $ht = h$ for all $h \neq i$. The *identity* transformation of Q is denoted by $\mathbf{1}_Q$.

The set of all $n!$ permutations of a finite set $Q = \{0, \ldots, n-1\}$ of n elements is isomorphic to the symmetric group of degree n. The set of all n^n transformations of Q is a semigroup under composition, in fact, a monoid $\mathcal{T}_Q$. The following results are well-known:

Proposition 1 (Permutations) *For $n \geqslant 3$, the set of all $n!$ permutations of the set $Q = \{0, \ldots, n-1\}$ is generated by a cycle $(0, \ldots, n-1)$ of length n and a transposition (i,j), where $i, j \in Q$. One generator is not enough.*

Proposition 2 (Transformations) *For $n \geqslant 3$, the set of all n^n transformations of the set $Q = \{0, \ldots, n-1\}$ is generated by a cycle of length n, a transposition (i,j), and a singular transformation $(k \to l)$, where $i, j, k, l \in Q$. Fewer than three generators do not suffice.*

Every word w in Σ^+ performs a transformation of the set of states of a DFA defined by $q \to \delta(q, w)$. The set of all such transformations is the *transition semigroup* of the DFA [21]. The syntactic semigroup of a language L is isomorphic to the transition semigroup of the minimal DFA of L [21], and this transition semigroup is normally used to represent the syntactic semigroup.

3. Conditions for the Complexity of Languages

Consider now a stream $(L_n \mid n \geqslant k)$ of languages. If a language L_n is most complex, what properties should it have? Below are some suggested answers to this question.

3.1. *Properties of a Single Language*

A0: The (state/quotient) complexity of $L_n \subseteq \Sigma^*$ should be n. It is assumed that the complexity of the language is fixed at some integer $n \geqslant 1$, and all the other properties are expressed in terms of n.

A1: The syntactic semigroup of L_n should have cardinality n^n. Since there are n^n possible transformations of a set of n elements, n^n is an upper bound on the size of the syntactic semigroup of L_n. It was first noted without proof by Maslov [16] in 1970 that n^n is a tight bound; the proof follows from Proposition 2.

The following result was shown recently by Brzozowski and Davies [3]:

Proposition 3 (Syntactic Semigroup and Complexity of Atoms) *Let $\mathcal{D}$ be a minimal DFA with n states accepting a language L. If the transition semigroup of $\mathcal{D}$ has n^n elements, then L has 2^n atoms and the quotient complexities of these atoms meet the bounds given in **A4** below.*

A2: The complexity of each quotient of L_n should be n. The complexity of each quotient is bounded from above by n, because the DFA $\mathcal{D} = (Q, \Sigma, \delta, q_0, F)$ that defines L_n also defines the quotient $w^{-1}L_n$ for any word $w \in \Sigma^*$, if its initial state is changed to $\delta(q_0, w)$. This requirement is met by every language accepted by a strongly connected DFA.

Condition **A2** is implied by **A1**: If **A1** holds, the transition semigroup of $\mathcal{D}$ contains all possible transformations and so $\mathcal{D}$ is strongly connected.

A3: The number of atoms of L_n should be 2^n. It is reasonable that L_n should have the maximal number of building blocks. In view of Proposition 3, this condition need not be checked if **A1** holds.

A4: The complexity of each atom of L_n should be maximal. It was proved in [7] that the complexity of the atoms with 0 or n complemented quotients is bounded from above by $2^n - 1$, and the complexity of any atom with r complemented quotients, where $1 \leqslant r \leqslant n-1$, by

$$f(n,r) = 1 + \sum_{k=1}^{r} \sum_{h=k+1}^{n-r+k} \binom{n}{h}\binom{h}{k}. \tag{2}$$

It was also shown in [7] that these bounds are tight. Again, it seems reasonable to expect that the building blocks of a language should be as complex as possible. By Proposition 3, it is not necessary to verify **A4** if **A1** holds.

3.2. *Unary Operations*

B1: The complexity of the reverse of L_n should be 2^n. It follows from the 1959 subset construction of Rabin and Scott [22] that the upper bound on this complexity is 2^n. It was first shown by Mirkin [17] in 1966 that this bound can be met. Salomaa, Wood, and Yu [24] showed the following result:

Proposition 4 (Transformations and Reversal) *Let $\mathcal{D}$ be a minimal DFA with n states accepting a language L. If the transition semigroup of $\mathcal{D}$ has n^n elements, then the quotient complexity of L^R is 2^n.*

In view of Proposition 4, **B1** needs not be checked if **A1** holds.

B2: The complexity of the star of L_n should be $2^{n-1} + 2^{n-2}$. It was first noted without proof by Maslov [16] in 1970 that this is a tight upper bound. A proof was provided by Yu, Zhuang and Salomaa [27] in 1994.

3.3. *Binary Operations*

Two types of binary operations are examined next: boolean operations and product (concatenation or catenation). By *boolean operation* we mean any one of the following operations: union ($\cup$), symmetric difference ($\oplus$), intersection ($\cap$) and difference ($\setminus$); these operations are chosen because the complexity of every other binary

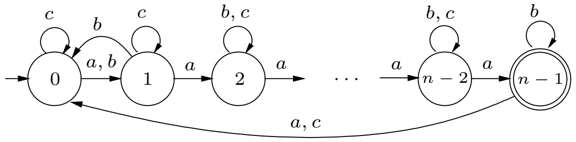


Fig. 1. DFA $\mathcal{U}_n(a,b,c)$ of witness language $U_n(a,b,c)$.

boolean operation can be obtained from the complexities of these four. Denote by $K_m \circ L_n$ any one of these four operations.

C1: The complexity of $K_m \circ L_n$ should be mn. The upper bound for the boolean operations is mn, since $w^{-1}(K_m \circ L_n) = (w^{-1}K_m) \circ (w^{-1}L_n)$. That the bound is tight for union was noted without proof by Maslov [16] in 1970, and proved for both union and intersection by Yu, Zhuang and Salomaa [27] in 1994. Symmetric difference and difference were treated in 2010 in [1].

C2: The complexity of the product K_mL_n should be $(m-1)2^n + 2^{n-1}$. Maslov [16] stated without proof in 1970 that this bound is tight, and Yu, Zhuang and Salomaa [27] provided a proof in 1994.

4. The Main Theorem

The following convention is used: If $\mathcal{X}$ is a DFA, then X is the language accepted by $\mathcal{X}$ and, if X is a regular language, then $\mathcal{X}$ is the minimal DFA accepting X.

