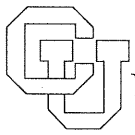
Pumping Lemmas for Regular Sets

A. Ehrenfeucht, R. Parikh, & G. Rozenberg

CU-CS-155-79 August 1979



University of Colorado at Boulder
DEPARTMENT OF COMPUTER SCIENCE

ANY OPINIONS, FINDINGS, AND CONCLUSIONS OR RECOMMENDATIONS EXPRESSED IN THIS PUBLICATION ARE THOSE OF THE AUTHOR(S) AND DO NOT NECESSARILY REFLECT THE VIEWS OF THE AGENCIES NAMED IN THE ACKNOWLEDGMENTS SECTION.

Introduction: Work on regular sets, sets recognizable by finite automata, goes back to the middle and late fifties, to the work of Kleene, Rabin, Scott and others. Most preliminary questions were solved at this early stage and most of the questions that remain, appear to be quite hard.

Among the properties that regular sets have are so called pumping lemmas or iteration theorems. These theorems follow from the fact that a finite automaton that accepts long strings must repeat internal states, i.e. loop. The existence of such a loop implies that the corresponding portion of the input string may be eliminated or iterated without affecting acceptance or rejection by the automaton.

The question that we intend to consider in this paper is that of a converse, i.e. the question whether a given pumping property implies regularity. We present both positive and negative results[1] and compare them with recent results by Jaffe [J] and Beauquier [B].

We close the paper with some open questions and suggestions for further work.

We are indebted to Rao Kosaraju for raising this general question of pumping lemmas. We are also indebted to Jaffe, Pratt, and Meyer for useful discussions.

Notational Remarks: Throughout the paper $\sum$ will be some fixed unspecified, finite alphabet. However, in theorem 1, $\sum$ is explicitly given. a, b, σ are symbols, x, y, z are strings. $|y|$ is the length of the string y. If $L \subseteq \sum^*$ then $\bar{L} = \sum^* - L$. Letters i,j,k,l,m,n,p denote natural numbers ≥ 0.

2. Negative Results: We begin this section by stating the pumping lemma.

Pumping lemma: Let $L \subseteq \Sigma^*$ be regular. Then L satisfies the pumping condition. I.e. there exists a $k > 0$ such that for all $x,y,z \in \Sigma^*$ if $|y| \geq k$ then there are $u,v,w \in \Sigma^*$, $v \neq \wedge$ such that $uvw = y$ and for all $i \geq 0$

$$xu(v)^i wz \in L \quad \text{iff} \quad xyz \in L \ .$$

This is a well known result, see ([H]). It can be proved by letting $k \geq$ the number of states of $\mathcal{A}$ where $\mathcal{A}$ is an automaton that recognizes L.

Question (Rao Kosaraju) Does the pumping condition imply regularity?

Theorem[2]: (Beauquier) There is a context free language L which satisfies the pumping condition but is not regular.

We prove below a somewhat stronger theorem.

Theorem 1: There are $2^{\aleph_0}$ languages which satisfy the pumping condition. Thus the pumping condition does not even imply recursiveness. Some of these languages are CF but not regular.

Proof: We prove this result by the following device. Let $\Sigma_1 = \{a,b\}$ and $X \subseteq \Sigma_1^*$. We take a 16 letter alphabet Σ and code X as a subset L(X) of Σ^*. L(X) satisfies the pumping condition and the map $X \to L(X)$ is 1-1. Since X is an arbitrary subset of Σ_1^*, there are $2^{\aleph_0}$ possibilities for X and hence the same number for L(X). This proves the first part of the lemma. We show moreover that if X is the language $\{a^n b^n \mid n \geq 0\}$ then L(X) is CF but not regular.

Details of proof: $\Sigma = \{a_{i,j} \mid 0 \leq i,j \leq 3\}$. We define two maps f_a, f_b from Σ to Σ.

$$f_a(a_{i,j}) = a_{i+1,j} \pmod 4,$$

$$f_b(a_{i,j}) = a_{i,j+1} \pmod 4.$$

The functions f_a, f_b are permutations of Σ and have moreover the property that applying two functions can never have the same effect as applying one. E.g. for all $\sigma \in \Sigma$,

$$f_b(f_a(\sigma)) \neq f_a(\sigma) \neq f_a(f_a(\sigma))$$

This is because applying two functions increments both subscripts i,j by one (mod 4) or one subscript by two (mod 4) and a single application of a function can never achieve this.

