Incomplete Transition Complexity of Basic Operations on Finite Languages *

Eva Maia**, Nelma Moreira, Rogério Reis

CMUP & DCC, Faculdade de Ciências da Universidade do Porto
Rua do Campo Alegre, 4169-007 Porto, Portugal
e-mail:{emaia,nam,rvr}@dcc.fc.up.pt

Abstract. The state complexity of basic operations on finite languages (considering complete DFAs) has been extensively studied in the literature. In this paper we study the incomplete (deterministic) state and transition complexity on finite languages of boolean operations, concatenation, star, and reversal. For all operations we give tight upper bounds for both descriptional measures. We correct the published state complexity of concatenation for complete DFAs and provide a tight upper bound for the case when the *right* automaton is larger than the *left* one. For all binary operations the tightness is proved using family languages with a variable alphabet size. In general the operational complexities depend not only on the complexities of the operands but also on other refined measures.

1 Introduction

Descriptional complexity studies the measures of complexity of languages and operations. These studies are motivated by the need to have good estimates of the amount of resources required to manipulate the smallest representation for a given language. In general, having succinct objects will improve our control on software, which may become smaller and more efficient. Finite languages are an important subset of regular languages with many applications in compilers, computational linguistics, control and verification, etc. [9, 1, 8, 3]. In those areas it is also usual to consider deterministic finite automata (DFA) with partial transition functions. As an example we can mention the manipulation of compact natural language dictionaries using Unicode alphabets. This motivates the study of the transition complexity of DFAs (not necessarily complete), besides the usual state complexity. The operational transition complexity of basic operations on regular languages was studied by Gao *et al.* [4] and Maia *et al.* [7]. In this paper we continue that line of research by considering the class of finite

* This work was partially funded by the European Regional Development Fund through the programme COMPETE and by the Portuguese Government through the FCT under projects PEst-C/MAT/UI0144/2011 and CANTE-PTDC/EIA-CCO/101904/2008.

** Eva Maia is funded by FCT grant SFRH/BD/78392/2011.

languages. For finite languages, Salomaa and Yu [10] showed that the state complexity of the determinization of a nondeterministic automaton (NFA) with m states and k symbols is $\Theta(k^{\frac{m}{1+\log k}})$ (lower than 2^m as it is the case for general regular languages). Câmpeanu *et al.* [2] studied the operational state complexity of concatenation, Kleene star, and reversal. Finally, Han and Salomaa [5] gave tight upper bounds for the state complexity of union and intersection on finite languages. In this paper we give tight upper bounds for the state and transition complexity of all the above operations, for non necessarily complete DFAs with an alphabet size greater than 1 . For the concatenation, we correct the upper bound for the state complexity of complete DFAs [2], and show that if the *right* automaton is larger than the *left* one, the upper bound is only reached using an alphabet of variable size. The transition complexity results are all new, although the proofs are based on the ones for the state complexity and use techniques developed by Maia *et al.* [7]. Table 1 presents a comparison of the transition complexity on regular and finite languages, where the new results are highlighted. Note that the values in the table are obtained using languages for which the upper bounds are reached. All the proofs not presented in this paper can be found in an extended version of this work[1].

Operation	Regular	$\|\Sigma\|$	Finite	$\|\Sigma\|$
$L_1 \cup L_2$	$2n(m+1)$	2	$\mathbf{3(mn\text{-}n\text{-}m)+2}$	$f_1(m,n)$
$L_1 \cap L_2$	nm	1	$\mathbf{(m-2)(n-2)(2+\sum_{i=1}^{\min(m,n)-3}(m-2-i)(n-2-i))+2}$	$f_2(m,n)$
L^C	$m+2$	1	$\mathbf{m+1}$	1
L_1L_2	$2^{n-1}(6m+3)-5$, if $m,n \geq 2$	3	$\mathbf{2^n(m-n+3)-8}$, if $m+1 \geq n$	2
			See Theorem 3 **(4)**	$n-1$
$L^\star$	$3.2^{m-1}-2$, if $m \geq 2$	2	$\mathbf{9 \cdot 2^{m-3} - 2^{m/2} - 2}$, if m is odd	3
			$\mathbf{9 \cdot 2^{m-3} - 2^{(m-2)/2} - 2}$, if m is even	
L^R	$2(2^m-1)$	2	$\mathbf{2^{p+2}-7}$, if $m=2p$	2
			$\mathbf{3 \cdot 2^p - 8}$, if $m = 2p-1$	

