# A note on the Pumping Lemma for regular languages

Cai, Jin-Yi
Department of Computer Science
State University of New York at Buffalo

Samuthiram, Karthikeyan
Department of Computer Science
State University of New York at Buffalo

December 4, 1996

**Abstract**

We discuss the non-sufficency of pumping lemma for regularity of languages.

During recent lectures to CS396 several pumping lemmas of varying strengths were discussed. This note discusses the necessity but non-sufficiency of these lemmas for regularity. See [1] for details.

There are different versions of the pumping lemma for regular languages.

**Version I (weak version)**
If $L$ is a regular language then $\exists$ a number $l$ such that $\forall s \in L$ and $|s| \geq l$, $\exists$ a way of splitting $s$ into $xyz$ such that $|y| > 0$ and $xy^iz \in L$, $\forall i \geq 0$.

**Version II**
If $L$ is a regular language then $\exists$ a number $l$ such that $\forall s \in L$ and $|s| \geq l$, $\exists$ a way of splitting $s$ into $xyz$ such that $|xy| \leq l$, $|y| > 0$ and $xy^iz \in L$, $\forall i \geq 0$.

**Version III (strong version)**
If $L$ is a regular language then $\exists$ a number $l$ such that $\forall s \in L$ and $\forall$ ways of splitting $s$ into $v_1wv_2$ such that $|w| \geq l$, $\exists$ a way of splitting $w$ into $xyz$ such that $|xy| \leq l$, $|y| > 0$ and $v_1xy^izv_2 \in L$, $\forall i \geq 0$.

If $L$ is regular, then it satisfies all the three versions of the pumping lemma. Of course, Version III implies Version II and Version II implies Version I.

But is this a sufficient condition for regularity? In other words if a language satisfies the pumping lemma, is it regular?

The answer is no.

For the weak version there is a relatively simple example.

Let $L_{composite}$= $\{0^m \mid m$ is composite$\}$.

**Claim:** $L_{composite}$ satisfies the weak version of the pumping lemma.

**Proof:**

Choose $l = 5$.

Let $s$ be a string $\in L_{composite}$ with $|s| \geq 5$.

Let $|s| = k$.

Since $s \in L_{composite}$, $k$ is composite.

Let $k = p \cdot q$ with $q \geq p > 1$.

Since $k \geq 5$ and $q \geq p$, $q > 2$.

Here $y$ is chosen as $0^p$.

$s = 0^k = x0^p z$.

$|x| + |z| = k - p$. The exact way of splitting $x, z$ is not important.

$xy^i z = 0^{k+(i-1)p} = 0^{pq+(i-1)p} = 0^{p(q+i-1)}$.

Since $q > 2$, $p(q+i-1)$ is composite $\forall i \geq 0$.

Therefore $xy^i z \in L_{composite}$, $\forall i \geq 0$.

**Claim:** However $L_{composite}$ doesn't satisfy version II of the pumping lemma.

**Proof:**

Let $l$ be the critical value.

Choose $s = 0^{pq}$ such that $p, q$ are primes and $p, q > l$.

$s \in L_{composite}$. Let there exist a way of splitting $s$ into $xyz$ such that $|xy| \leq l$ and $|y| > 0$.

Let $|y| = k$.

Therefore $xy^i z = 0^{pq+(i-1)k}$.

*Dirichlet's Theorem*: If $u, v$ are relatively prime and $v < u$, then $\exists i \geq 0$ such that $u + iv$ is prime.

Therefore by *Dirichlet*'s theorem there will exist an $i$ such that $pq + (i-1)k$ is prime since $pq$ and $k$ are relatively prime.

Therefore $L_{composite}$ doesn't satisfy version II of the pumping lemma.

Here is an example of a non-regular language that satisfies the Strong version of the pumping lemma.

Consider the following language $L$. $L$ is the union of two languages $L_1$ and $L_2$.

A string belonging to $L_1$ is concatenation of blocks, each separated by a 1. Each block is made up of runs of 2s separated by a 0. A run of 2s is a non empty sequence of 2s. As we move from left to right the number of runs in a block increases. In addition the string must begin and end with a 1. Each block must begin and end with a 0.

$L_2$ consists of all strings with two consecutive 0s.

Formally we define,

$L_1 = \{10(2^+0)^{i_1}10(2^+0)^{i_2}10(2^+0)^{i_3}\cdots 10(2^+0)^{i_k}1 |\ 0 < i_1 < i_2 < i_3 \cdots i_{k-1} < i_k$ and $k \geq 1\}$.

$L_2 = (1+0)^*00(1+0)^*$.

$L = L_1 \cup L_2$.

**Claim:** $L$ satisfies the strong version of pumping lemma.

**Proof:**

Choose the window size $l$ as 4.

Let $s \in L$ with $|s| \geq 4$.

case (a) $s \in L_2$.

If the window is placed on a substring containing two consecutive 0s choose $y$ as a symbol such that two consecutive 0s are still intact. After pumping $\forall i \geq 0$, the two 0s remain intact. Thus the pumped strings still belong to $L_2$ and hence belong to $L$.

If the window is not placed on a substring with two consecutive 0s any string in the window can be chosen as $y$. The pumped strings still belong to $L_2$ and hence to $L$.

case (b) $s \in L_1$.

Since the window size is 4 the window must contain at least one 2.

Choose $y = 2$.

case (i)

The 2 is an isolated 2, that is the symbols to its left and right are 0s.

When $i = 0$, the lone 2 is removed and 2 0s come together. Therefore $xy^0z \in L_2$ and hence to $L$.

When $i > 0$ the current run of 2s is made longer, but the total number of runs within the block remains the same. Therefore $xy^iz \in L_1$. Therefore $xy^iz \in L$.

case(ii)

The 2 has another 2 either to its left or right. Then after pumping the pumped string $xy^iz \in L_1$, $\forall i \geq 0$.

Therefore $L$ satisfies all the three versions of the pumping lemma.

**Claim:**Even though $L$ satisfies the pumping lemma, $L$ is not regular.

**Proof**

$L$ can be proved to be not regular by the Myhill Nerode theorem.

Consider $S$=$\{10(20)^i | i > 1\}$.

Let $x = 10(20)^i$ and $y = 10(20)^j$ be 2 distinct elements belonging to $S$.

wlog $i > j$.

Let $z = 10(20)^{j+1}1$.

$xz \notin L$ but $yz \in L$.

Thus $x$ and $y$ are distinguished by $z$.

Therefore, every element in $S$ is not equivalent to any other element in the relation defined in the Myhill Nerode theorem. Hence the number of equivalence classes is infinite.

By the Myhill Nerode theorem $L$ is not regular.

## References

[1] J.E.Hopcroft and J.D.Ullman. *Introduction to Automata Theory, Languages, and Computation*