We define a legal string as any string $(\sigma_1)^{n_1}(\sigma_2)^{n_2} \ldots (\sigma_m)^{n_m} = x$ where σ_1 is $a_{o,o}$ and for all $l < m$, σ_{l+1} is either $f_a(\sigma_l)$ or $f_b(\sigma_l)$. If we think of the transition from σ_l to σ_{l+1} as being caused by an a or b, then there are m-1 transitions above and they correspond to a string y in Σ_1^*. We shall say that x codes y. (The powers n_i are all positive so y is unambiguously determined by x). Thus the string $x = a_{o,o}a_{1,o}a_{1,o}a_{1,1}$ is legal, $n_1 = 1$, $n_2 = 2$, and $n_3 = 1$. The coded string y is ab. Note that the same y has many codes x.

The parity of a string is the sum of all subscripts i,j (mod 2). Thus the parity of the string x above is 0.

Now we let

$$L(X) = \{x \mid x \text{ is legal and } x \text{ codes a } y \text{ such that } y \in X\} \cup \{x \mid x \text{ is illegal and parity of } x = 0\}$$

Clearly the map L is 1-1, for suppose $X \neq X^1$ and say $x \in X - X^1$. Then let y code x and $y \in L(X) - L(X^1)$. We shall now show that L(X) always satisfies the pumping condition. Let $k = 6$.

Let $zyz' \in \Sigma^*$ and $|y| \geq 6$. We consider cases :

(1) (a) zyz' is legal and y contains a doublet $\sigma\sigma$. Let $y = u\sigma w$ where the last symbol of u is also σ and let $v = \sigma$. Then for all i,

$zu(\sigma)^i wz'$ is legal and codes the <u>same</u> x that zyz' does. Hence $zu(\sigma)^i wz' \in L(X)$ iff $zyz' \in L(X)$.

(b) zyz' is legal but y contains no doublet. We now have to consider parities. Say for example that $zyz' \in L(X)$ and has odd parity. Now y itself must contain a symbol of <u>odd</u> parity. Let σ be that symbol and $y = uvw$. We can choose v so that it is not an end symbol of y. Then for all $i \geq 1$,

$zu(v)^i wz'$ codes the <u>same</u> string as zyz' and is legal so $zu(v)^i wz' \in L(X)$.

For $i = 0$, $zu(v)^i wz' = zuwz'$ has zero parity and is <u>illegal</u> so again $zu(v)^i wz' \in L(X)$.

The cases where zyz' has even parity and/or $zyz' \notin L(X)$ are similar.

(2) zyz' is illegal. The illegality may be caused by the initial symbol being other than $a_{o,o}$ or by a bad transition. In any case zyz' contains a subpiece y' of length ≤ 2 such that preserving that piece will preserve illegality. Hence since $|y| \geq 6$, we can find a v' such that

(a) v' is <u>disjoint</u> from y'. (This would be automatic if y' were in z or z' and can also be achieved if y' overlaps y)

(b) $|v'| = 2$.

Now let v be a subpiece of v' of parity 0. There must be a nontrivial such subpiece with one or two symbols. Let $y = uvw$. Then for all $i \geq 0$,

$zu(v)^i wz'$ has the same parity as zyz' and is illegal. Hence $zu(v)^i wz'$ is in $L(X)$ iff zyz' is.