Table 1. Incomplete transition complexity for regular and finite languages, where m and n are the (incomplete) state complexities of the operands, $f_1(m,n) = (m-1)(n-1)+1$ and $f_2(m,n) = (m-2)(n-2)+1$. The column $|\Sigma|$ indicates the minimal alphabet size for each the upper bound is reached.

[1] http://www.dcc.fc.up.pt/Pubs/TReports/TR13/dcc-2013-02.pdf

2 Preliminaries

We assume that the reader is familiar with the basic notions about finite automata and regular languages. For more details, we refer the reader to the standard literature [6, 12, 11]. In this paper we consider DFAs to be not necessarily complete, *i.e.* with partial transition functions. The *state complexity* of L ($sc(L)$) is equal to the number of states of the minimal complete DFA that accepts L. The *incomplete state complexity* of a regular language L ($isc(L)$) is the number of states of the minimal DFA, not necessarily complete, that accepts L. Note that $isc(L)$ is either equal to $sc(L)-1$ or to $sc(L)$. The *incomplete transition complexity*, $itc(L)$, of a regular language L is the minimal number of transitions over all DFAs that accepts L. We omit the term *incomplete* whenever the model is explicitly given. A τ-*transition* is a transition labeled by $\tau \in \Sigma$. The τ-*transition complexity* of L, $itc_\tau(L)$ is the minimal number of τ-transitions of any DFA recognizing L. It is known that $itc(L) = \sum_{\tau\in\Sigma} itc_\tau(L)$ [4, 7]. For determining the transition complexity of an operation, we also consider the following measures and refined numbers of transitions. Let $A = ([0, n-1], \Sigma, \delta, 0, F)$ be a DFA, $\tau \in \Sigma$, and $i \in [0, n-1]$. We define $f(A) = |F|$, $f(A, i) = |F \cap [0, i-1]|$, $t_\tau(A, i)$ as 1 if exist a τ-transition leaving i and 0 otherwise, and $\overline{t}_\tau(a, i)$ as its complement. Let $s_\tau(A) = t_\tau(A, 0)$, $e_\tau(A) = \sum_{i\in F} t_\tau(A, i)$, $t_\tau(A) = \sum_{i\in Q} t_\tau(A, i)$, $t_\tau(A, [k, l]) = \sum_{i\in[k,l]} t_\tau(A, i)$, and the respective complements $\overline{s}_\tau(A) = \overline{t}_\tau(A, 0)$, $\overline{e}_\tau(A) = \sum_{i\in F} \overline{t}_\tau(A, i)$, etc. We denote by $in_\tau(A, i)$ the number of transitions reaching i, $a_\tau(A) = \sum_{i\in F} in_\tau(A, i)$ and $c_\tau(A, i) = 0$ if $in_\tau(A, i) > 0$ and 1 otherwise. Whenever there is no ambiguity we omit A from the above definitions. All the above measures, can be defined for a regular language L, considering the measure values for its minimal DFA. We define $s(L) = \sum_{\tau\in\Sigma} s_\tau(L)$ and $a(L) = \sum_{\tau\in\Sigma} a_\tau(L)$. Let A be a minimal DFA accepting a finite language, where the states are assumed to be topologically ordered. Then, $s(\mathcal{L}(A)) = 0$ and there is exactly one final state, denoted π and called *pre-dead*, such that $\sum_{\tau\in\Sigma} t_\tau(\pi) = 0$. The *level* of a state i is the size of the shortest path from the initial state to i, and never exceeds $n-1$. The level of A is the level of π.