The language stream and its minimal DFA that turns out to be the universal witness for all the properties and operations listed above is defined next. DFAs of this type have already appeared in the work of Lupanov [15] in 1963, Mirkin [17] in 1966, and Moore [18] in 1971.

Definition 5. *For $n \geqslant 3$, let $\mathcal{U}_n(a,b,c) = (Q, \Sigma, \delta, q_0, F)$, where $Q = \{0, \dots, n-1\}$ is the set of states*[b]*, $\Sigma = \{a,b,c\}$ is the alphabet, $\delta(q,a) = q+1 \pmod{n}$, $\delta(0,b) = 1$, $\delta(1,b) = 0$, $\delta(q,b) = q$ for $q \in \{2,3,\dots,n-1\}$, $\delta(n-1,c) = 0$, $\delta(q,c) = q$ for $q \in \{0,1,\dots,n-2\}$, $q_0 = 0$ is the initial state, and $F = \{n-1\}$ is the set of final states. See Figure 1. Let $U_n(a,b,c)$ be the language accepted by $\mathcal{U}_n(a,b,c)$.*

Note that in $\mathcal{U}_n$ a performs the cyclic permutation $(0,\dots,n-1)$, b, the transposition $(0,1)$ and c, the singular transformation $(n-1 \to 0)$.

A language $K \subseteq \Sigma^*$ is *permutationally equivalent* to a language $L \subseteq \Sigma^*$ if K can be obtained from L by permuting the letters of Σ. For example, let π be the permutation $\pi(a) = b$, $\pi(b) = c$ and $\pi(c) = a$; then $\pi(a(b^* \cup cc)) = b(c^* \cup aa)$. Similarly, let $\mathcal{K} = \mathcal{L}(\pi(a), \pi(b), \pi(c))$ be the DFA obtained from $\mathcal{L}(a,b,c)$ by changing the roles of the inputs according to permutation π. Then $\mathcal{K}$ is *permutationally*

[b] Although Q, δ, and F depend on n, this dependence is not shown to keep the notation simple.

equivalent to $\mathcal{L}$. In such cases, K ($\mathcal{K}$) is essentially the same language (DFA) as L ($\mathcal{L}$), except that its inputs have been renamed. If two languages are permutationally equivalent, then they have the same single-language complexity properties, and the same complexities of unary operations.

Specifically, let $\mathcal{U}_n(b,a,c)$ be the DFA obtained from $\mathcal{U}_n(a,b,c)$ by interchanging the roles of the inputs a and b. For some operations input c is not needed; then let $\mathcal{U}_n(a,b)$ be the DFA of Definition 5 restricted to inputs a and b, and let $U_n(a,b)$ be the language recognized by this binary DFA. Also, $\mathcal{U}_n(a)$ and $U_n(a)$ are $\mathcal{U}_n(a,b,c)$ and $U_n(a,b,c)$ restricted to a.

Theorem 6 (Universal Witness) *The stream $(U_n(a,b,c) \mid n \geqslant 3)$ meets conditions **A0–A4**, **B1**, **B2**, **C1** if $m \neq n$, and **C2**, whereas **C1** with no restrictions on m and n is met by two permutationally equivalent streams $(U_m(a,b,c) \mid m \geqslant 3)$ and $(U_n(b,a,c) \mid n \geqslant 3)$. Moreover,*

- ***A0** and **A2** are met by $(U_n(a) \mid n \geqslant 3)$.*
- ***B2** is met by $(U_n(a,b) \mid n \geqslant 3)$.*
- ***C1** in general is met by $(U_m(a,b) \mid m \geqslant 3)$ and $(U_n(b,a) \mid n \geqslant 3)$.*
- ***C1** with $m \neq n$ is met by $(U_m(a,b) \mid m \geqslant 3)$ and $(U_n(a,b) \mid n \geqslant 3)$.*

These claims are discussed in Sections 5 and 6. It is pointed out where some of the claims have been proved, and the remaining claims are demonstrated below.

5. Properties of a Single Language and Unary Operations

Conditions **A0** and **A1** are now briefly discussed for the language $U_n(a,b,c)$.

A0 Complexity of the Language: $U_n(a)$ has n quotients because DFA $\mathcal{U}_n(a)$ is minimal. This holds since state i accepts a^{n-1-i} and no other state accepts this word, for $0 \leqslant i \leqslant n-1$; hence no two states are equivalent.

A1 Cardinality of the Syntactic Semigroup: By Proposition 2, the syntactic semigroup of $U_n(a,b,c)$ has cardinality n^n, since inputs a, b and c generate all possible transformations of Q.

It now follows from Propositions 3 and 4 that our witness also satisfies **A2**, **A3**, **A4** and **B1**. Next, a proof is given for **B2**.

B2 Star: The following uses the well-known construction of an ε-NFA to accept the Kleene star of a regular language accepted by a DFA. The language $(U_n(a,b))^*$ is accepted by the ε-NFA $\mathcal{N}_n = (Q_\mathcal{N}, \{a,b\}, \delta_\mathcal{N}, \{s\}, \{s, n-1\})$, where $Q_\mathcal{N} = Q \cup \{s\}$, $s \notin Q$, $\delta_\mathcal{N}(s,a) = \delta_\mathcal{N}(s,b) = \{1\}$, $\delta_\mathcal{N}(q,x) = \{\delta(q,x)\}$ for all $q \in Q$, $x \in \Sigma$, and $\delta_\mathcal{N}(n-1,\varepsilon) = \{0\}$. The ε-NFA $\mathcal{N}_6$ is shown in Figure 2.

Throughout the paper, the notation $p \xrightarrow{w} q$ means that state q is reachable by word w from state p. Similarly, $P \xrightarrow{w} R$ means that state set R is reachable from state set P by word w.

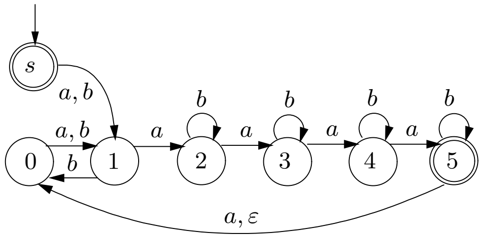


Fig. 2. NFA $\mathcal{N}_6$ for $(U_6(a,b))^*$.

Theorem 7 (Star) *For $n \geqslant 3$, the complexity of $(U_n(a,b))^*$ is $2^{n-1}+2^{n-2}$.*

Proof. To get the complexity of $(U_n(a,b))^*$ one applies the subset construction to the ε-NFA $\mathcal{N}_n$. It will be proved that $\{s\}$, all 2^{n-1} subsets of Q containing 0, and all $2^{n-2}-1$ non-empty subsets of $\{1,\dots,n-2\}$ are reachable and pairwise distinguishable, giving the DFA of $(U_n(a,b))^*$ a total of $2^{n-1}+2^{n-2}$ states.

