Finite Orbits of Language Operations

Émilie Charlier[1], Mike Domaratzki[2], Tero Harju[3], and Jeffrey Shallit[1]

[1] University of Waterloo, Waterloo, ON N2L 3G1 Canada `echarlier@uwaterloo.ca`, `shallit@cs.uwaterloo.ca`

[2] University of Manitoba, Winnipeg, MB R3T 2N2 Canada `mdomarat@cs.umanitoba.ca`

[3] University of Turku, FIN-20014 Turku, Finland `harju@utu.fi`

Abstract. We consider a set of natural operations on languages, and prove that the orbit of any language L under the monoid generated by this set is finite and bounded, independently of L. This generalizes previous results about complement, Kleene closure, and positive closure.

1 Introduction

If t, x, y, z are (possibly empty) words with $t = xyz$, we say

- x is a *prefix* of t;
- z is a *suffix* of t; and
- y is a *factor* of t.

If $t = x_1 t_1 x_2 t_2 \cdots x_n t_n x_{n+1}$ for some $n \geq 1$ and some (possibly empty) words t_i, x_j, $1 \leq i \leq n$, $1 \leq j \leq n+1$, then $t_1 \cdots t_n$ is said to be a *subword* of t. Thus a factor is a contiguous block, while a subword can be "scattered".

Let L be a language over the finite alphabet Σ, that is, $L \subseteq \Sigma^*$. We consider the following eight natural operations applied to L:

$$\begin{aligned}
k &: \ L \to L^* \\
e &: \ L \to L^+ \\
c &: \ L \to \overline{L} = \Sigma^* - L \\
p &: \ L \to \mathrm{pref}(L) \\
s &: \ L \to \mathrm{suff}(L) \\
f &: \ L \to \mathrm{fact}(L) \\
w &: \ L \to \mathrm{subw}(L) \\
r &: \ L \to L^R
\end{aligned}$$

Here

$$\begin{aligned}
\mathrm{pref}(L) &= \{x \in \Sigma^* \ : \ x \text{ is a prefix of some } y \in L\}; \\
\mathrm{suff}(L) &= \{x \in \Sigma^* \ : \ x \text{ is a suffix of some } y \in L\}; \\
\mathrm{fact}(L) &= \{x \in \Sigma^* \ : \ x \text{ is a factor of some } y \in L\}; \\
\mathrm{subw}(L) &= \{x \in \Sigma^* \ : \ x \text{ is a subword of some } y \in L\}; \\
L^R &= \{x \in \Sigma^* \ : \ x^R \in L\};
\end{aligned}$$

where x^R denotes the reverse of the word x.

We compose these operations as follows: if $x = a_1 a_2 \cdots a_n \in \{k, e, c, p, s, f, w, r\}^*$, then

$$x(L) = a_1(a_2(a_3(\cdots(a_n(L))\cdots))).$$

Thus, for example, $ck(L) = \overline{L^*}$. We also write $\epsilon(L) = L$.

Given two elements $x, y \in \{k, e, c, p, s, f, w, r\}^*$, we write $x \equiv y$ if $x(L) = y(L)$ for all languages L, and we write $x \subseteq y$ if $x(L) \subseteq y(L)$ for all languages L.

Given a subset $S \subseteq \{k, e, c, p, s, f, w, r\}$, we can consider the orbit of languages

$$\mathcal{O}_S(L) = \{x(L) \ : \ x \in S^*\}$$

under the monoid of operations generated by S. We are interested in the following questions: when is this monoid finite? Is the cardinality of $\mathcal{O}_S(L)$ bounded, independently of L?

These questions were previously investigated for the sets $S = \{k, c\}$ and $S = \{e, c\}$ [4, 1], where the results can be viewed as the formal language analogues of Kuratowski's celebrated "14-theorem" for topological spaces [3, 2]. In this paper we consider the questions for other subsets of $\{k, e, c, p, s, f, w, r\}$. Our main result is Theorem 9 below, which shows finiteness for any subset of these eight operations.

2 Operations with infinite orbit

We point out that the orbit of L under an arbitrary operation need not be finite. For example, consider the operation q defined by

$$q(L) = \{x \in \Sigma^* \ : \ x \text{ there exists } y \in L \text{ such that } x \text{ is a proper prefix of } y \ \}.$$

Here by "x is a proper prefix of y", we mean that x is a prefix of y with $|x| < |y|$.