3 Union and Intersection

Given two incomplete DFAs $A = ([0, m-1], \Sigma, \delta_A, 0, F_A)$ and $B = ([0, n-1], \Sigma, \delta_B, 0, F_B)$ adaptations of the classical cartesian product construction can be used to obtain DFAs accepting $\mathcal{L}(A) \cup \mathcal{L}(B)$ and $\mathcal{L}(A) \cap \mathcal{L}(B)$ [7].

Theorem 1. *For any two finite languages L_1 and L_2 with $isc(L_1) = m$ and $isc(L_2) = n$, one has:*

1. $isc(L_1 \cup L_2) \le mn - 2$ and

$$itc(L_1 \cup L_2) \le \sum_{\tau\in\Sigma} (s_\tau(L_1) \boxplus s_\tau(L_2) - (itc_\tau(L_1) - s_\tau(L_1))(itc_\tau(L_2) - s_\tau(L_2))) \\ + n(itc(L_1) - s(L_1)) + m(itc(L_2) - s(L_2)),$$

where for x, y boolean values, $x \boxplus y = \min(x+y, 1)$.

2. *$isc(L_1 \cap L_2) \le mn - 2m - 2n + 6$ and*

$$itc(L_1 \cap L_2) \le \sum_{\tau\in\Sigma} (s_\tau(L_1)s_\tau(L_2) + (itc_\tau(L_1) - s_\tau(L_1) - a_\tau(L_1))(itc_\tau(L_2) - s_\tau(L_2) - a_\tau(L_2)) + a_\tau(L_1)a_\tau(L_2)) \,.$$

All the above upper bounds are tight but can only be reached with an alphabet of size depending on m and n.

4 Concatenation

Câmpeanu *et al.* [2] studied the state complexity of the concatenation of a m-state complete DFA A with a n-state complete DFA B over an alphabet of size k and proposed the upper bound

$$\sum_{i=0}^{m-2} \min\left\{k^i, \sum_{j=0}^{f(A,i)} \binom{n-2}{j}\right\} + \min\left\{k^{m-1}, \sum_{j=0}^{f(A)} \binom{n-2}{j}\right\}, \qquad (1)$$

which was proved to be tight for $m > n-1$. It is easy to see that the second term of (1) is $\sum_{j=0}^{f(A)} \binom{n-2}{j}$ if $m > n-1$, and k^{m-1}, otherwise. The value k^{m-1} indicates that the DFA resulting from the concatenation has states with level at most $m-1$. But that is not always the case, as we can see by the example[2] in Figure 2. This implies that (1) is not an upper bound if $m < n$. With these changes, we have

Theorem 2. *For any two finite languages L_1 and L_2 with $sc(L_1) = m$ and $sc(L_2) = n$ over an alphabet of size $k \ge 2$, one has*

$$sc(L_1L_2) \le \sum_{i=0}^{m-2} \min\left\{k^i, \sum_{j=0}^{f(L_1,i)} \binom{n-2}{j}\right\} + \sum_{j=0}^{f(L_1)} \binom{n-2}{j}. \qquad (2)$$