Since s is the initial state, $\{s\}$ is reachable by ε, and $\{0\}$ by ab. It will be shown how to reach the remaining sets from $\{0\}$. Note that any subset containing $n-1$ must also contain 0.

First it is proved that all 2^{n-1} subsets of Q containing 0 are reachable. Since

$$\{0\} \xrightarrow{a^{n-1}} \{0,n-1\} \xrightarrow{a} \{0,1\} \xrightarrow{(ab)^{i-1}} \{0,i\}$$

for $2 \leqslant i \leqslant n-2$, all two-element subsets of Q containing 0 are reachable.

For $k \geqslant 2$, if any k-element set with 0 can be reached, then so can be any $(k+1)$-element set with 0 and $n-1$, for if $0 < i_1 < i_2 < \dots < i_{k-1} \leqslant n-1$, then

$$\{0, i_2-i_1, \dots, i_{k-1}-i_1, n-1-i_1\} \xrightarrow{a^{i_1}} \{0, i_1, i_2, \dots, i_{k-1}, n-1\}.$$

For $k \geqslant 3$, if any k-element set containing 0 and $n-1$ can be reached, then so can be any k-element set containing 0. This holds because

$$\{0, i_2-i_1, \dots, i_{k-1}-i_1, n-1\} \xrightarrow{a(ab)^{i_1-1}} \{0, i_1, \dots, i_{k-1}\}.$$

It follows now that all 2^{n-1} subsets of Q containing 0 are reachable. Since also

$$\{0, i_2-i_1, \dots, i_k-i_1\} \xrightarrow{a^{i_1}} \{i_1, i_2, \dots, i_k\}$$

for $i_k \leqslant n-2$, all the $2^{n-2}-1$ non-empty subsets of $\{1,\dots,n-2\}$ are reachable.

It remains to prove that all subsets are pairwise distinguishable. Set $\{s\}$ and any subset of Q containing $n-1$ differ from any subset of Q not containing $n-1$, because the former accept the empty word. Also, $\{s\}$ differs from any subset of Q containing $n-1$, because the latter accepts b. Finally, if set P contains i with $0 \leqslant i < n-1$ but set R does not, then P accepts a^{n-1-i}, and R does not. □

Since the required number of subsets can be reached by words in $\{a,b\}^*$, and these subsets are pairwise distinguishable by words in $\{a,b\}^*$, it follows that the complexity of $(U_n(a,b,c))^*$ with the added input c is also $2^{n-1}+2^{n-2}$.

For $n=1$, there are only two languages, $\emptyset$ and Σ^*. The complexity of $\emptyset^* = \varepsilon$ is 2, and that of $(\Sigma^*)^* = \Sigma^*$ is 1; the bound does not apply here.

For $n=2$, the language of Definition 5 is well defined, but inputs a and b coincide. The star of $U_2(a,c)$ has complexity 2 only; hence $U_2(a,b,c)$ is not most complex here. However, the bound $2^1+2^0=3$ is met by the language over $\{a,b\}$ of all the words with an odd number of a's [27].

6. Binary Operations

Next we examine the binary operations from the set $\{\cup, \oplus, \cap, \setminus\}$. The case where the complexities m and n of the two arguments are arbitrary is considered first.

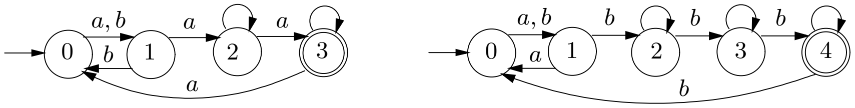
6.1. *C1 Boolean Operations in General*

Since $K_n \cup K_n = K_n \cap K_n = K_n$, and $K_n \setminus K_n = K_n \oplus K_n = \emptyset$, two different languages have to be used to reach the bound mn if $m=n$. It turns out that the streams $(U_n(a,b) \mid n \geqslant 3)$ and the permutationally equivalent stream $(U_n(b,a), n \geqslant 3)$ are witnesses. Figure 3 shows the DFAs $\mathcal{U}_4(a,b)$ and $\mathcal{U}_5(b,a)$. The direct product of $\mathcal{U}_4(a,b)$ and $\mathcal{U}_5(b,a)$ is in Figure 4.

Theorem 8 (Boolean Operations) *The complexity of $U_m(a,b) \circ U_n(b,a)$ is mn for $m,n \geqslant 3$.*

Proof. Consider the direct product DFA of $\mathcal{U}_m(a,b)$ with $\mathcal{U}_n(b,a)$. In that DFA, $(0,0)$ is the initial state. Since $(0,0) \xrightarrow{(ab)^j a^i} (i,j+1)$, for $i \geqslant 0, j \geqslant 1$, all the states in columns 2 to $n-1$ are reachable. Since $(0,0) \xrightarrow{(ba)^i b^j} (i+1,j)$, for $i \geqslant 1, j \geqslant 0$, all the states in rows 2 to $m-1$ are reachable. Also $(0,0) \xrightarrow{a} (1,1)$, $(m-1,0) \xrightarrow{a} (0,1)$, and $(0,n-1) \xrightarrow{b} (1,0)$. Hence all states are reachable.

It remains to prove that all the states are pairwise distinguishable. Let H (for *horizontal*) be the set $H = \{(m-1,0),(m-1,1),\dots,(m-1,n-2)\}$, and let V (for



$\mathcal{U}_4(a,b)$ $\mathcal{U}_5(b,a)$

Fig. 3. DFAs of $U_4(a,b)$ and $U_5(b,a)$.

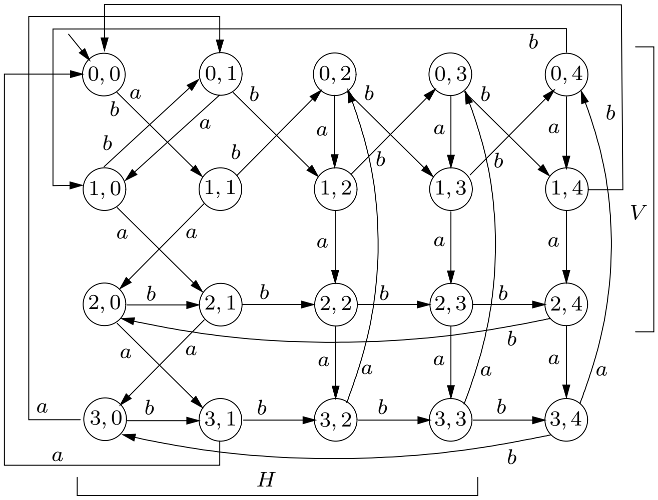


Fig. 4. Direct product of $\mathcal{U}_4(a,b)$ with $\mathcal{U}_5(b,a)$.

vertical) be $V = \{(0, n-1), (1, n-1), \ldots, (m-2, n-1)\}$. The boolean operations are now considered one by one.