Let $L = \{a^n b^n \ : \ n \geq 1\}$. Then it is easy to see that the orbit

$$\mathcal{O}_{\{q\}}(L) = \{L, q(L), q^2(L), q^3(L), \ldots\}$$

is infinite, since the shortest word in $q^i(L) \ \cap \ a^+ b$ is $a^{i+1}b$.

The situation is somewhat different if L is regular:

Theorem 1. *Let q denote the proper prefix operation, and let L be a regular language accepted by a DFA of n states. Then $\mathcal{O}_{\{q\}}(L) \leq n$, and this bound is tight.*

Proof. Let $M = (Q, \Sigma, \delta, q_0, F)$ be an n-state DFA accepting L. Note that a DFA accepting $q(L)$ is given by $M' = (Q, \Sigma, \delta, q_0, F')$ where

$$F' = \{q \in Q \ : \ \text{there exists a path of length} \geq 1 \text{ from } q \text{ to a state of } F \ \}.$$

Reinterpreting this in terms of the underlying transition diagram, given a directed graph G on n vertices, and a distinguished set of vertices F, we are interested in the number of different sets obtained by iterating the operation that maps F to the set of all vertices that can reach a vertex in F by a path of length ≥ 1. We claim this is at most n. To see this, note that if a vertex v is part of any directed cycle, then once v is included, further iterations will retain it. Thus the number of distinct sets is as long as the longest directed path that is not a cycle, plus 1 for the inclusion of cycle vertices.

To see that the bound is tight, consider the language $L_n = \{\epsilon, a, a^2, \ldots, a^{n-2}\}$, which is accepted by a (complete) unary DFA of n states. Then $q(L_n) = L_{n-1}$, so this shows $|\mathcal{O}_{\{q\}}(L_n)| = n$. $\square$

It is possible for the orbit under a single operation to be infinite even if the operation is (in the terminology of the next section) expanding and inclusion-preserving. As an example, consider the operation of fractional exponentiation, defined by

$$\begin{aligned} n(L) &= \{x^\alpha \ : \alpha \geq 1 \text{ rational } \} \\ &= \bigcup_{x \in L} x^+ p(\{x\}). \end{aligned}$$

Proposition 1. *Let $L = \{ab\}$. Then the orbit $\mathcal{O}_{\{n\}}(L)$ is infinite.*

Proof. We have $aba^i \in n^i(\{ab\})$, but $aba^i \notin n^j(\{ab\})$ for $j < i$. $\square$

3 Kuratowski identities

Let $a : 2^{\Sigma^*} \to 2^{\Sigma^*}$ be an operation on languages. Suppose a satisfies the following three properties:

1. L is a subset of $a(L)$ (expanding);
2. If $L \subseteq M$ then $a(L) \subseteq a(M)$ (inclusion-preserving);
3. $a(a(L)) = a(L)$ (idempotent).

Then we say a is a *closure operation*. Examples of closure operations include k, e, p, s, f, and w.

Note that if a, b are closure operations, then their composition ab trivially satisfies properties 1 and 2 above, but may not satisfy property 3. For example, pk is not idempotent, as can be seen by examining its action on $L = \{ab\}$ ($aab \notin pk(L)$, but $aab \in pkpk(L)$).

Lemma 1. *Let $a \in \{k, e\}$ and $b \in \{p, s, f, w\}$. Then $aba \equiv bab \equiv ab$.*

Proof. We prove the result only for $b = p$; the other results are similar.

Since $L \subseteq a(L)$, we get $p(L) \subseteq pa(L)$, and then $ap(L) \subseteq apa(L)$. It remains to see $apa(L) \subseteq ap(L)$.

Any element of $a(L)$ is either ϵ or of the form $t = t_1 t_2 \cdots t_n$ for some $n \geq 1$, where each $t_i \in L$. Then any prefix of t looks like $t_1 t_2 \cdots t_{i-1} p_i$ for some $i \geq 1$, where p_i is a prefix of t_i, and hence in $p(L)$. But each t_i is also in $p(L)$, so this shows

$$pa(L) \subseteq ap(L). \tag{1}$$

Since a is a closure operation, $apa(L) \subseteq aap(L) = ap(L)$.

