MTL783: Theory of Computation 1st Semester, 2025-2026

Lecture 08— 25/08/2025

Lecturer: Prof Minati De *Scribe: Team 9*

Scribed by:

1. Astha Lohia (2022MT11927)
2. Kaneesha Jain (2022MT11929)
3. Shreya Jain (2022MT11914)
4. Tanya Jain (2022MT11935)
5. Tejas Asija (2022MT61890)

1 Overview

In the previous lecture, we studied the properties of regular languages, proved the equivalence between regular languages and regular expressions, and learned how to use GNFA to obtain regular expressions. We also began discussing how to prove whether a language is regular or not.

In this lecture, we will learn about the Pumping Lemma, an important tool used to show that certain languages are not regular.

2 Pumping Lemma

2.1 Pumping Lemma for Regular Languages

Theorem. Let L be an infinite regular language. Then there exists some positive integer m such that any $w \in L$ with $|w| \geq m$ can be decomposed as

$$w = xyz$$

with the following conditions:

$$|xy| \leq m, \quad |y| \geq 1,$$

and

$$w_i = xy^iz \in L \quad \text{for all } i = 0, 1, 2, \ldots$$

To paraphrase this, every sufficiently long string in L can be broken into three parts in such a way that an arbitrary number of repetitions of the middle part yields another string in L. We say that the middle string is "pumped," hence the term *pumping lemma* for this result.

Proof. Since L is regular, there exists a DFA $M = (Q, \Sigma, \delta, q_0, F)$ that recognizes L. Let $|Q| = n$ and set $m = n + 1$.

Choose any string $w \in L$ with $|w| \geq m$. (Such a choice is possible because L is infinite.) When M processes w starting from q_0, it visits a sequence of states

$$q_0, q_1, q_2, \ldots, q_{|w|}$$

(where $q_k = \delta^*(q_0, \text{prefix of } w \text{ of length } k)$). This sequence has $|w|+1$ entries. Because $|w| \geq m = n+1$ and there are only n distinct states, by the pigeonhole principle at least one state must repeat among the first m visited states; hence some state q_r appears twice within the first $m+1$ positions of the sequence.(Figure 1)

Therefore we can split w into three parts $w = xyz$ corresponding to the transitions:

$$\delta^*(q_0, x) = q_r, \qquad \delta^*(q_r, y) = q_r, \qquad \delta^*(q_r, z) = q_f,$$

where $q_f \in F$ is the final state reached after the whole string. By construction the repetition occurs within the first m symbols, so $|xy| \leq m$, and because a nonempty portion produced the loop we have $|y| \geq 1$.

Now for any $i \geq 0$,

$$\delta^*(q_0, xy^iz) = \delta^*(\delta^*(q_0, x), y^iz) = \delta^*(q_r, y^iz).$$

Since $\delta^*(q_r, y) = q_r$, repeated application gives $\delta^*(q_r, y^i) = q_r$ for all $i \geq 0$, hence

$$\delta^*(q_0, xy^iz) = \delta^*(q_r, z) = q_f \in F.$$

Thus $xy^iz \in L$ for every $i = 0, 1, 2, \ldots$, which completes the proof. $\square$

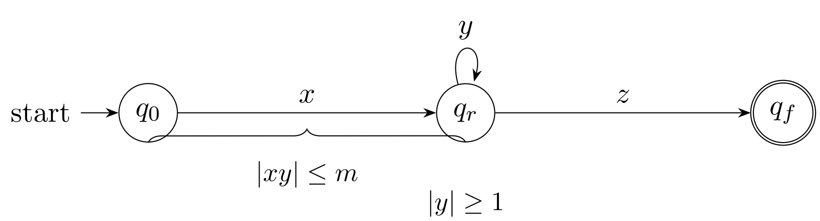

Figure 1: Cycle at q_r for the decomposition $w = xyz$, with $|xy| \leq m$ and $|y| \geq 1$.