Union: The final states are $H \cup V \cup \{(m-1, n-1)\}$. Two non-final states in different rows (columns) can be distinguished by a word in a^* (b^*). Two distinct final states in H (V) go to two distinct non-final states by a (b). Any state from H is distinguished from any state in V by a. Finally, $(m-1, n-1)$ is the only final state that accepts both a and b. Hence all states are distinguishable.

Symmetric Difference: The final states are those in $H \cup V$. The final states are all distinguishable by the argument used for union. The non-final states other than $(m-1, n-1)$ are distinguishable by the same words as for union. State $(m-1, n-1)$ accepts both ab^n and ba^m, and no state other than $(m-2, n-2)$ accepts both of these words. But $(m-1, n-1)$ and $(m-2, n-2)$ can be distinguished as follows: If $m = 3$ and $n \geqslant 3$, then $(m-1, n-1)$ rejects ba, while $(m-2, n-2)$ accepts it. If $m \geqslant 3$ and $n = 3$, then $(m-1, n-1)$ rejects ab, while $(m-2, n-2)$ accepts it. For $m, n > 3$, $(m-1, n-1)$ rejects aba, while $(m-2, n-2)$ accepts it. So all non-final states are also distinguishable.

Intersection: For intersection, there is only one final state $(m-1, n-1)$. Non-final states q and words w_q accepted only by those states are listed below:

(1) $q = (0, j)$ with $n-1-j$ even, $w_q = b^{n-1-j}a^{m-1}$,
(2) $q = (0, j)$ with $n-1-j$ odd, $w_q = b^{n-1-j}a^{m-2}$,
(3) $q = (1, j)$ with $n-1-j$ even, $w_q = b^{n-1-j}a^{m-2}$,
(4) $q = (1, j)$ with $n-1-j$ odd, $w_q = b^{n-1-j}a^{m-1}$,
(5) for $i \geqslant 2$, $q = (i, j)$, $w_q = b^{n-1-j}a^{m-1-i}$.

Difference: For difference, the final states are those in H.

State $(m-1, j)$ rejects b^{n-1-j}, but other final states accept it. So all final states are distinguishable.

Now consider non-final states $p = (i, j)$ and $q = (h, l)$.

(1) If $i > h$ and $j \neq n-1$, then a^{m-1-i} distinguishes p and q. The case $h > i$ and $l \neq n-1$ is symmetric.
(2) If $i > h$ and $j = n-1$, then $a^{m-1-i}b$ distinguishes p and q. The case $h > i$ and $l = n-1$ is symmetric.
(3) If $i = h$ and $j > l$, then $b^{n-1-j}a^{m-1-i}$ distinguishes p and q. The case $i = h$ and $l > j$ is symmetric. □

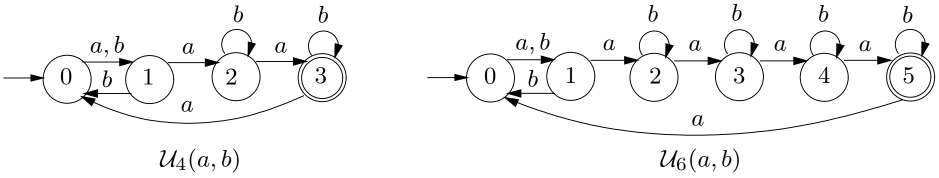
6.2. *C1 Boolean Operations with* $\boldsymbol{m \neq n}$

Although it is impossible for the stream $(U_n(a, b), n \geqslant 3)$ to meet the bound for boolean operations when $m = n$, this stream is as complex as it could possibly be as is shown below. DFAs $\mathcal{D}_1 = \mathcal{U}_4(a, b)$ and $\mathcal{D}_2 = \mathcal{U}_6(a, b)$ are shown in Figure 5. Their direct product $\mathcal{P}$, shown in Figure 6, serves as a basis for all four operations. The following result was conjectured in [2]; the proof is due to Brzozowski and Liu:

Theorem 9 ($\boldsymbol{K_m \circ L_n}$, $\boldsymbol{m \neq n}$) *For $m, n \geqslant 3$ and $m \neq n$, the complexity of $U_m(a, b) \circ U_n(a, b)$ is mn.*

Proof. Consider the direct product of $U_m(a, b)$ with $U_n(a, b)$. It will be shown that all mn states of the direct product are reachable from the initial state $(0, 0)$. Without loss of generality, assume that $m < n$. We have $(0, 0) \xrightarrow{a^m} (0, m) \xrightarrow{(ab)^{n-1-m}a} (1, 0)$. For $1 \leqslant i \leqslant m-2$, ab takes $(i, 0)$ to $(i+1, 0)$; hence all states in column 0 can be reached. State (i, j) can be reached from state $(i - j \pmod{m}, 0)$ by a^j. Therefore all the states are reachable.

It remains to prove that all the states are pairwise distinguishable. Given a state (i, j), we define $d_{i,j}$ to be the minimal integer such that $a^{d_{i,j}}$ takes (i, j) to a final state, or infinity, if no final state is reachable by a's from (i, j); note that $d_{i,j} = 0$ if and only if (i, j) is final. The boolean operations are now considered one by one.



$\mathcal{U}_4(a, b)$ $\mathcal{U}_6(a, b)$

Fig. 5. DFAs $\mathcal{U}_4(a, b)$ and $\mathcal{U}_6(a, b)$.