Similarly, we have $ap(L) \subseteq pap(L)$. Substituting $p(L)$ for L in (1) gives $pap(L) \subseteq app(L) = ap(L)$. □

Lemma 2. *The operations $kp, ks, kf, kw, ep, es, ef$ and ew are closure operations.*

Proof. We prove the result for kp, with the other results being similar. It suffices to prove property 3. From Lemma 1 we have $pkp(L) = kp(L)$. Applying k to both sides, and using the idempotence of k, we get $kpkp(L) = kkp(L) = kp(L)$. □

If a is a closure operation, and c denotes complement, then it is well-known (and shown, for example, in [4]) that $acacaca \equiv aca$. However, we will need the following more general observation, which seems to be new:

Theorem 2. *Let x, y be closure operations. Then $xcycxcy \equiv xcy$.*

Proof. $xcycxcy \subseteq xcy$: We have $L \subseteq y(L)$ by the expanding property. Then $cy(L) \subseteq c(L)$. By the inclusion-preserving property we have $xcy(L) \subseteq xc(L)$. Since this identity holds for all L, it holds in particular for $cxcy(L)$. Substituting, we get $xcycxcy(L) \subseteq xccxcy(L)$. But $xccxcy(L) = xcy(L)$ by the idempotence of x.

$xcy \subseteq xcycxcy$: We have $L \subseteq x(L)$ by the expanding property. Then, replacing L by $cy(L)$, we get $cy \subseteq xcy$. Applying c to both sides, we get $cxcy \subseteq ccy = y$. Applying y to both sides, and using the inclusion-preserving property and idempotence, we get $ycxcy \subseteq yy = y$. Applying c to both sides, we get $cy \subseteq cycxcy$. Finally, applying x to both sides and using the inclusion-preserving property, we get $xcy \subseteq xcycxcy$. □

Remark 1. Theorem 2 would also hold if c were replaced by any inclusion-reversing operation satisfying $cc \equiv \epsilon$.

As a corollary, we get [4, 1]:

Corollary 1. *If $S = \{a, c\}$, where a is any closure operation, and L is any language, the orbit $\mathcal{O}_S(L)$ contains at most 14 distinct languages.*

Proof. The 14 languages are given by the image of L under the 14 operations

$$\epsilon, a, c, ac, ca, aca, cac, acac, caca, acaca, cacac, acacac, cacaca, cacacac.$$

□

Remark 2. Theorem 2, together with Lemma 2, thus gives 196 separate identities.

In a similar fashion, we can obtain many kinds of Kuratowski-style identities involving k, e, c, p, s, f, w and r.

Theorem 3. *Let $a \in \{k, e\}$ and $b \in \{p, s, f, w\}$. Then we have the following identities:*

4. $abcacaca \equiv abca$
5. $bcbcbcab \equiv bcab$
6. $abcbcabcab \equiv abcab$

Proof. We only prove the first; the rest are similar. From Theorem 2 we get $acacaca \equiv aca$. Hence $ab(acacaca) \equiv ab(aca)$, or equivalently, $aba(cacaca) \equiv aba(ca)$. Since $aba \equiv ab$ from Lemma 1, we get $abcacaca \equiv abca$. □

4 Additional identities

In this section we prove some additional identities connecting the operations $\{k, e, c, p, s, f, w, r\}$.

Theorem 4. *We have*

7. $rp \equiv sr$
8. $rs \equiv pr$
9. $rf \equiv fr$
10. $rc \equiv cr$
11. $rk \equiv kr$
12. $rw \equiv wr$
13. $ps \equiv sp \equiv f$
14. $pf \equiv fp \equiv f$
15. $sf \equiv fs \equiv f$
16. $pw \equiv wp \equiv sw \equiv ws \equiv fw \equiv wf \equiv w$
17. $kw \equiv wk$
18. $rkw \equiv kw$
19. $ek \equiv ke \equiv k$
20. $fks \equiv pks$
21. $fkp \equiv skp$
22. $rkf \equiv skf \equiv pkf \equiv kf$

Proof. All of these are relatively straightforward. To see (20), note that $p(L) \subseteq f(L)$ for all L, and hence $pks(L) \subseteq fks(L)$. Hence it suffices to show the reverse inclusion.