Given two incomplete DFAs $A = ([0, m-1], \Sigma, \delta_A, 0, F_A)$ and $B = ([0, n-1], \Sigma, \delta_B, 0, F_B)$, that represent finite languages, the algorithm by Maia *et al.* for the concatenation of regular languages can be applied to obtain a DFA $C = (R, \Sigma, \delta_C, r_0, F_C)$ accepting $\mathcal{L}(A)\mathcal{L}(B)$. The set of states of C is contained in the set $([0, m-1] \cup \{\Omega_A\}) \times 2^{[0,n-1]}$, the initial state r_0 is $(0, \emptyset)$ if $0 \notin F_A$, and is $(0, \{0\})$ otherwise; $F_C = \{(i, P) \in R \mid P \cap F_B \neq \emptyset\}$, and for $\tau \in \Sigma$, $i \in [0, m-1]$, and $P \subseteq [0, n-1]$, $\delta_C((i, P), \tau) = (i', P')$ with $i' = \delta_A(i, \tau)$, if $\delta_A(i, \tau) \downarrow$ or $i' = \Omega_A$ otherwise, and $P' = \delta_B(P, \tau) \cup \{0\}$ if $i' \in F_A$ and $P' = \delta_B(P, \tau)$ otherwise. For the incomplete state and transition complexity we have

[2] Note that we are omitting the dead state in the figures.

Theorem 3. *For any two finite languages L_1 and L_2 with $isc(L_1) = m$ and $isc(L_2) = n$ over an alphabet of size $k \geq 2$, and making $\Lambda_j = \binom{n-1}{j} - \binom{\overline{t}_\tau(L_2)-\overline{s}_\tau(L_2)}{j}$, $\Delta_j = \binom{n-1}{j} - \overline{s}_\tau(L_2)\binom{\overline{t}_\tau(L_2)-\overline{s}_\tau(L_2)}{j}$ one has*

$$isc(L_1L_2) \leq \sum_{i=0}^{m-1} \min\left\{k^i, \sum_{j=0}^{f(L_1,i)} \binom{n-1}{j}\right\} + \sum_{j=0}^{f(L_1)} \binom{n-1}{j} - 1. \qquad (3)$$

and

$$itc(L_1L_2) \leq k\sum_{i=0}^{m-2} \min\left\{k^i, \sum_{j=0}^{f(L_1,i)} \binom{n-1}{j}\right\} + \\ + \sum_{\tau\in\Sigma}\left(\min\left\{k^{m-1} - \overline{s}_\tau(L_2), \sum_{j=0}^{f(L_1)-1} \Delta_j\right\} + \sum_{j=0}^{f(L_1)} \Lambda_j\right). \qquad (4)$$

Proof. The τ-transitions of the DFA C accepting $\mathcal{L}(A)\mathcal{L}(B)$ have three forms: (i, β) where i represents the transition leaving the state $i \in [0, m-1]$; $(-1, \beta)$ where -1 represents the absence of the transition from state π_A to Ω_A; and $(-2, \beta)$ where -2 represents any transition leaving Ω_A. In all forms, β is a set of transitions of DFA B. The number of τ-transitions of the form (i, β) is at most $\sum_{i=0}^{m-2} \min\{k^i, \sum_{j=0}^{f(L_1,i)} \binom{n-1}{j}\}$ which corresponds to the number of states of the form (i, P), for $i \in [0, m-1]$ and $P \subseteq [0, n-1]$. The number of τ-transitions of the form $(-1, \beta)$ is $\min\{k^{m-1} - \overline{s}_\tau(L_2), \sum_{j=0}^{f(L_1)-1} \Delta_j\}$. We have at most k^{m-1} states in this level. However, if $s_\tau(B, 0) = 0$ we need to remove the transition $(-1, \emptyset)$ which leaves the state $(m-1, \{0\})$. On the other hand, the size of β is at most $f(L_1) - 1$ and we know that β has always the transition leaving the initial state by τ, if it exists. If this transition does not exist, *i.e.* $\overline{s}_\tau(B, 0) = 1$, we need to remove the sets with only non-defined transitions, because they originate transitions of the form $(-1, \emptyset)$. The number of τ-transitions of the form $(-2, \beta)$ is $\sum_{j=0}^{f(L_1)} \Lambda_j$ and this case is similar to the previous one.

To prove that the bounds are reachable, we consider two cases depending whether $m + 1 \geq n$ or not.

Case 1: $\boldsymbol{m+1 \geq n}$ The witness languages are the ones presented by Câmpeanu *et al.* (see Figure 1).