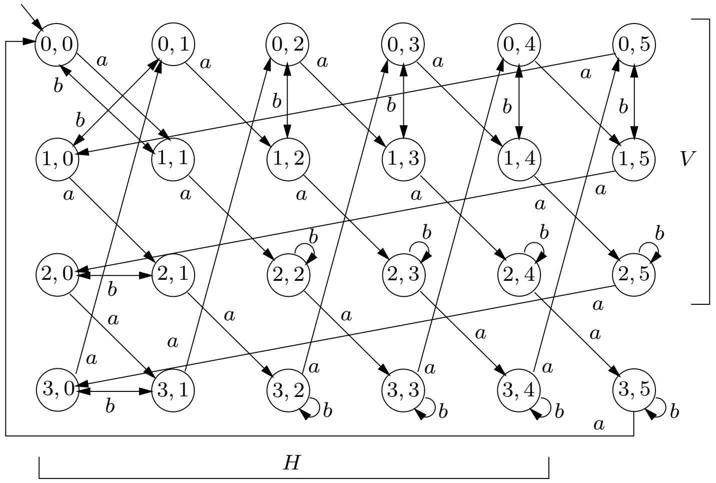


Fig. 6. Direct product $\mathcal{P}$ of $\mathcal{D}_1 = \mathcal{U}_4(a,b)$ with $\mathcal{D}_2 = \mathcal{U}_6(a,b)$.

Union: The final states are those in $H \cup V \cup \{(m-1, n-1)\}$. We have here $d_{i,j} = \min\{m-1-i, n-1-j\} \leqslant m-1$.

Let (i,j) and (k,l) be two distinct states, with $d_{i,j} \leqslant d_{k,l}$. If $d_{i,j} < d_{k,l}$, then the two states are distinguished by $a^{d_{i,j}}$. If $d_{i,j} = d_{k,l} = d$, apply a^{d+1} to both states. The resulting states must be distinct and each must have at least one zero component.

If the two states are of the form $(0, n-1-g)$ and $(0, n-1-h)$, $h < g$, then $(ab)^h$ distinguishes them. A symmetric argument works for $(m-1-g, 0)$ and $(m-1-h, 0)$. Suppose now the states are $(0, n-1-g)$ and $(m-1-h, 0)$. If $g \neq h$, then the states are distinguished by $(ab)^{\min\{g,h\}}$. If $g = h$, then applying $(ab)^{g+1}$ results in the two states $(1,0)$ and $(0,1)$. Since $d_{1,0} < d_{0,1}$ (because $m < n$), these two states are distinguished by $d_{1,0}$.

Symmetric Difference: The final states are those in $H \cup V$.

The removal of $(m-1, n-1)$ from the set of final states causes all of the $d_{i,j}$ to increase by m when $m - i = n - j$, and leaves the rest unchanged. Since all of the other $d_{i,j}$ are at most $m-1$, and the change maps distinct $d_{i,j}$ to distinct $d'_{i,j}$, the same argument for unequal $d_{i,j}$ applies to all pairs involving at least one of the states affected by the change. Since state $(m-1, n-1)$ was never used to distinguish equal $d_{i,j}$ cases in union, all remaining equality cases can be dealt with in the same way as in union.

Difference: The final states are those in H.

In this case only, we do not assume $m < n$. The $d_{i,j}$ here are as follows: $d_{i,j} = m-1-i$ if $m-i \neq n-j$, and otherwise $d_{i,j} = 2m-1-i$. The same distinguishability argument applies when $d_{i,j} \neq d_{k,l}$. Suppose $d_{i,j} = d_{k,l}$. Then $i = k$, and hence

$j \neq l$. Apply a^{m-i} to get two distinct states $(0,g)$ and $(0,h)$, $g \neq 0$. As repeated applications of ab cycle through states $(0,1),(0,2),\ldots,(0,n-1)$, there exists a d such that $(ab)^d$ sends $(0,g)$ to $(0,n-m)$, and $(0,h)$ to a different state. Therefore applying $(ab)^d a^{m-1}$ maps $(0,g)$ to a non-final state, and $(0,h)$ to a final state.

Intersection: The only final state is $(m-1,n-1)$.

We assume that $m < n$. If $\gcd(m,n) = 1$, then by the Chinese Remainder Theorem there is a bijection between the integers $\{0,1,\ldots,mn-1\}$ and the states of the direct product given by $k \leftrightarrow (k \pmod m, k \pmod n)$. Applying a to the state corresponding to k results in the state corresponding to $k+1$. Thus, for state (i,j) corresponding to k, $d_{i,j} = mn-1-k$; hence all states are distinguishable by multiple applications of a.

Now suppose $\gcd(m,n) > 1$. The states which can reach $(m-1,n-1)$ through multiple applications of a are exactly those which can be written in the form $(k \pmod m, k \pmod n)$ for some integer k. Let S denote the set of these states. Any two states in S have different finite values of $d_{i,j}$, and hence are distinguishable.

Let $(i,j),(k,l) \notin S$; that is, $d_{i,j} = d_{k,l} = \infty$. These states can be distinguished from states in S using only a's. Suppose $i \neq k$. Apply a^{m-i} to get two distinct states $(0,j')$ and (k',l'), $k' \neq 0$. Since $(0,j') \notin S$, $j' \neq 0$. As $m < n$ and $(0,m) \in S$, there exists a d such that applying $(ab)^d$ to $(0,j')$ results in $(0,m)$. Then let d be the minimal integer such that applying $(ab)^d$ to the two states results in at least one state in S. Because the two resulting states are distinct, they must be distinguishable. □

6.3. *C2 Product*

It is shown next that the complexity of the product of $U_m(a,b,c)$ with $U_n(a,b,c)$ reaches the maximal possible bound.

To avoid confusion of states, let $\mathcal{U}_m = \mathcal{U}_m(a,b,c) = (Q_m, \Sigma, \delta_m, q_0, \{q_{m-1}\})$, where $Q_m = \{q_0,\ldots,q_{m-1}\}$, and let $\mathcal{U}_n = \mathcal{U}_n(a,b,c)$, as in Definition 5. Define the ε-NFA $\mathcal{N} = (Q_m \cup Q_n, \Sigma, \delta_{\mathcal{N}}, \{q_0\}, \{n-1\})$, where $\delta_{\mathcal{N}}(q,a) = \{\delta_m(q,a)\}$ if $q \in Q_m$, $a \in \Sigma$, $\delta_{\mathcal{N}}(q,a) = \{\delta_n(q,a)\}$ if $q \in Q_n$, $a \in \Sigma$, and $\delta_{\mathcal{N}}(q_{m-1},\varepsilon) = \{0\}$. This ε-NFA accepts $U_m(a,b,c) \cdot U_n(a,b,c)$, and is illustrated in Figure 7 for $m = 4$ and $n = 5$.

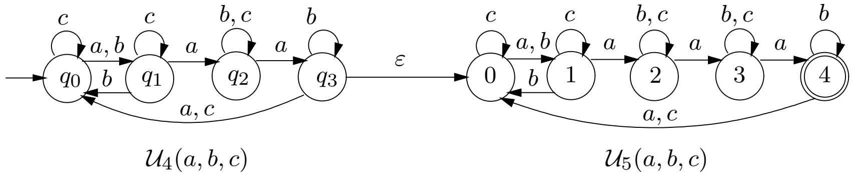


Fig. 7. ε-NFA $\mathcal{N}$ of $U_4(a,b,c) \cdot U_5(a,b,c)$.