Note that every element of $ks(L)$ is either ϵ or can be written $x = s_1 s_2 \cdots s_n$ for some $n \geq 1$, where each $s_i \in s(L)$. In the latter case, any factor of x must be of the form $y = s_i'' s_{i+1} \cdots s_{j-1} s_j'$, where s_i'' is a suffix of s_i and s_j' is a prefix of s_j. Then $s_i'' s_{i+1} \cdots s_{j-1} s_j \in ks(L)$ and hence $y \in pks(L)$.

Similarly, we have $pkf \equiv pk(ps) \equiv (pkp)s \equiv (kp)s \equiv k(ps) = kf$, which proves part of (22).

Theorem 5. *We have*

23. $pcs(L) = \Sigma^*$ *or* $\emptyset$.
24. *The same result holds for* $pcf, fcs, fcf, scp, scf, fcp, wcp, wcs, wcf, pcw, scw, fcw, wcw$.

Proof. Let us prove the first statement. Either $s(L) = \Sigma^*$, or $s(L)$ omits some word v. In the former case, $cs(L) = \emptyset$, and so $pcs(L) = \emptyset$. In the latter case, we have $s(L)$ omits v, so $s(L)$ must also omit $\Sigma^* v$ (for otherwise, if $xv \in f(L)$ for some x, then $v \in s(L)$). So $\Sigma^* v \subseteq cs(L)$. Hence $pcs(L) = \Sigma^*$.

The remaining statements are proved similarly. □

Lemma 3. *Let* L *be any language.*

(a) *If* $xy \in kp(L)$ *then* $x \in kp(L)$ *and* $y \in kf(L)$.
(b) *If* $xy \in ks(L)$ *then* $x \in kf(L)$ *and* $y \in ks(L)$.
(c) *If* $xy \in kf(L)$ *then* $x, y \in kf(L)$.
(d) *If* $xy \in kw(L)$, *then* $x, y \in kw(L)$.

Proof. We prove only (b), with the others being proved similarly. If $xy \in ks(L)$, then $x \in pks(L)$ and $y \in sks(L)$. But $s \subseteq f$, so $pks \subseteq pkf$, and $pkf = kf$ by (22). Hence $x \in kf(L)$. Similarly, $sks \equiv ks$ by Lemma 1, so $\in ks(L)$. □

Theorem 6. *Let* $b \in \{p, s, f, w\}$. *Then*

25. $kcb(L) = cb(L) \cup \{\epsilon\}$
26. $kckb(L) = ckb(L) \cup \{\epsilon\}$

Proof. We prove only two of these identities; the others can be proved similarly.

$kcp(L) = cp(L) \cup \{\epsilon\}$: If $cp(L) = \emptyset$ then the result is clear. Now assume $x \in kcp(L)$. Either $x = \epsilon$ or we can write $x = x_1 x_2 \cdots x_n$ for some $n \geq 1$, where each $x_i \in cp(L)$. Then each $x_i \notin p(L)$. In particular $x_1 \notin p(L)$. Then $x_1 x_2 \cdots x_n \notin p(L)$, because if it were, then $x_1 \in p(L)$, a contradiction. Hence $x \in cp(L)$.

$kckp(L) = ckp(L) \cup \{\epsilon\}$: If $ckp(L) = \emptyset$ then the result is clear. Now assume $x \in kckp(L)$. Either $x = \epsilon$ or we can write $x = x_1 x_2 \cdots x_n$ for some $n \geq 1$, where each $x_i \in ckp(L)$. Then each $x_i \notin kp(L)$. In particular $x_1 \notin kp(L)$. Hence $x_1(x_2 \cdots x_n) \notin kp(L)$, because if it were, then $x_1 \in kp(L)$ by Lemma 3, a contradiction. Hence $x \notin kp(L)$, so $x \in ckp(L)$, as desired. □

Theorem 7. *We have*

27. $sckp(L) = \Sigma^*$ *or* $\emptyset$.
28. *The same result holds for* $fckp, pcks, fcks, pckf, sckf, fckf, wckp, wcks, wckf, wckw, pckw, sckw, fckw$.