This proves the first part of the lemma. To show the second part, consider first the strings $x = a^n b^n$ where n is divisible by 4 and the strings y which represent them. Consider the CF rules

$$S \rightarrow A_{o,o}\, A_{1,o}\, A_{2,o}\, A_{3,o}\, S\, A_{o,1}\, A_{o,2}\, A_{o;3}\, A_{oo}$$

$$S \rightarrow \wedge$$

$$A_{i,j} \rightarrow a_{i,j}\, A_{i,j}$$

$$A_{i,j} \rightarrow \wedge$$

The set generated is CF and is the set of y which represent some $a^n b^n$ where $n \equiv o \bmod 4$. The cases $n \equiv i \bmod 4$ for $i = 1, 2, 3$ are similar. Hence $L(X) \cap$ Legal is CF, being a union of four CF sets. But then

$L(X) = (L(X) \cap \text{Legal}) \cup (\text{not Legal} \cap \text{0-parity})$, and not Legal, 0-parity are regular. Thus $L(X)$ is CF, being a union of two CF sets.

But the set cannot be regular, for consider strings y_i such that y_i represents a^i and z_i such that z_i represents b^i and i is divisible by 4. (This is for convenience since $a_{o,o}$ is the starting symbol of all legal strings, and is also the last symbol of y_i if 4 divides i). Now for all i, j, if $i \neq j$ then

$y_i z_i \in L(X)$ and $y_j z_i \notin L(X)$. Hence, by Myhill's theorem [RS] or Nerode's theorem [N], $L(X)$ is not regular. Q.E.D.

3. Positive results: We saw in the last section that the pumping lemma does not imply regularity. This is also true (cf. footnote 2) of a somewhat stronger "marked pumping lemma". Thus the question arises

whether there is any form of pumping that is a necessary and sufficient condition for regularity. We begin the discussion by defining the notion of a pump and quoting a recent result of Jaffe.

Def: Let $L \subseteq \Sigma^*$, $x \in \Sigma^*$ and $x = uvw$, then v is a pump for x relative to L iff for all $i \geq 0$,

$$u(v)^i w \in L \text{ iff } x \in L.$$

Note that being a pump is really a property of the particular occurrence of v. x may have two occurrences of v of which one is a pump, the other not.

Theorem (Jaffe): L is regular if there is a k such that for all $x \in \Sigma^*$, if $|x| \geq k$ then $\forall u,v,w$, $x = uvw$, $v \neq \Lambda$ and for all z, v is a pump for xz relative to L. I.e. for all $i \geq 0$, all $z \in \Sigma^*$,

$$u(v)^i wz \in L \text{ iff } xz \in L .$$

Jaffe himself gives a direct proof of his result though it can also be shown from Nerode's theorem and the following three observations.

(1) It is sufficient to show that if $\equiv$ is the Nerode equivalence relation, then for every x there is an x' such that $|x'| < k$ and $x \equiv x'$. For then the index of $\equiv$ must be finite. There are only finitely many x' with $|x'| < k$.

(2) It is sufficient to show for (1) above that if $|x| \geq k$ then there is an x'' such that $|x''| < |x|$ and $x \equiv x''$, for repeating this will eventually yield the desired x' as above.

(3) However, if u,v,w,x are as in Jaffe's theorem then $x \equiv uw$ and $|uw| < |x|$.

(The "only if" part of the theorem is also true but quite easy)

However, Jaffe's pumping condition is not local. Given an x we don't need a pump just for x but a uniform pump for all xz, $z \in \Sigma^*$. So the question that arises is whether we can find a local pumping condition that is equivalent to regularity. Our theorem 2 below gives a positive solution to this question.

Def: $L \subseteq \Sigma^*$ has the block pumping property if there is a k such that for all $x, w, y_1, -, y_k, w'$ in Σ^*, if $x = wy_1 .. y_k w'$ then there exist l, j, $1 \leq l \leq j \leq k$ such that $y_l y_{l+1} \ldots y_j$ is a pump for x relative to L. (Note that because the x,w etc. are universally quantified over, we need not specify that the y_j be nonempty. The fact that some cases under the condition are vacuous does not imply that the condition itself is vacuous).

Def: $L \subseteq \Sigma^*$ has the block cancellation property if there is a k such that for all $x, w, y_1 .. y_k, w'$ in Σ^*, if $x = wy_1 .. y_k w'$ then there exist l, j, $1 \leq l \leq j \leq k$ such that $wy_1 .. y_{l-1} y_{j+1} \ldots y_k w' \in L$ iff $x \in L$.