Theorem 10 (Product) *For $m \geqslant 3$, $n \geqslant 2$, the complexity of the product $U_m(a,b,c) \cdot U_n(a,b,c)$ is $(m-1)2^n + 2^{n-1}$.*

Proof. It will be shown that all $(m-1)2^n$ subsets of states of $\mathcal{N}$ of the form $\{q_i\} \cup S$, where $i < m-1$ and S is any subset of Q_n, are reachable, as well as all 2^{n-1} subsets of the form $\{q_{m-1}, 0\} \cup S$, where S is any subset of $\{1, \dots, n-1\}$. *All the arithmetic below is modulo n.*

First, study how states of the form $\{q_0\} \cup S$ can be reached. Since $\{q_0\}$ is the initial set of states, it is reached by ε. Sets $\{q_i\}$ are reached from $\{q_0\}$ by a^i, for $i = 1, \dots, m-2$, and $\{q_{m-1}, 0\}$, by a^{m-1}. From $\{q_{m-1}, 0\}$, $\{q_0, 0\}$ is reached by c, and $\{q_0, 1\}$ by a. From $\{q_0, 1\}$, $\{q_0, i\}$ is reached by $(ab)^{i-1}$, for $i = 2, \dots, n-1$. Hence all the sets of the form $\{q_0\} \cup S$, where $|S| \leqslant 1$ are reachable.

Second, it will be shown that, if $\{q_{m-1}, 0\} \cup S$ can be reached for all sets $S \subseteq \{1, \dots, n-1\}$ with $|S| = k \geqslant 0$, then $\{q_0\} \cup T$ can be reached for all $T = \{t_0, t_1, \dots, t_k\} \subseteq \{0, \dots, n-1\}$ with $0 \leqslant t_0 < t_1 < \cdots < t_k \leqslant n-1$. There are two cases to consider:

(1) $t_0 = 0$: Use $\{q_{m-1}, 0, t_2 - t_1, \dots, t_k - t_1, n-1\} \xrightarrow{a(ab)^{t_1 - 1}} \{q_0, t_1, t_2, \dots, t_k, 0\}$.

(2) $t_0 > 0$: Use $\{q_{m-1}, 0, t_1 - (t_0 - 1), \dots, t_k - (t_0 - 1)\} \xrightarrow{a(ab)^{t_0 - 1}} \{q_0, t_0, t_1, \dots, t_k\}$.

Third, consider sets $\{q_{m-1}, 0\} \cup S$, $S \subseteq \{1, \dots, n-1\}$. It has already been shown that $\{q_{m-1}, 0\}$ is reachable. Suppose that all the sets of the form $\{q_0\} \cup S$ with $|S| = k \geqslant 1$, $0 \notin S$ can be reached. Then to reach $\{q_{m-1}, 0, t_1, \dots, t_k\}$ with $1 \leqslant t_1 < \cdots < t_k \leqslant n-1$, use $\{q_0, t_1 - (m-1), \dots, t_k - (m-1)\} \xrightarrow{a^{m-1}} \{q_{m-1}, 0, t_1, \dots, t_k\}$.

Fourth, for $0 < i < m-1$, $0 \leqslant t_1 < \cdots < t_k \leqslant n-1$, $\{q_i, t_1, \dots, t_k\}$ is reached by a^i from $\{q_0, t_1 - i, \dots, t_k - i\}$. Hence all the required states can be reached.

It will now be proved that all these subsets are pairwise distinguishable. Consider $s = \{q_i\} \cup S$ and $t = \{q_j\} \cup T$, where $0 \leqslant i, j \leqslant m-1$ and $S \neq T$, $S, T \subseteq Q_n$. If k is in $S \oplus T$, then a^{n-1-k} distinguishes s and t.

Next suppose $s = \{q_i\} \cup S$ and $t = \{q_j\} \cup S$ with $i < j < m-1$. Applying $(ca)^{m-1-j}$ sends $t = \{q_j\} \cup S$ to $t' = \{q_{m-1}, 0\} \cup S'$ for some $S' \subseteq \{1, \dots, n-1\}$, but sends $s = \{q_i\} \cup S$ to $s' = \{q_{i+m-1-j}\} \cup S'$, and this pair can be distinguished since the subsets of Q_n are different. If $i > 0$ and $j = m-1$, apply $(ca)^{m-1-i}$. Then $s = \{q_i\} \cup S$ is sent to $s' = \{q_{m-1}, 0\} \cup S'$, and $t = \{q_{m-1}\} \cup S$ is sent to $t' = \{q_k\} \cup S'$ for some $S' \subseteq \{1, \dots, n-1\}$ and $k < m-1$.

This leaves the case where $i = 0$ and $j = m-1$. Then use ba to send $t = \{q_j\} \cup S$ to $t' = \{q_0\} \cup S'$ and $s = \{q_i\} \cup S$ to $s' = \{q_2\} \cup S'$. Now $(ca)^{m-3}$ can be applied to make the subsets of Q_n different.

Since all reachable sets are pairwise distinguishable, the bound is met. □

The restrictions of U_n to two letters do not meet the bound for product, although there do exist binary witnesses [16].

7. Combined Operations

A *combined operation* is one that involves at least two basic operations; $K \cup L^*$ is an example. Although the witness $U_n(a, b, c)$ works for quite a few combined operations, it does not apply in all cases, and other approaches may be needed.

The extension of $U_n(a, b, c)$ to $U_n(a, b, c, d)$, where d performs the identity transformation $\mathbf{1}_Q$, has considerable merit as will be seen below. There is also some evidence that $U_n(a, b, c, d, e)$, where e performs the cycle $(1, \dots, n-1)$, may be useful. However, extending the alphabet still does not cover all the cases; hence, the following was proposed in [2]:

Definition 11. *A* dialect *of $U_n(a, b, c)$ is any ternary language $U'_n(a, b, c)$ of complexity n, in which a performs a cyclic permutation of the n states in the minimal DFA of U'_n, b performs a transposition, and c is a singular transformation. By convention, the initial state of the minimal DFA of a dialect is 0, but the set of final states is arbitrary, as long as the DFA remains minimal. A* dialect *of $U_n(a, b, c, d)$ with d performing $\mathbf{1}_Q$ is $U'_n(a, b, c, d)$, where $U'_n(a, b, c)$ is a dialect of $U_n(a, b, c)$.*

In [2], numerous conjectures were made about the complexity of combined operations. Since then, Brzozowski and Liu [4, 5] proved many of these conjectures.