Proof. To prove (27), note that either $kp(L) = \Sigma^*$, or $kp(L)$ omits some word v. In the former case, $ckp(L) = \emptyset$, and so $sckp(L) = \emptyset$. In the latter case, we have $kp(L)$ omits v, so $kp(L)$ must also omit $v\Sigma^*$ (for otherwise, if $vx \in kp(L)$ for some x, then $v \in kp(L)$ by Lemma 3, a contradiction). Then $v\Sigma^* \in ckp(L)$ and hence $sckp(L) = \Sigma^*$.

The other results can be proved similarly. □

Lemma 4. *Let L be any language.*

(a) If $xy \in skp(L)$, then $x, y \in skp(L)$.
(b) If $xy \in pks(L)$, then $x, y \in pks(L)$.

Proof. We prove only (a), with (b) being proved similarly.

If $xy \in skp(L)$, then $x \in pskp(L)$ and $y \in sskp(L)$.

But $pskp \equiv (ps)kp \equiv fkp \equiv skp$ by (21). So $x \in skp(L)$. Also, $sskp = skp$, so $y \in skp(L)$. □

Theorem 8. *We have*

29. $scskp(L) = \Sigma^*$ or $\emptyset$.
30. The same result holds for pcpks.

Proof. We prove only the first result; the second can be proved analogously. Either $skp(L) = \Sigma^*$, or it omits some word v. In the first case we have $cskp(L) = \emptyset$ and hence $scskp(L) = \emptyset$. In the second case, $skp(L)$ must omit $v\Sigma^*$ (for if $vx \in skp(L)$ for any x, then by Lemma 4 we have $v \in skp(L)$, a contradiction). Hence $scskp(L) = \Sigma^*$. □

5 Results

Our main result is the following:

Theorem 9. *Let $S = \{k, e, c, p, f, s, w, r\}$. Then for every language L, the set $\mathcal{O}_S(L)$ contains at most 7652 distinct languages.*

Proof. Our proof was carried out mechanically. We used breadth-first search to examine the set $S^* = \{k, e, c, p, f, s, w, r\}^*$ by increasing length of the words; within each length we used lexicographic order with $k < e < c < p < f < s < w < r$. The nodes remaining to be examined are stored in a queue Q.

As each new word x from the top of Q is examined, we test to see if any factor is of the form given in identities (23)–(24) or (28)–(30). If it is, then the corresponding language must be either Σ^*, $\emptyset$, $\{\epsilon\}$, or Σ^+; furthermore, each descendant language will be of this form. In this case the word x is discarded. One sticky point is that some of the identities, such as (25), are not strict identities, but involve ϵ. However, L and $L \cup \{\epsilon\}$ behave identically, up to inclusion or removal of $\{\epsilon\}$, under each of the operations k, c, p, s, f, r, e. Therefore each language L in our list actually represents $L \pm \{\epsilon\}$, which means that the total number of distinct languages may be as much as twice as what we conclude.[4] Since $k(L) = e(L) \cup \{\epsilon\}$, we do not have to examine any words containing e.

Otherwise, we use the remaining identities above to try to reduce x to an equivalent word that (a) is shorter or (b) is of the same length but lexicographically smaller. If we succeed, then we have already examined an equivalent word,

[4] This is a rather crude upper bound, which could be substantially improved.

so x is discarded. Otherwise $x(L)$ is potentially a new language, so we concatenate all elements of S to it and append them to the end of the queue.

If the process terminates, then $\mathcal{O}_S(L)$ is of finite cardinality.

We wrote our program in APL. For $S = \{k, e, c, p, f, s, w, r\}$, the process terminated with 3824 nodes that could not be simplified using our identities. As mentioned above, this does not include the ambiguity by either including or not including ϵ, so this number needs to be multiplied by two, giving 7648. Finally, we also did not count the four possibilities $\emptyset, \{\epsilon\}, \Sigma^+$, and Σ^*. The total is 7652.

The longest word examined was *cpcpckpckpckpcpcpckckckcr*, of length 25, and the same word with p replaced by s. □

Our program generates a complete description of the words and how they simplify, which can be viewed at `http://www.cs.uwaterloo.ca/~shallit/papers.html`.