2.2 Examples

Example 1. Use the Pumping Lemma to show that

$$L = \{a^nb^n : n \geq 0\}$$

is not a regular language.

Solution. Assume, for contradiction, that L is regular. By the Pumping Lemma there is a pumping length $m \geq 1$ such that every string $w \in L$ with $|w| \geq m$ can be written $w = xyz$ satisfying

$$|xy| \leq m, \qquad |y| \geq 1,$$

and

$$xy^iz \in L \quad \text{for all } i = 0, 1, 2, \ldots$$

Choose the particular string

$$w = a^m b^m \in L.$$

(The language L is infinite, so such a choice with $|w| \geq m$ is always possible.) Because $|xy| \leq m$, the substring xy lies entirely in the block of a's at the start of w. Hence the substring y consists only of a's; write $|y| = k$ with $k \geq 1$.

Now apply the pumping lemma with $i = 0$. The pumped-down string is

$$w_0 = xy^0z = xz = a^{m-k}b^m.$$

But w_0 has $m - k$ a's followed by m b's, and therefore is *not* of the form a^nb^n (its number of a's and b's differ). That is, $w_0 \notin L$, contradicting the Pumping Lemma requirement that $xy^iz \in L$ for all $i \geq 0$.

This contradiction shows the assumption that L is regular is false. Therefore $L = \{a^nb^n : n \geq 0\}$ is not regular. □

Example 2. Show that

$$L = \{\, ww^R : w \in \Sigma^* \,\}$$

is not regular (here w^R denotes the reverse of w).

Solution. Assume, for contradiction, that L is regular. Let m be the pumping length given by the Pumping Lemma.

Pick the palindrome

$$u = a^m b a^m,$$

and consider the string

$$s = uu = (a^m b a^m)(a^m b a^m) \in L.$$

Note $|s| \geq m$, so s can be decomposed $s = xyz$ with $|xy| \leq m$ and $|y| \geq 1$. Because $|xy| \leq m$, the substring xy lies entirely in the first block of a's of s. Hence y consists only of a's; write $|y| = k \geq 1$ (Figure 2) .

Pump down ($i = 0$) to obtain

$$s_0 = xz.$$

The effect of removing y is to reduce the number of leading a's before the first b in s. However, the right half of s (the second copy of u) is unchanged, so the symmetry required for a string of the form ww^R is lost: the left part and its corresponding mirror on the right no longer match. Thus $s_0 \notin L$, contradicting the Pumping Lemma. Therefore L is not regular. □

Example 3. Let $\Sigma = \{a, b\}$. Show that the language

$$L = \{\, w \in \Sigma^* : n_a(w) < n_b(w) \,\}$$

(is the set of strings with strictly fewer a's than b's) is not regular.

