Non-regular Languages
Theory of Computation (CSCI 3500)

Prof. Harley Eades (heades@gru.edu).

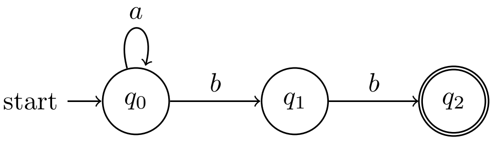
Read chapter 1.4

In summary, we have been investigating the theory and applications of regular languages. We began with deterministic finite automata (DFA) and gave the definition of a regular language, and then moved on to non-deterministic finite automata (NFA). NFAs turned out to be equivalent to DFAs, and thus, we obtained the lemma showing that a language is regular if and only if it can be accepted by an NFA. Following NFAs we introduced regular expressions which can be considered the programming language of regular languages. We showed that regular expressions can be compiled, or translated, into NFAs, and thus, a third result was obtained showing that a language is regular if and only if a regular expression recognizes it.

We now ask the question, are there languages that are not regular? That is, is there a language such that no regular expression can be defined that recognizes it, and hence, no NFA or DFA can be defined that accepts it? In this lecture we answer this question in the affirmative. However, how do we prove this? For example, given a language, L, how do we prove that L is non-regular? To answer this question we first describe a property of regular languages called the pumping lemma. Then to prove a language L is non-regular we assumes L is regular, and then use the pumping lemma to obtain a contradiction. Our first goal in this lecture is to understand the pumping lemma and its proof.

1 The Pumping Lemma

Consider the following DFA:



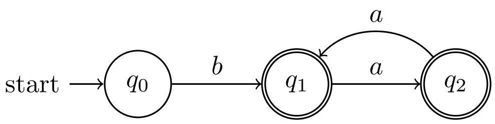
We call a sequence of states, $s_1, \ldots, s_n$, of a DFA, $M = (Q, \Sigma, \delta, q_0, F)$, such that $s_{i+1} = \delta(s_i, a)$ for some $a \in \Sigma$ a **run**. An **accepting run** is a run $s_1, \ldots, s_n$ where $s_1 = q_0$ and $s_n \in F$. An example run using the above example DFA is q_0, q_1, q_2, and it happens to be accepting. A non-accepting run would be q_0, q_1. Notice that there is a correspondence between runs and words, for example, the run q_0, q_0, q_1, q_2 corresponds to the word *aabb*. Accepting runs correspond to accepted words.

The run $r = q_0, q_0, q_1, q_2$ of the example DFA above is accepting. Let $x = q_0$, $y = q_0$, and $z = q_1, q_2$ be three runs. Note that $r = x, y, z$. What happens if we repeat y? We obtain the following:

$$\begin{aligned}
x, y, y, z &= q_0, q_0, q_0, q_1, q_2 \\
x, y, y, y, z &= q_0, q_0, q_0, q_0, q_1, q_2 \\
x, y, y, y, y, z &= q_0, q_0, q_0, q_0, q_0, q_1, q_2 \\
x, y, y, y, y, y, z &= q_0, q_0, q_0, q_0, q_0, q_0, q_1, q_2 \\
&\vdots \\
x, y^n, z &= q_0, \underbrace{q_0, \ldots, q_0}_{n}, q_1, q_2
\end{aligned}$$

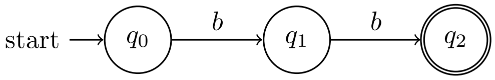
Furthermore, by inspection we can see that these are all accepting runs! Is it always possible to take a run and factor it into three pieces x, y, and z, such that y can be repeated any number times, where each new run is accepting? This question is certainly not obvious. It turns out that if the run, r, is big enough, then there is some x, y, and z such that $r = x, y, z$ and y can be repeated and always remain an accepting run. This property is called the pumping lemma.

Consider the following DFA:



Then $r = q_0, q_1$ is an accepting run, but notice that there is no factorization of r into x, y, and z such that we can repeat y and be an accepting run. However, notice that the run $r = q_0, q_1, q_2, q_1$ can be factored where $x = q_0$, $y = q_1, q_2$, and $z = q_1$. We can now see that x, y^i, z for any $i \in \mathbb{N}$ is an accepting run. So what's going on when we repeat y? The factorization we choose is very specific, and we choose it in such a way to insure that y is a cycle in the DFA. Then repeating y is equivalent to **repeatedly using the loop** in the DFA. What if there are no loops?

Consider the following DFA:



Here we can see that there are no accepting runs that will contain the location of a cycle. By inspecting all of the previous accepting runs that we could factor, and then repeat y, we see that the number of states in the initial run is equal to $n + 1$ where n is the number of states in the DFA. In fact, we will prove that in order to be able to factor an accepting run, and repeat y, the number of states in the run must be at least one more than the number of states in the DFA. So for the example above there is only one accepting run $r = q_0, q_1, q_2$ and its length is the size n and not the size $n+1$. Thus, there are no runs that can be factored.