Remark 3. If we use *two* arbitrary closure operations a and b with no relation between them, then the monoid generated by $\{a, b\}$ could potentially be infinite, since any two finite prefixes of $ababab\cdots$ are distinct.

Here is an example. Let p denote prefix, as above, and define the exponentiation operation

$$t(L) = \{x^i \ : \ x \in L \text{ and } i \text{ is an integer} \geq 1\}. \tag{2}$$

Then it is easy to see that t is a closure operation, and hence the orbits $\mathcal{O}_{\{p\}}(L)$ and $\mathcal{O}_{\{t\}}(L)$ are finite, for all L. However, for $L = \{ab\}$, the orbit $\mathcal{O}_{\{p,t\}}(L)$ is infinite, as $aba^i \in (pt)^i(L)$, but $aba^i \notin (pt)^j(L)$ for all $j < i$.

Thus our proof of Theorem 9 crucially depends on the properties of the operations $\{k, e, c, p, s, f, w, r\}$.

We now give some results for some interesting subsets of S.

5.1 Prefix and complement

In this case at most 14 distinct languages can be generated. The bound of 14 can be achieved, e.g., by the regular language over $\Sigma = \{a, b, c, d\}$ given by the regular expression

$$a^*((b + c)(a(\Sigma\Sigma)^* + b + d\Sigma^*) + d\Sigma^+)$$

and accepted by the DFA in Figure 1.

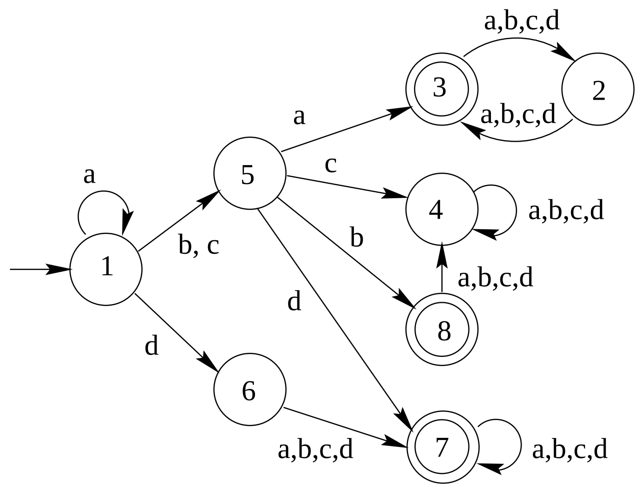

Fig. 1. DFA accepting a language L with orbit size 14 under operations p and c

Table 1 gives the appropriate set of final states under the operations.

language	final states
L	3,7,8
$c(L)$	1,2,4,5,6
$p(L)$	1,2,3,5,6,7,8
$pc(L)$	1,2,3,4,5,6,8
$cp(L)$	4
$cpc(L)$	7
$pcp(L)$	1,4,5,8
$pcpc(L)$	1,5,6,7
$cpcp(L)$	2,3,6,7
$cpcpc(L)$	2,3,4,8
$pcpcp(L)$	1,2,3,5,6,7
$pcpcpc(L)$	1,2,3,4,5,8
$cpcpcp(L)$	4, 8
$cpcpcpc(L)$	6, 7

Table 1. Final states for composed operations

5.2 Prefix, Kleene star, complement

The same process, described above for the operations $\{k, e, c, p, s, f, w, r\}$, can be carried out for other subsets, such as $\{k, c, p\}$. For this our breadth-first search gives 722 languages, but again, because we used the identity $kcp(L) =$

$cp(L) \cup \{\epsilon\}$, each language could potentially either include or not include ϵ. So we need to multiply this figure by 2, as before, giving an upper bound of at most 1444 distinct languages. The longest word examined was *cpcpckpckpckpcpcpckckckc*.

5.3 Complement, Kleene star, factor

Similarly, we can examine $\{k, c, f\}$. Here breadth-first search gives 54 languages, so our bound is $2 \cdot 54 + 4 = 112$. The longest word examined was *cfckckckc*.