7.1. *Single Operations Combined with Reversal*

The first group of combined operations studied by Brzozowski and Liu [4] involves boolean operations and product with one or two reversed arguments, and also $(L^R)^*$. Eight of these operations were previously studied in five papers:

- $K_m \cup L_n^R$ and $K_m \cap L_n^R$ by Gao and Yu [12];
- $K_m^R \cup L_n^R$, $K_m^R \cap L_n^R$, and $(K_m L_n)^R$ (upper bound only) by Liu, Martin-Vide, A. Salomaa, and Yu [14];
- $K_m L_n^R$ by Cui, Gao, Kari and Yu [9];
- $K_m^R L_n$ and $(K_m L_n)^R$ (lower bound) by Cui, Gao, Kari and Yu [8];
- $(L^*)^R$ by Gao, K. Salomaa, and Yu [11].

Brzozowski and Liu added the difference and symmetric difference with one or two reversed arguments, for a total of 13 operations. For these 13 operations the following universal witnesses and their dialects were found [4] for $m, n \geqslant 3$:

(1) $U_m(a, b, c)$ and $U_n(a, b, c)$ for $K_m \cup L_n^R$, $K_m \cap L_n^R$, $K_m \setminus L_n^R$, $K_m \oplus L_n^R$, $L_n^R \setminus K_m$, and $K_m L_n^R$. Here the same stream is used for both arguments.
(2) $U_{\{0\},n}(a, b, c)$ for $(L^R)^* = (L^*)^R$. The set of final states is changed to $\{0\}$.
(3) $U_m(a, b, c)$ and $U_n(b, a, c)$ for $K_m^R \cup L_n^R$, $K_m^R \cap L_n^R$, $K_m^R \setminus L_n^R$, and $K_m^R \oplus L_n^R$, except when $m = n = 4$. Here a and b are permuted in the second argument. The case $m = n = 4$ is included if the sets of final states are changed as follows: Use $U_{\{0,2\},m}(a, b, c)$ for $m \geqslant 3$, $U_{\{1\},3}(b, a, c)$, and $U_{\{1,3\},n}(b, a, c)$ for $n \geqslant 4$.

(4) $U_m(a,b,c,d)$ and $U_n(d,c,b,a)$ for $(K_mL_n)^R = L_n^RK_m^R$. Here the identity transfromation $\mathbf{1}_Q$ performed by d is added, and the inputs are permuted.
(5) $V_m(a,b,c,d)$ and $V_n(d,c,b,a)$ for $K_m^RL_n$. This is the only case where the transition functions of the witnesses needed to be changed. In $\mathcal{V}$, a does $(0,\dots,n-1)$ and d does $\mathbf{1}_Q$ as above, but b does $(n-2,n-1)$, and c does $(n-1\to n-2)$.

These results show that it is efficient to deal with reversed arguments for several operations together, and to consider all four boolean operations at the same time.

7.2. *Single Operations Combined with Star*

The second group of combined operations studied by Brzozowski and Liu [5] involves boolean operations and product with one or two starred arguments. Seven of these operations were previously studied in five papers:

- $K_m \cup L_n^*$ and $K_m \cap L_n^*$ by Gao and Yu [12];
- $K_m^* \cup L_n^*$, $K_m^* \cap L_n^*$ by Gao, Kari, and Yu [10];
- $K_m^*L_n$ by Cui, Gao, Kari and Yu [8];
- $K_mL_n^*$ by Cui, Gao, Kari and Yu [9];
- $(K_mL_n)^*$ by Gao, K. Salomaa and Yu [11].

Brzozowski and Liu added the difference and symmetric difference operations with one or two starred arguments, and the product $K_m^*L_n^*$, for a total of 13 operations. For these 13 operations the following universal witnesses and their dialects were found [5] for $m,n \geqslant 3$:

(1) $U_m(a,b,c)$ and $U_n(b,a,c)$ for $K_m \cup L_n^*$, $K_m \oplus L_n^*$, $L_n^* \setminus K_m$.
(2) $U_{\{0\},m}(a,b,c)$ and $U_n(b,a,c)$ for $K_m \cap L_n^*$ and $K_m \setminus L_n^*$.
(3) $T_m(a,b,c)$ and $T_n(b,a,c)$ for $K_mL_n^*$. In $\mathcal{T}_n$, a does $(0,\dots,n-1)$ and b does $(0,1)$ as before, but c does $(1\to 0)$.
(4) $U_m(a,b,c,d)$ and $U_{\{0\},n}(d,c,b,a)$ for $K_m^*L_n$ and $K_m^*L_n^*$.
(5) $W_m(a,b,c,d)$ and $W_n(d,c,b,a)$ for $K_m^* \cup L_n^*$, $K_m^* \cap L_n^*$ and $(K_mL_n)^*$. In $\mathcal{W}_n$, a does $(0,\dots,n-1)$ and d is $\mathbf{1}_Q$, but b does $(n-2,n-1)$, and c does $(1\to 0)$.
(6) $W_{\{0,n-1\},m}(a,b,c,d)$ and $W_n(d,c,b,a)$ for $K_m^* \setminus L_n^*$ and $K_m^* \oplus L_n^*$. Here the set of final states is changed in the first argument.

As was the case with reversal, these results show that it is efficient to deal with starred arguments for several operations together, and to consider all four boolean operations at the same time.

In connection with the star, there are four more operations; they are of the form $(K_m \circ L_n)^*$. A. Salomaa, K. Salomaa, and Yu [23] showed that the complexity of $(K_m \cup L_n)^*$ is $2^{m+n-1} - (2^{m-1} + 2^{n-1} - 1)$ with ternary witnesses. Jirásková and Okhotin [13] proved that binary witnesses suffice. In [2] it was shown that dialects $S_{\{0\},m}(a,c)$ and $S_{\{0\},n}(c,a)$ can also be used, where a does $(0,\dots,n-1)$ as before, but b is absent, and c does $(0\to 1)$.

It was also proved in [13] with witnesses over a 6-letter alphabet that the complexity of $(K_m \cap L_n)^*$ is $2^{mn-1} + 2^{mn-2}$. It is possible that $U_m(a,b,c,d,e)$ (defined at the beginning of Section 7) and $U_n(e,c,b,a,d)$ also work, as calculations with small values of m and n indicate.