Now we mentioned above that if a run has enough unique states then we could factor it and repeat y, but how big does it have to be? The size of r must be big enough to insure that if a loop exists in the DFA, then the loop had to be used at least once. Then we can always factor the run in such a way to capture the looping portion of r in y. Then repeating y is allowed.

The property we have been discussing can be characterized by the following lemma.

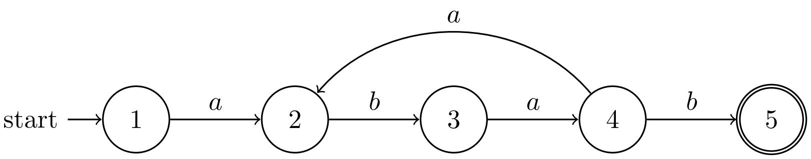
Lemma 1 (DFA Pumping Lemma)**.** *Suppose $M = (Q, \Sigma, \delta, q_0, F)$ is a DFA. Then there exists a p where if $r = s_1, \ldots, s_n$ is an accepting run of M and $n \geq p$, then r can be factored into three pieces $r = x, y, z$ satisfying the following requirements:*

1. *for any $i \in \mathbb{N}, x, y^i, z$ is an accepting run,*
2. *y contains at least one state, and*
3. *x, y has length at most p.*

Before we prove this result lets dissect the statement of the lemma first. The quantifiers of this result are first a universal quantifier over DFAs, and hence, this result applies to any DFA, and then there is an existential quantifier over the natural numbers. It states that there is some $p \in \mathbb{N}$. Then we universally quantify over runs such that the length of the run is at least p. Then we existentially quantify over the factorizations of r that meet the requirements of the pumping lemma. Thus, to prove this result we have to consider every DFA, but only find one p, such that, for every run of size at least p, there is at least one factorization. So we need to find only one p and only one factorization. There may be lots that do not work, but we need only one that does.

Proof. Suppose $M = (Q, \Sigma, \delta, q_0, F)$ is a DFA. Then we choose $p = |Q| + 1$. Let $w = s_1, \ldots, s_n$ be an accepting run where $n \geq p$. Since there are only $|Q|$ states there must be two of the same states among the first p states in w by the pigeonhole principle. We call the first of these two states s_l and s_r. We know $r \leq p$, because s_r occurs among the first p states of r. Using these facts we can now factor w into three pieces x, y, and z. Let $x = s_1, \ldots, s_l$, $y = s_{l+1}, \ldots, s_r$, and $z = s_{r+1}, \ldots, s_n$. By assumption we know $s_l = s_r$, and hence, $y = s_{l+1}, \ldots, s_l$. Thus, x, y^i, z for any $i \in \mathbb{N}$ is indeed an accepting run, because when we repeat y the last state just before z will always be s_l. The pigeon hole principle tells us that $l \neq r$, and hence, y contains at least one state. Finally, we know that $r \leq p$, and thus x, y has length at most p. Therefore, $p = |Q| + 1$ is the pumping length necessary to obtain the proper factorization of a run of M. □

Consider an example making use of the concepts in the previous proof.



Suppose we have the run $r = 1, 2, 3, 4, 2, 3, 4, 5$ which is at least $p = 6$. So we should be able to factor this run such that it meets the requirements of the pumping lemma. Following the construction in the proof we need to find the location of the first state that is repeated here that state is 2. We can number the positions as follows:

$$\begin{array}{cccccccc} s_1, & s_2, & s_3, & s_4, & s_5, & s_6, & s_7, & s_8 \\ 1, & 2, & 3, & 4, & 2, & 3, & 4, & 5 \end{array}$$

Now set $s_l = s_2$ and $s_r = s_5$. Clearly, $r = 5 \leq p$ just as the proof stated. We can now factor this run as follows:

$$\begin{aligned} x &= s_1, s_2 = 1, 2 \\ y &= s_3, s_4, s_5 = 3, 4, 2 \\ z &= s_6, s_7, s_8 = 3, 4, 5 \end{aligned}$$

Certainly, we can repeat y as many times as we want and always end up with an accepting run. That is, the run x, y^i, z for any $i \in \mathbb{N}$ is accepting. In addition, y contains at least one state, and x, y has length at most p. Thus, the factorization given in the proof is indeed the correct one.

We can prove the general pumping lemma for regular languages using the previous result. Keep in mind that the previous result reveals that the essence of the pumping lemma is factoring a word into three pieces where y is placed on a cycle of the corresponding DFA.