5.4 Kleene star, prefix, suffix, factor

Here there are at most 13 distinct languages, given by the action of

$$\{\epsilon, k, p, s, f, kp, ks, kf, pk, sk, fk, pks, skp\}.$$

The bound of 13 is achieved, for example, by $L = \{abc\}$.

5.5 Summary of results

Table 2 gives our upper bounds on the number of distinct languages generated by the set of operations. An entry in **bold** indicates that the bound is known to be tight. Some entries, such as p, r, are omitted, since they are the same as others (in this case, p, s, f, r). Most bounds were obtained directly from our program, and others by additional reasoning. An asterisk denotes those bounds for which some additional reasoning was required to reduce the upper bound found by our program to the bound shown in Table 2.

6 Further work

We plan to continue to refine our estimates in Table 2, and pursue the status of other sets of operations. For example, if t is the exponentiation operation defined in (2), then, using the identities $kt = tk = k$, and the inclusion $t \subseteq k$, we get the additional Kuratowski-style identities

- $kctckck \equiv kck$
- $kckctck \equiv kck$
- $kctctck \equiv kck$
- $tctctck \equiv tck$
- $kctctct \equiv kct$.

This allows us to prove that the orbit $\mathcal{O}_{\{k,c,t\}}(L)$ is finite and of cardinality at most 126.

7 Acknowledgments

We thank John Brzozowski for his comments.

r	**2**	w	**2**	f	**2**
s	**2**	p	**2**	c	**2**
k	**2**	w, r	**4**	f, r	**4**
f, w	**3**	s, w	**3**	s, f	**3**
p, w	**3**	p, f	**3**	c, r	**4**
c, w	**6**$*$	c, f	**6**$*$	c, s	**14**
c, p	**14**	k, r	**4**	k, w	**4**
k, f	**5**	k, s	**5**	k, p	**5**
k, c	**14**	f, w, r	**6**	s, f, w	**4**
p, f, w	**4**	p, s, f	**4**	c, w, r	**10**$*$
c, f, r	**10**$*$	c, f, w	**8**$*$	c, s, w	20
c, s, f	**16**$*$	c, p, w	20	c, p, f	**16**$*$
k, w, r	**7**	k, f, r	**9**	k, f, w	**6**
k, s, w	**7**	k, s, f	**9**	k, p, w	**7**
k, p, f	**9**	k, c, r	28	k, c, w	88
k, c, f	74$*$	k, c, s	1444	k, c, p	1444
p, s, f, r	**8**	p, s, f, w	**5**	c, f, w, r	22
c, s, f, w	24	c, p, f, w	24	c, p, s, f	32
k, f, w, r	**11**	k, s, f, w	**10**	k, p, f, w	**10**
k, p, s, f	**13**	k, c, w, r	148	k, c, f, r	196
k, c, f, w	144	k, c, s, w	1504	k, c, s, f	1960
k, c, p, w	1504	k, c, p, f	1960	p, s, f, w, r	**10**
c, p, s, f, r	62	c, p, s, f, w	36	k, p, s, f, r	**25**
k, p, s, f, w	**14**	k, c, f, w, r	260	k, c, s, f, w	1992
k, c, p, f, w	1992	k, c, p, s, f	3808	c, p, s, f, w, r	70
k, p, s, f, w, r	**27**	k, c, p, s, f, r	7588	k, c, p, s, f, w	3840
k, c, p, s, f, w, r	7652				

Table 2. Upper bounds on the size of the orbit

References

1. J. A. Brzozowski, E. Grant, and J. Shallit. Closures in formal languages and Kuratowski's theorem. In V. Diekert and D. Nowotka, eds., *Developments in Language Theory (DLT), 13th International Conference*, Lecture Notes in Computer Science, Vol. 5583, Springer, 2009, pp. 125–144.
2. B. J. Gardner and M. Jackson. The Kuratowski closure-complement theorem. *New Zealand J. Math.* **38** (2008), 9–44. Available electronically at `http://nzjm.math.auckland.ac.nz/index.php/Volume_38_2008`.
3. C. Kuratowski. Sur l'opération $\overline{A}$ de l'analysis situs. *Fund. Math.* **3** (1922), 182–199.
4. D. Peleg. A generalized closure and complement phenomenon. *Discrete Math.* **50** (1984), 285–293.