The following is clear:

Proposition 12 ($(\boldsymbol{K_m \setminus L_n})^*$) *The complexity of $(K_m \setminus L_n)^*$ is $2^{mn-1} + 2^{mn-2}$ for $m, n \geqslant 3$, and it is met by the witnesses K_m and $\overline{L_n}$, where K_m and L_n are the witnesses for intersection.*

The complexity of $(K_m \oplus L_n)^*$ remains open.

7.3. *Other Combined Operations*

Several other combined operations have been studied in the literature. Conjectures were made in [2] about universal witnesses for the following boolean operations combined with product: $(K_mL_n) \circ M_p$, $M_p \setminus (K_mL_n)$, $(K_m \cup L_n)M_p$, $(K_m \cap L_n)M_p$, $K_m(L_n \cup M_p)$, $K_m(L_n \cap M_p)$ and $K_m(L_n \setminus M_p)$. This topic requires further study.

8. Conclusions

It has been shown that the witnesses $U_n(a,b,c)$ and $U_n(a,b,c,d)$ and a handful of their dialects are sufficient for all the basic operations and many combined operations. These witnesses ought to be considered when one is looking at new operations. Although a search is still required to find the appropriate dialects, this search is much simpler than that among all regular languages. It is hoped that these results are a step towards a theory of complexity of regular languages.

Acknowledgments: This research was supported by the Natural Sciences and Engineering Research Council of Canada under grant No. OGP0000871. I am very grateful to David Liu for his contributions to Theorem 9. I am much indebted to the referees for their very careful proofreading and for simplifying some of the proofs.

References

[1] J. Brzozowski, Quotient complexity of regular languages, *J. Autom. Lang. Comb.* **15**(1/2) (2010) 71–89.

[2] J. Brzozowski, In search of the most complex regular languages, *CIAA 2012*, eds. N. Moreira and R. Reis *LNCS* **7381**, (Springer, 2012), pp. 5–24.

[3] J. Brzozowski and G. Davies, Maximal syntactic complexity of regular languages implies maximal quotient complexities of atoms, `http://arxiv.org/abs/1302.3906` (May 2013).

[4] J. Brzozowski and D. Liu, Universal witnesses for state complexity of basic operations combined with reversal, *CIAA 2013*, ed. S. Konstantinidis *LNCS* **7982**, (Springer, 2013), pp. 72–83.

[5] J. Brzozowski and D. Liu, Universal witnesses for state complexity of boolean operations and concatenation combined with star, *DCFS 2013*, eds. H. Jürgensen and R. Reis *LNCS* **8031**, (Springer, 2013), pp. 30–41.

[6] J. Brzozowski and H. Tamm, Theory of átomata, *DLT 2011*, eds. G. Mauri and A. Leporati *LNCS* **6795**, (Springer, 2011), pp. 105–116.

[7] J. Brzozowski and H. Tamm, Quotient complexities of atoms of regular languages, *DLT 2012*, eds. H.-C. Yen and O. H. Ibarra *LNCS* **7410**, (Springer, 2012), pp. 50–61.

[8] B. Cui, Y. Gao, L. Kari and S. Yu, State complexity of combined operations with two basic operations, *Theoret. Comput. Sci.* **437** (2012) 82–102.

[9] B. Cui, Y. Gao, L. Kari and S. Yu, State complexity of two combined operations: catenation-star and catenation-reversal, *Int. J. Found. Comput. Sc.* **23** (2012) 51–66.

[10] Y. Gao, L. Kari and S. Yu, State complexity of union and intersection of star on k regular languages, *Theoret. Comput. Sci.* **429** (2012) 98–107.

[11] Y. Gao, K. Salomaa and S. Yu, The state complexity of two combined operations: star of catenation and star of reversal, *Fund. Inform.* **83**(1–2) (2008) 75–89.

[12] Y. Gao and S. Yu, State complexity of combined operations with union, intersection, star, and reversal, *Fund. Inform.* **116** (2012) 1–12.

[13] G. Jirásková and A. Okhotin, On the state complexity of star of union and star of intersection, *Fund. Inform.* **109** (2011) 1–18.

[14] G. Liu, C. Martin-Vide, A. Salomaa and S. Yu, State complexity of basic language operations combined with reversal, *Inform. and Comput.* **206** (2008) 1178–1186.

[15] O. B. Lupanov, A comparison of two types of finite sources, *Problemy Kibernetiki* **9** (1963) 321–326 (Russian), German translation: Über den Vergleich zweier Typen endlicher Quellen. *Probleme der Kybernetik* **6** (1966), 328–335.

[16] A. N. Maslov, Estimates of the number of states of finite automata, *Dokl. Akad. Nauk SSSR* **194** (1970) 1266–1268 (Russian)., English translation: Soviet Math. Dokl. **11** (1970) 1373–1375.

[17] B. G. Mirkin, On dual automata, *Kibernetika (Kiev)* **2** (1966) 7–10 (Russian), English translation: Cybernetics **2**, (1966) 6–9.

[18] F. R. Moore, On the bounds for state-set size in the proofs of equivalence between deterministic, nondeterministic, and two-way finite automata, *IEEE Trans. Comput.* **C20**(10) (1971) 1211–1214.

[19] J. Myhill, Finite automata and representation of events., *Wright Air Development Center Technical Report* **57–624** (1957).

[20] D. Perrin, Finite automata, *Handbook of Theoretical Computer Science*, ed. J. van Leewen, **B** (Elsevier, 1990), pp. 1–57.

[21] J.-E. Pin, Syntactic semigroups, *Handbook of Formal Languages, vol. 1: Word, Language, Grammar*, (Springer, New York, NY, USA, 1997), pp. 679–746.

[22] M. Rabin and D. Scott, Finite automata and their decision problems, *IBM J. Res. and Dev.* **3** (1959) 114–129.

[23] A. Salomaa, K. Salomaa and S. Yu, State complexity of combined operations, *Theoret. Comput. Sci.* **383** (2007) 140–152.

[24] A. Salomaa, D. Wood and S. Yu, On the state complexity of reversals of regular languages, *Theoret. Comput. Sci.* **320** (2004) 315–329.

[25] S. Yu, Regular languages, *Handbook of Formal Languages, vol. 1: Word, Language, Grammar*, eds. G. Rozenberg and A. Salomaa (Springer, 1997), pp. 41–110.

[26] S. Yu, State complexity of regular languages, *J. Autom. Lang. Comb.* **6** (2001) 221–234.

[27] S. Yu, Q. Zhuang and K. Salomaa, The state complexities of some basic operations on regular languages, *Theoret. Comput. Sci.* **125** (1994) 315–328.