Theory of Automata

Closure properties and Pumping Lemma for Regular Languages

Dr. Sabina Akhtar

Revision

- Design DFA for
  - $L = \{w \mid w$ is of the form 0^n1^n , for all $n \geq 0\}$
  - $L1 = \{w \mid w$ is of the form 0^n , for all $n \geq 0\}$
  - $L2 = \{w \mid w$ is of the form 0^n1^n , for all $n < 10000000\}$

- Why it is not possible? Informally state.
- When language is regular, how do we prove it?

PUMPING LEMMA FOR REGULAR LANGUAGES

Regular Languages

- Finite regular language
  - Always regular?
- Infinite regular language
- Examples?

The Infiniteness Problem

- Is a given regular language infinite?
- Start with a DFA for the language.
- Key idea: if the DFA has n states, and the language contains any string of length n or more, then the language is infinite.
- Otherwise, the language is surely finite.
  - Limited to strings of length $n-1$ or less.

Proof of Key Idea

- If an n-state DFA accepts a string w of length *n* or more, then there must be a state that appears twice on the path from the start state to a final state.
- Because there are at least n+1 states along the path.

Regular or not?

When is a language regular?

if we are able to construct one of the following: DFA *or* NFA *or* ε -NFA *or* regular expression

When is it not?

If we can show that no FA can be built for a language

How to prove languages are ***not*** regular?

What if we cannot come up with any FA?

A) Can it be language that is not regular?

B) Or is it that we tried wrong approaches?

How do we *decisively* prove that a language is not regular?

Examples

- $L_1 = \{w \mid w \text{ is of the form } 0^n \text{ , for all } n \geq 0\}$
- $L_2 = \{w \mid w \text{ is of the form } 0^n 1^n \text{ , for all } n \geq 0\}$

The Pumping Lemma for Regular Languages

- **What it is?**
  The Pumping Lemma is a property of all regular languages.
- **How is it used?**
  A technique that is used to show that a given language is not regular

Pumping Lemma for Regular Languages

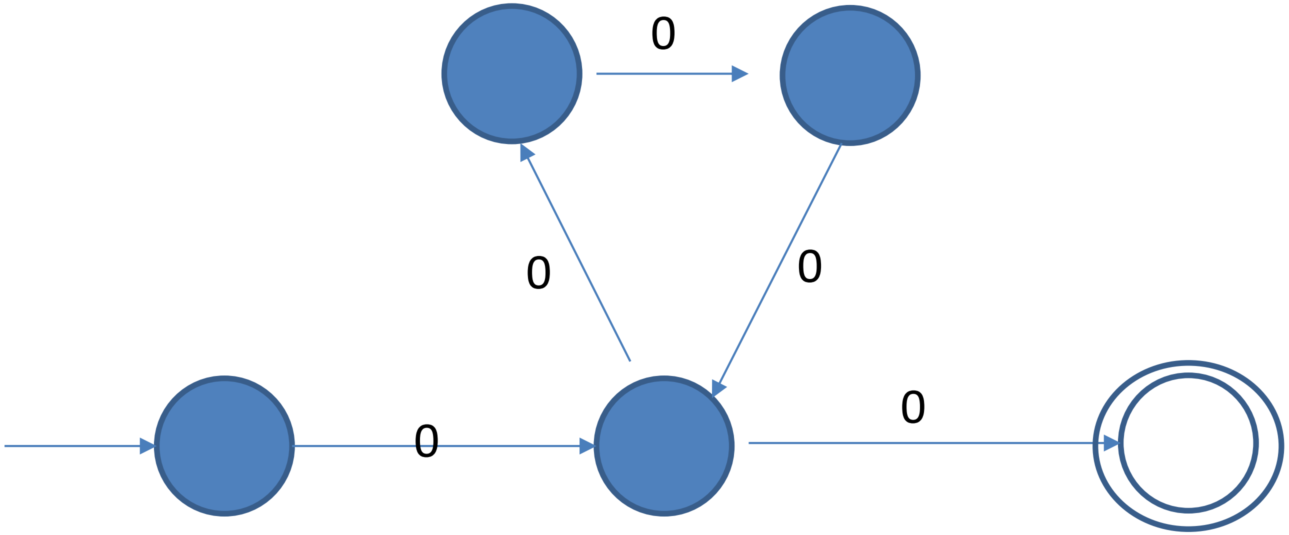
Let L be a regular language

Then ***there exists*** some constant ***N*** such that ***for every*** string $w \in L$ s.t. $|w| \geq N$, ***there exists*** a way to break w into three parts, $w=xyz$, such that:

1. $y \neq \varepsilon$
2. $|xy| \leq N$
3. For all $k \geq 0$, all strings of the form $xy^kz \in L$

Definition: *N* is called the "Pumping Lemma Constant"

This property should hold for all regular languages.