Lemma 2 (Pumping Lemma)**.** *Suppose L is a regular language. Then there exists a $p \in \mathbb{N}$ where if $w \in L$ and $|w| \geq p$, then w can be factored into three pieces, $w = xyz$, satisfying the following properties:*

1. *for any $i \in \mathbb{N}, xy^iz \in A$,*
2. *$|y| > 0, and$*
3. *$|xy| \leq p$.*

Proof. Suppose L is a regular language, and $M = (Q, \Sigma, \delta, q_0, F)$ is a DFA recognizing L. Now we must find some p such that any $w \in L$ of length at least p can be factored into x, y, and z satisfying conditions 1, 2, and 3.

Suppose $w = a_1 a_2 \cdots a_n \in A$ where $|w| = n$. Furthermore, suppose $r = r_1, \ldots, r_{n+1}$ is the sequence of states such that $r_{i+1} = \delta(r_i, a_i)$ for $1 \leq i \leq n$. Thus, r is a run, and since $w \in A$ then r must be an accepting run. Now by the pumping lemma for DFAs there must exist a p' such that $n + 1 \geq p'$, and r can be factored into three pieces $r = x', y', z'$ satisfying the following requirements:

1. for any $i \in \mathbb{N}, x', y'^i, z'$ is an accepting run,
2. y' contains at least one state, and
3. x', y' has length at most p'.

Finally, choose $p = p' - 1$, and factor w into xyz such that x is the word associated with x', and likewise for y, and z. Then since there is only one more state then character in a run we know that all the requirements of the pumping lemma are satisfied by the fact that the run $r = x', y', z'$ satisfies similar properties for runs. □

2 Some non-regular languages

The point of the pumping lemma is that it provides a means to prove that some languages are not regular. Suppose we are given a language L, and we want to prove that L is not regular. Then we can assume that it is and derive a contradiction, but what contradiction do we derive? This is the job of the pumping lemma. We know that if a language is regular, then the pumping lemma tells us that there exists a pumping length such that any word of size at least the pumping length can be factored in such a way to satisfy the conditions of the pumping lemma.

> **To prove the L is not regular we simply need to find a word in L that is at least the size of the pumping length, then show that for all possible factorizations of the word one of the conditions of the pumping lemma breaks.**

Lets put this to work on several examples. First, the prototypical example.

Example 3. *Show that $L = \{w \mid w \in \{0,1\}^*$ and $w = 0^n 1^n\}$ is non-regular.*

Suppose L is regular and derive a contradiction. We know by the pumping lemma – because L is regular – that there must exist a number p such that any $w \in L$ of size at least p can be factor into three pieces satisfying the conditions of the pumping lemma. Let $w = 0^p 1^p$. Then, certainly, $|w| = 2p \geq p$. Thus, we can factor $w = xyz$ satisfying the conditions of the pumping lemma. Now we consider every possible factorization of w and derive a contradiction. Notice that by condition 3 of the pumping lemma y can only contain 0's or the concatenation xy would have a size larger than p which is a contradiction. Thus, we only have one remaining factorization of w to consider:

> ***Case 1:*** *Suppose y contains all zeros. Then $xy^i z$ for any $i > 0$ has more zero's than ones, and hence is not in L a contradiction. Notice that if we pump down then we also get a contradiction. That is $xy^0 z = xz$ has more ones than zeros which is also a contradiction.*

Since all factorizations arrive a contradictions L cannot be regular.

Example 4. *Show that $L = \{w \mid w \in \{a, b, c, d, e\}$ and w is a palindrome$\}$ is non-regular.*

Assume L is regular and derive a contradiction. Furthermore, let p be the pumping length whose existence is a result of the pumping lemma. Then choose the word $w = a^p b a^p$. Now we consider every possible factorization *of w and derive a contradiction. Notice that by condition 3 again that y cannot contain a b or else $|xy| > p$ which is a contradiction. Thus, y must contain all a's. So we only have one case to consider:*

> ***Case 1:*** *Suppose y contains all a's. Then $xyyz$ must have more a's at the beginning than at the end, and thus, is no longer a palindrome. This is a contradiction.*

Since all factorizations arrive at contradictions L cannot be regular.

Example 5. *Show that $L = \{w \mid w \in \{a, b\}$ and w has the same number of a's and b's$\}$ is non-regular.*

Assume L is regular and derive a contradiction. Furthermore, let p be the pumping length whose existence is a result of the pumping lemma. Then choose the word $w = a^{p-1}bb^{p-2}$. Now we consider every possible factorization of w and derive a contradiction. However, notice that $x = a^{p-2}$, $y = ab$, and $z = b^{p-2}$ is a correct factorization. Thus, we have chosen a poor word.