Notation: If L has the block cancellation property for a particular k we shall say that $L \in \mathcal{L}_k$.

Theorem 2: Regularity, the block pumping property and the block cancellation property are equivalent.

Proof: (1) If L is regular, let A be an automaton accepting L and let k be the number of states of A. Let s^j be the state just after reading y_j then $s^0, s^1, \ldots s^k$ are k+1 occurrences of states and there must be l, j such that $l \neq j$ but $s^l = s^j$. Then $v = y_{l+1} .. y_j$ is the required pump for x relative to L.

(2) If L satisfies the block pumping property, then by taking i = 0, it satisfies the block cancellation property.

Thus the theorem reduces to the lemma below:

Lemma 1: The cancellation property implies regularity.

We shall use the following finite version of Ramsey's theorem. If X is a set, $X[2]$ denotes the set of all two element subsets of X. If X has n elements then $X[2]$ has $\frac{1}{2} n (n-1)$ elements.

Theorem: (Ramsey) For every k there is a number $r(k)$ such that if a set X has $r(k)$ elements or more and $X[2] = Z \cup Z'$ then there is a a $Y \subseteq X$ such that Y has at least k+1 elements and $Y[2] \subseteq Z$ or $Y[2] \subseteq Z'$.

The number $r(k)$ is usually denoted $N(k+1,k+1,2)$ corresponding to a more general statement of the theorem, but we shall use the simpler notation. In Lemmas 2 and 3, $\sum$ is fixed.

Lemma 2: $\mathcal{C}_k$ is finite. To be precise, if L,L' are in $\mathcal{C}_k$ and for all strings x with $|x| < r(k)$, $x \in L$ iff $x \in L'$ then $L = L'$.

Proof: The lemma claims that a language L in $\mathcal{C}_k$ is completely determined by a finite subpiece of it. If $\sum$ has n elements, $n > 1$, there are at most $m = n^{r(k)}$ strings of length $< r(k)$ and hence at most 2^m sets of such strings. Thus the lemma claims that $\mathcal{C}_k$ has at most 2^m languages in it.

Claim: We will show by induction on n that if $|x| = n$ then $x \in L$ iff $x \in L'$. This is clear if $n < r(k)$. Assume the claim for all $p < n$.

Suppose $|x| = n$ and $n \geq r(k)$. Write $x = w y_1 \ldots y_{r(k)} w'$ where all the y_j are nonempty Let $X = \{0, \ldots, r(k)\}$ and for $i,j \in X$, $i < j$, $\{i,j\} \in Z$ if $w y_1 \ldots y_{i-1} y_j \ldots y_{r(k)} w' \notin L$ and otherwise $\{i,j\} \in Z'$.

Then $X[2] = Z \cup Z'$ and by Ramsey's theorem, there is a Y with at least k+1 elements such that $Y[2] \subseteq Z$ or $Y[2] \subseteq Z'$.

In any case the elements of Y split x into k+2 pieces, the piece u before the first element of Y, the piece u' after the last element, and all the k intermediate pieces, $z_1 \ldots z_k$. Thus $x = uz_1 \ldots z_k u'$. We have two cases.

(i) $Y[2] \subseteq Z$. Then removing any consecutive block of z's from x corresponds to some set $\{1, j\}$ in $Y[2]$ and the shortened string x' is always in L. However, by the cancellation condition, there is some consecutive block of z's, whose removal leads to an x' such that $x' \in L$ iff $x \in L$. Hence $x \notin L$.

(ii) $Y[2] \subseteq Z'$. A similar argument tells us that $x \notin L$. Hence $x \in L$ iff there is a Y with k+1 elements etc. Such that $Y[2] \subseteq Z$.

Similarly for L'.

However, note that the condition on the right of the "iff" is the same for L and L' since for all strings x' shorter than x, $x' \in L$ iff $x' \in L'$. Thus for x also, we get, $x \in L$ iff $x \in L'$. This proves the claim and the lemma.