Fig. 1. DFA A with m states and DFA B with n states.

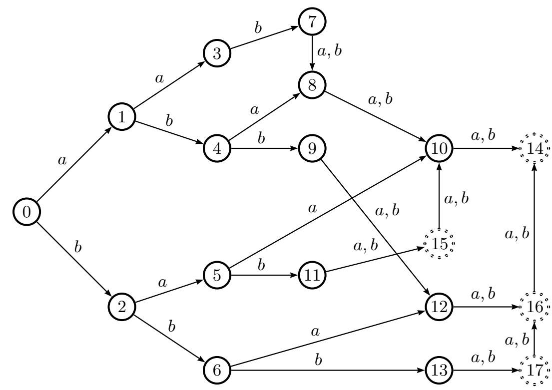

Fig. 2. DFA resulting of the concatenation of DFA A with $m = 3$ and DFA B with $n = 5$, of Fig. 1. The states with dashed lines have level > 3 and are not accounted for by formula (1).

Theorem 4. *For any two integers $m \geq 2$ and $n \geq 2$ such that $m+1 \geq n$, there exist an m-state DFA A and an n-state DFA B, both accepting finite languages, such that any DFA accepting $\mathcal{L}(A)\mathcal{L}(B)$ needs at least $(m-n+3)2^{n-1}-2$ states and $2^n(m-n+3)-8$ transitions.*

Case 2: $\boldsymbol{m+1 < n}$ Let $\Sigma = \{b\} \cup \{a_i \mid i \in [1, n-2]\}$. Let $A = ([0, m-1], \Sigma, \delta_A, 0, [0, m-1])$ where $\delta_A(i, \tau) = i+1$, for any $\tau \in \Sigma$. Let $B = ([0, n-1], \Sigma, \delta_B, 0, \{n-1\})$ where $\delta_B(i, b) = i+1$, for $i \in [0, n-2]$, $\delta_B(i, a_j) = i+j$, for $i, j \in [1, n-2]$, $i+j \in [2, n-1]$, and $\delta_B(0, a_j) = j$, for $j \in [2, n-2]$. Note that A and B are minimal DFAs.

Theorem 5. *For any two integers $m \geq 2$ and $n \geq 2$, with $m+1 < n$, there exist an m-state DFA A and an n-state DFA B, both accepting finite languages over an alphabet of size depending on m and n, such that the number of states and transitions of any DFA accepting $\mathcal{L}(A)\mathcal{L}(B)$ reaches the upper bounds.*

Proof. We need to show that the DFA C accepting $\mathcal{L}(A)\mathcal{L}(B)$ is minimal, *i.e.*, (i) every state of C is reachable from the initial state; (ii) each state of C defines a distinct equivalence class. To prove (i), we first show that all states $(i, P) \subseteq R$ with $i \in [1, m-1]$ are reachable. The following facts hold for the automaton C: 1) every state of the form $(i+1, P')$ is reached by a transition from a state (i, P) (by the construction of A) and $|P'| \leq |P|+1$, for $i \in [1, m-2]$; 2) every state of the form (Ω_A, P') is reached by a transition from a state $(m-1, P)$ (by the construction of A) and $|P'| \leq |P|+1$; 3) for each state (i, P), $P \subseteq [0, n-1]$, $|P| \leq i+1$ and $0 \in P$, $i \in [1, m-1]$; 4) for each state (Ω_A, P), $\emptyset \neq P \subseteq [0, n-1]$, $|P| \leq m$ and $0 \notin P$.