00

00000

Pumping Lemma: Proof

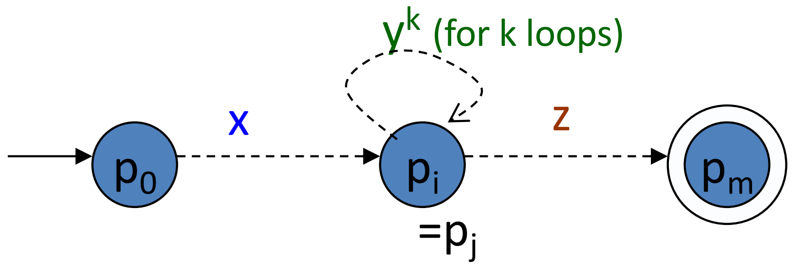
- L is regular => it should have a DFA.
  - Set N := number of states in the DFA
- Any string $w \in L$, s.t. $|w| \geq N$, should have the form:

  $w = a_1 a_2 \ldots a_m$, where $m \geq N$
- Let the states traversed after reading the first N symbols be: $\{p_0, p_1, \ldots p_N\}$
  - ==> There are N+1 p-states, while there are only N DFA states
  - ==> at least one state has to repeat

    i.e, $p_i = p_J$ where $0 \leq i < j \leq N$ (by PHP)

Pumping Lemma: Proof…

- => We should be able to break $w=xyz$ as follows:
  - $x=a_1a_2..a_i$; $y=a_{i+1}a_{i+2}..a_J$; $z=a_{J+1}a_{J+2}..a_m$
  - x's path will be $p_0..p_i$
  - y's path will be $p_i\,p_{i+1}..p_J$ (but $p_i=p_J$ implying a loop)
  - z's path will be $p_Jp_{J+1}..p_m$
- Now consider another string $w_k=xy^kz$, where $k\geq 0$

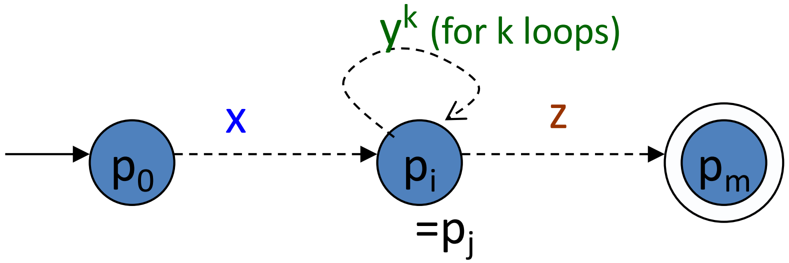


- Case k=0
  - DFA will reach the accept state p_m
- Case k>0
  - DFA will loop for y^k, and finally reach the accept state p_m for z
- In either case, $w_k \in L$

This proves part (3) of the lemma

Pumping Lemma: Proof…

- For part (1):
  - Since i<j, y ≠ ε



- For part (2):
  - By PHP, the repetition of states has to occur within the first N symbols in w
  - ==> |xy|≤N

$\Box$

The Purpose of the Pumping Lemma for RL

- To prove that some languages *cannot be* regular.

Example of using the Pumping Lemma to prove that a language is not regular

Let L = {w | w is of the form 0^n1^n , for all $n\geq0$}

- Your Claim: L is not regular
- Proof:
  - By contradiction, let L be regular
  - P/L constant should exist
    - Let *N* = that P/L constant
  - Consider input w = 0^N1^N s.t. $|w|\geq N$
    (your choice for the template string)
  - By pumping lemma, we should be able to break w=xyz, such that:
    1) $y\neq \varepsilon$
    2) $|xy|\leq N$
    3) For all $k\geq0$, the string xy^kz is also in L

Proof…

Template string $w = 0^N 1^N = \underbrace{00 \ldots 0}_{N}\underbrace{11 \ldots 1}_{N}$

- Because $|xy| \leq N$, xy should contain only 0s
  - (This and because $y \neq \varepsilon$, implies $y = 0^+$)
- Therefore x can contain *at most* N-1 0s
- Also, all the N 1s must be inside z
- One possible division is
- $x = 0^{N-1}$, $y = 0$, $z = 1^N$
- By (3), any string of the form $xy^kz \in L$ for all $k \geq 0$
- Case k=0: xz has at most N-1 0s but has N 1s
- Therefore, $xy^0z \notin L$
- This violates the P/L (a contradicion)

Setting k=0 is referred to as **"pumping down"**

Setting k>1 is referred to as **"pumping up"**

Exercise 2

Prove $L = \{0^n 1^m \mid n < m\}$ *is not regular*

$$w_1 = 0^{N-1} 1^N$$
$$w_2 = 0^N 1^{N+1}$$

Exercise 3

Prove $L = \{0^n 1 0^n \mid n \geq 1\}$ is not regular

<u>Note:</u> This n is not to be confused with the pumping lemma constant N. That *can* be different.

In other words, the above question is same as proving:

- $L = \{0^m 1 0^m \mid m \geq 1\}$ is not regular

References

- Book Chapter 4
- Lectures from Washington State University
  - http://www.eecs.wsu.edu/~ananth/CptS317/Lectures/
- Lectures from Stanford University
  - http://infolab.stanford.edu/~ullman/ialc/spr10/spr10.html#LECTURE%20NOTES