Lemma 3: (1) $L \in \mathcal{C}_k$ iff $\bar{L} \in \mathcal{C}_k$

(2) $\Lambda \in L$ or $\Lambda \in \bar{L}$

(3) Let $L_x = \{z \mid xz \in L\}$ then if $L \in \mathcal{C}_k$ then $L_x \in \mathcal{C}_k$.

Proof: (1, 2) are obvious. To see (3), note that $z \in L_x$ iff $xz \in L$. Now suppose that $z \in \Sigma^*$ and $z = wy_1 \ldots y_k w'$. Consider $xz = w''y_1 \ldots y_k w'$ where $w'' = xw$. Then $\exists l, j$, $1 \leq l \leq j \leq k$ such that

$$w'' y_1 \ldots y_{l-1} y_{j+1} \ldots y_k w' \in L \text{ iff } xz \in L,$$

Since $L \in \mathcal{C}_k$.

But then

$$wy_1 \ldots y_{l-1} y_{j+1} \ldots y_k w' \in L_x \text{ iff } z \in L_x.$$

Hence L_x is also in $\mathcal{C}_k$.

PROOF OF LEMMA 1: If L_o has the cancellation property we may as well assume that $\Lambda \notin L_o$ since otherwise we can work with $\bar{L}_o$ which is also in $\mathcal{L}_k$ and the regularity of $\bar{L}_o$ will imply that of L_o. We define the automaton A as follows:

$S = \mathcal{L}_k$,

s_o = Start state = L_o,

$M(L, \sigma)$ = next language (state) after reading σ

$= L_\sigma$

F = set of accepting states

$= \{L \mid \Lambda \in L\}$.

Clearly $M(L, x) = L_x$. For $L_\Lambda = L$ and $L_{x\sigma} = (L_x)_\sigma = M(M(L,x), \sigma)$ (induction hypothesis) $= M(L, x\sigma)$

Thus

A accepts x iff $M(L_o, x) \in F$

iff $\Lambda \in M(L_o, x)$

iff $\Lambda \in (L_o)_x$ iff $x\Lambda \in L_o$

iff $x \in L_o$.

Q.E.D.

We close this section by listing some open questions:

(1) Is there an analogue of theorem 2 for context free languages?

(2) The automation that we have constructed in the proof of theorem 2 which recognizes the language L_o, has a very large number of states. For a two element Σ, the number would be $2^{2^{r(k)}}$. By considering nondeterministic automata we see that there is a lower bound of 2^k. Can we bridge this gap?

(3) The block pumping lemma depends for its strength principally on the cancellation property, i.e. the case i = 0. Is there a pumping property which is positive, i.e. uses $i \geq 1$ only and which is equivalent to regularity?

Footnotes:

1. Theorem 1 and an approximate formulation of the block pumping condition are due to Parikh. The precise formulation of the block pumping condition and the proof of theorem 2 are due to Ehrenfeucht and Rozenberg.

2. Actually Beauquier's counter example is somewhat stronger. There is a marked pumping lemma where k distinct symbols in y are marked and the pump is required to contain one of the marked symbols. Beauquier shows that there is a CF language that satisfies the marked pumping condition but is not regular. Subsequent to our theorem 1, Vaughan Pratt showed (private communication) that there are $2^{\aleph_0}$ languages having the marked pumping property.

References:

[B] J. Beauquier, Private communication through R. Kosaraju.

[H] M.A. Harrison, Introduction to Formal Language Theory, Addison Wesley, 1978.

[J] J. Jaffe, A Necessary and Sufficient Pumping Lemma for Regular Languages, Sigact news, Summer 1978, pp. 48-49.

[R] F.P. Ramsey, The Foundations of Mathematics, Routledge and Kegan Paul (1954) pp 82-111. The paper in question is entitled "On a Problem of Formal Logic" and reprinted from Proc. Lon. Math. Soc. Ser 2 Vol.30 (1928) pp.338-84.

[N] A. Nerode, Linear Automata Transformations, Proc. Amer. Math. Soc. 9(1958) pp. 541-44.

[RS] M. Rabin and D. Scott, Finite Automata and their Decision Problems, IBM J. res. and dev. 3(1959) pp. 114-25.

...