Instead of choosing a word and using the pumping lemma one might notice that if we assume L is regular, then if we take the intersection of L with another regular language, then the result must be regular, but notice that $L \cap \{w \mid w \in \{a, b\}^*$ and $w = a^*b^*\} = \{w \mid w \in \{0, 1\}^*$ and $w = a^nb^n\}$, but this is a non-regular language. Thus, L cannot be regular.

Example 6. *Show that $L = \{w \mid w \in \{a, b\}$ and $|w|_a \leq |w|_b\}$ is non-regular.*

Assume L is regular and derive a contradiction. Furthermore, let p be the pumping length whose existence is a result of the pumping lemma. Then choose the word $w = a^pb^{p+1}$. Now we consider every possible factorization of w and derive a contradiction. Notice that by condition 3 y cannot contain a b or else $|xy| > p$ which is a contradiction. Thus, y must contain all a's. So we only have one case to consider:

> ***Case 1:*** *Suppose y contains all a's. Then $xyyz$ must have at least the same number a's at the beginning than at the end. This is a contradiction.*

Since all factorizations arrive at contradictions L cannot be regular.

Example 7. *Show that $L = \{w \mid w \in \{a, b\}$ and $w = a^ib^j$ where $i \neq j$ and $i, j \in \mathbb{N}\}$ is non-regular.*

The difficultly with showing this language is non-regular lies in choosing the right word. Suppose we choose $w = a^pb^{p+1}$, but this word can be factored: $x = a^{p-2}, y = aa, z = b^{p+1}$, then $xy^0z = xz \in L$, $xy^1z = xyz \in L$, $xy^2z = xyyz = a^{p+2}b^{p+1} \in L$, etc. In fact, any word that has the shape of just adding a constant to one of the indices will be factorable. To show L is non-regular we have to get creative.

Assume L is regular and derive a contradiction. Furthermore, let p be the pumping length whose existence is a result of the pumping lemma. Then choose the word $w = a^{p!}b^{(p+1)!}$. Now we consider every possible factorization of w and derive a contradiction. Notice that by condition 3 y cannot contain a b or else $|xy| > p$ which is a contradiction. Thus, y must contain all a's. Furthermore, notice that pumping down does not give us a contradiction, but pumping up will. We derive a contradiction by showing that for any $i > 1$ the word xy^iz has the same number of a's as b's.

Let $1 < i \in \mathbb{N}$ and $|y| = k \in \mathbb{N}$. Then $|xy^iz|_b = |z| = (p + 1)!$ and

$$\begin{array}{rcl} |xy^iz|_a & = & |x| + |y^i| \\ & = & |x| + |y| + |y^{i-1}| \\ & = & |xy| + |y^{i-1}| \\ & = & p! + |y^{i-1}| \\ & = & p! + (i-1)k \end{array}$$

We obtain a contradiction if we can show that

$$|xy^iz|_a = p! + (i - 1)k = (p + 1)! = |xy^iz|_b$$

First, solve for i:

$$\begin{aligned}
p! + (i-1)k &= (p+1)! \\
(i-1)k &= (p+1)! - p! \\
i - 1 &= \tfrac{(p+1)!-p!}{k} \\
i &= \tfrac{(p+1)!-p!}{k} + 1 \\
i &= \tfrac{(p!(p+1))-p!}{k} + 1 \\
i &= \tfrac{p!((p+1)-1)}{k} + 1 \\
i &= \tfrac{p!p}{k} + 1
\end{aligned}$$

Finally, substituting for i:

$$\begin{aligned}
p! + (i-1)k &= p! + ((\tfrac{p!p}{k} + 1) - 1)k \\
&= p! + (\tfrac{p!p}{k})k \\
&= p! + p!p \\
&= p!(1+p) \\
&= (p+1)!
\end{aligned}$$

Example 8. *Show that $L = \{w \mid w \in \{a\}$ and $w = a^{i^2}$ where $i \in \mathbb{N}\}$ is non-regular.*

Assume L is regular and derive a contradiction. Furthermore, let p be the pumping length whose existence is a result of the pumping lemma. Then choose the word $w = a^{(p+1)^2} \in L$. Now we consider every possible factorization of $w = xyz$ and derive a contradiction. Notice that by condition 2 and 3 $y = a^j$ for $0 < j \leq p$, or else we arrive at a contradiction. Then it suffices to show that $xy^0z = a^{(p+1)^2-j} \notin L$. This follows if we can show that $(p+1)^2 - j$ is not a square of some natural number. Consider the closest square less than $(p+1)^2$, which is, p^2. In addition, we know that $j \leq n$, and so, $(n+1)^2 - j = n^2 + 2n + 1 - i > n^2$. Therefore, $(n+1)^2 > (n+1)^2 - j \geq n^2 + n + 1 > n^2$, and thus, we arrive at our contradiction.