Suppose that for a $1 \leq i \leq m-2$, all states (i, P) are reachable. The number of states of the form $(1, P)$ is $m-1$ and of the form (i, P) with $i \in [2, m-2]$ is $\sum_{j=0}^{i} \binom{n-1}{j}$. Let us consider the states $(i+1, P')$. If $P' = \{0\}$, then

$\delta_C((i,\{0\}),a_1) = (i+1,P')$. Otherwise, let $l = \min(P'\setminus\{0\})$ and $S_l = \{s-l \mid s \in P'\setminus\{0\}\}$. Then, $\delta_C((i,S_l),a_l) = (i+1,P')$, if $2 \le l \le n-2$; $\delta_C((i,\{0\}\cup S_1),a_1) = (i+1,P')$, if $l = n-1$; and $\delta_C((i,S_1),b) = (i+1,P')$, if $l = 1$.. Thus, all $\sum_{j=0}^{i+1}\binom{n-1}{j}$ states of the form $(i+1,P')$ are reachable. Let us consider the states (Ω_A,P'). P' is always an non empty set by construction of C. Let $l = \min(P')$ and $S_l = \{s-l \mid s \in P'\}$. Thus, $\delta_C((m-1,S_l),a_l) = (\Omega_A,P')$, if $2 \le l \le n-2$; $\delta_C((m-1,\{0\}\cup S_1),a_1) = (\Omega_A,P')$, if $l = n-1$; and $\delta_C((m-1,S_1),b) = (\Omega_A,P')$, if $l = 1$ Thus, all $\sum_{j=0}^{m}\binom{n-1}{j} - 1$ states of the form (Ω_A,P') are reachable. To prove (ii), consider two distinct states $(i,P_1),(j,P_2) \in R$. If $i \neq j$, then $\delta_C((i,P_1),\ b^{n+m-2-i}) \in F_C$ but $\delta_C((j,P_2),b^{n+m-2-i}) \notin F_C$. If $i = j$, suppose that $P_1 \neq P_2$ and both are final or non-final. Let $P_1' = P_1 \setminus P_2$ and $P_2' = P_2 \setminus P_1$. Without loss of generality, let P_1' be the set which has the minimal value, let us say l. Thus $\delta_C((i,P_1),a_1^{n-1-l}) \in F_C$ but $\delta_C((i,P_2),a_1^{n-1-l}) \notin F_C$. The proof corresponding to the number of transitions is similar to the proof of Theorem 3.

Theorem 6. *The upper bounds for state and transition complexity of concatenation cannot be reached for any alphabet with a fixed size for $m \ge 0$, $n > m+1$.*

Proof. Let $S = \{(\Omega_A,P) \mid 1 \in P\} \subseteq R$. A state $(\Omega_A,P) \in S$ has to satisfy the following condition:

$$\exists i \in F_A \exists P' \subseteq 2^{[0,n-1]} \exists \tau \in \Sigma : \delta_C((i,P'\cup\{0\}),\tau) = (\Omega_A,P).$$

The maximal size of S is $\sum_{j=0}^{f(A)-1}\binom{n-2}{j}$, because by construction $1 \in P$ and $0 \notin P$. Assume that Σ has a fixed size $k = |\Sigma|$. Then, the maximal number of words that reach states of S from r_0 is $\sum_{i=0}^{f(A)} k^{i+1}$ since the words that reach a state $s \in S$ are of the form $w_A\sigma$, where $w_A \in L(A)$ and $\sigma \in \Sigma$. As $n > m$, for some $l \ge 0$ we have $n = m + l$. Thus for an l sufficiently large $\sum_{i=0}^{f(A)} k^{i+1} \ll \sum_{j=0}^{f(A)-1}\binom{m+l-2}{j}$, which is an absurd. The absurd resulted from supposing that k is fixed.

5 Star and Reversal

Given an incomplete DFA $A = ([0,m-1],\Sigma,\delta_A,0,F_A)$ accepting a finite language, we obtain a DFA accepting $\mathcal{L}(A)^\star$ using an algorithm similar to the one for regular languages [7] and a DFA that accepts $\mathcal{L}(A)^R$, reversing all transitions of A and then determinizing the resulting NFA. Note that if $f(A) = 1$ then the minimal DFA accepting $\mathcal{L}(A)^\star$ has also m states. Thus, for the Kleene star operation, we will consider DFAs with at least two final states.

Theorem 7. *For any finite language L with $isc(L) = m$ one has*

1. *if $f(L) \ge 2$, $isc(L^\star) \le 2^{m-f(L)-1} + 2^{m-2} - 1$ and*

$$itc(L^\star) \le 2^{m-f(L)-1}\left(k + \sum_{\tau\in\Sigma} 2^{e_\tau(L)}\right) - \sum_{\tau\in\Sigma} 2^{n_\tau} - \sum_{\tau\in X} 2^{n_\tau},$$

where $n_\tau = \overline{t}_\tau(L) - \overline{s}_\tau(L) - \overline{e}_\tau(L)$ and $X = \{\tau \in \Sigma \mid s_\tau(L) = 0\}$.

2. *if $m \geq 3$, $k \geq 2$, and l is the smallest integer such that $2^{m-l} \leq k^l$, $isc(L^R) \leq \sum_{i=0}^{l-1} k^i + 2^{m-l} - 1$ moreover if m is odd,*

$$itc(L^R) \leq \sum_{i=0}^{l} k^i - 1 + k2^{m-l} - \sum_{\tau \in \Sigma} 2^{\sum_{i=0}^{l-1} \overline{t}_\tau(L,i)+1},$$

and, if m is even,

$$itc(L^R) \leq \sum_{i=0}^{l} k^i - 1 + k2^{m-l} - \sum_{\tau \in \Sigma} \left(2^{\sum_{i=0}^{l-2} \overline{t}_\tau(L,i)+1} - c_\tau(L,l) \right).$$

6 Final Remarks

In this paper we studied the incomplete state and transition complexity of basic regularity preserving operations on finite languages. Note that for the complement operation these descriptional measures coincide with the ones on regular languages. Table 1 summarizes some of those results. For unary finite languages the incomplete transition complexity is equal to the incomplete state complexity of that language, which is always equal to the state complexity of the language minus one.

References

1. Beesley, K.R., Karttunen, L.: Finite State Morphology. CSLI Publications, Stanford University (2003)
2. Câmpeanu, C., II, K.C., Salomaa, K., Yu, S.: State complexity of basic operations on finite languages. In: Boldt, O., Jürgensen, H. (eds.) WIA 1999. LNCS, vol. 2214, pp. 60–70. Springer (2001)
3. Cassandras, C.G., Lafortune, S.: Introduction to discrete event systems. Springer (2006)
4. Gao, Y., Salomaa, K., Yu, S.: Transition complexity of incomplete DFAs. Fundam. Inform. 110(1-4), 143–158 (2011)
5. Han, Y.S., Salomaa, K.: State complexity of union and intersection of finite languages. Int. J. Found. Comput. Sci. 19(3), 581–595 (2008)
6. Hopcroft, J.E., Ullman, J.D.: Introduction to Automata Theory, Languages and Computation. Addison-Wesley (1979)
7. Maia, E., Moreira, N., Reis, R.: Incomplete transition complexity of some basic operations. In: van Emde et al., P. (ed.) SOFSEM 2013. LNCS, vol. 7741, pp. 319–331. Springer (2013)
8. Maurel, D., Guenthner, F.: Automata and Dictionaries. College Publications (2005)
9. Owens, S., Reppy, J.H., Turon, A.: Regular-expression derivatives re-examined. J. Funct. Program. 19(2), 173–190 (2009)
10. Salomaa, K., Yu, S.: NFA to DFA transformation for finite languages over arbitrary alphabets. J. of Aut., Lang. and Comb. 2(3), 177–186 (1997)
11. Shallit, J.: A Second Course in Formal Languages and Automata Theory. CUP (2008)
12. Yu, S.: Regular languages. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, vol. 1, pp. 41–110. Springer (1997)