Solution. Suppose L is regular and let m be the pumping length. Choose

$$w = a^m b^{m+1} \in L.$$

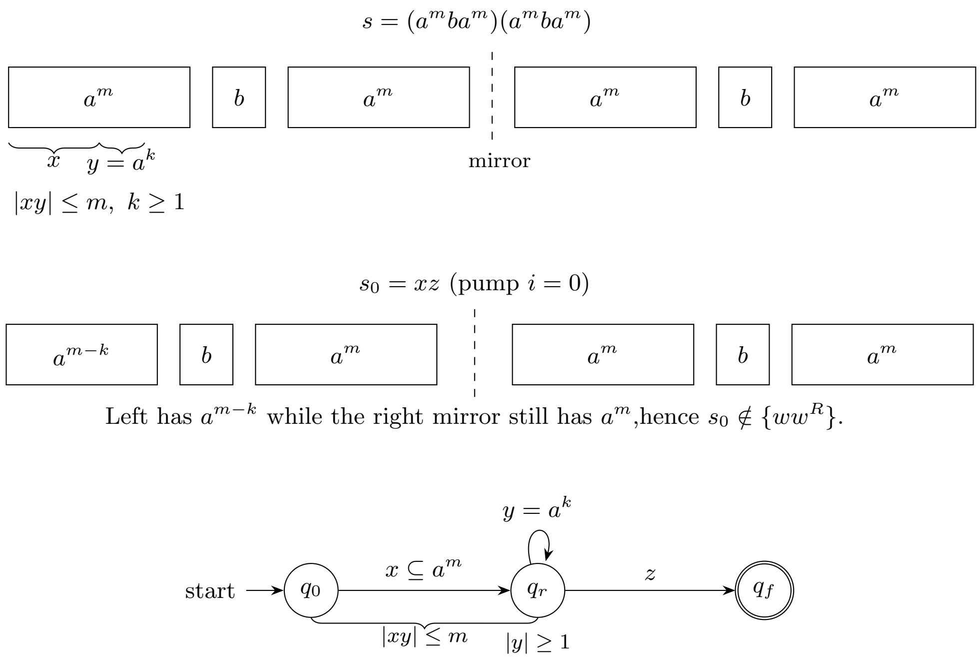

Figure 2: Top: the chosen s and the placement of x, y. Middle: $s_0 = xz$ after pumping down. Bottom: DFA fragment showing the loop for $y = a^k$.

Because $|xy| \leq m$, the substring y lies entirely inside the initial a^m block; hence $y = a^k$ for some $1 \leq k \leq m$.

Pump up with $i = 2$ to get

$$w_2 = xy^2z = a^{m+k}b^{m+1}.$$

Now $n_a(w_2) = m + k$ and $n_b(w_2) = m + 1$. Since $k \geq 1$, $m + k \geq m + 1$, so $n_a(w_2) \geq n_b(w_2)$. In particular, $w_2 \notin L$. This contradicts the Pumping Lemma, so L is not regular. □

Example 4. Show that

$$L = \{(ab)^n a^k : n > k, \; k \geq 0\}$$

is not regular.

Solution. Assume L is regular and let m be the pumping length. Choose

$$w = (ab)^{m+1}a^m \in L,$$

since here $n = m + 1$ and $k = m$, so $n > k$.

Decompose $w = xyz$ with $|xy| \leq m$ and $|y| \geq 1$. Because the first $2(m + 1)$ symbols of w are the (ab)-block and $|xy| \leq m$, both x and y lie entirely inside the prefix that consists of (ab)-repetitions. Thus y is one of: a, b, or ab (but not a substring that crosses more than one symbol boundary beyond these, because $|y| \leq m$ and the structure is repetitive).

Consider cases:

- If $y = a$: pump down ($i = 0$). The resulting string has fewer a's in some (ab) pair, so the number of (ab) pairs in the prefix may be unchanged but the total count of trailing a's relative to (ab) pairs can violate $n > k$. Concretely, the form no longer fits the constraint $n > k$ (it may produce an $(ab)^n a^k$ with $n \leq k$), so $xy^0 z \notin L$.

- If $y = b$: similarly pumping yields a string where a b in an (ab) pair is lost or duplicated; one can choose $i = 0$ to make the pattern mismatch the form $(ab)^n a^k$ with the required inequality.
- If $y = ab$: pumping with $i = 0$ removes an entire (ab), reducing n by 1 but leaving k unchanged, so the inequality $n > k$ may fail (specifically we can reach $n = k$), and hence $xy^0 z \notin L$.

In each case we get a contradiction to the Pumping Lemma. Therefore L is not regular. □

Example 5. Show that

$$L = \{a^n : n \text{ is a perfect square}\}$$

is not regular.

Solution. Assume L is regular and let m be the pumping length. Choose

$$w = a^{m^2} \in L.$$

Write $w = xyz$ with $|xy| \leq m$ and $|y| \geq 1$. Then $y = a^k$ for some $1 \leq k \leq m$. Pump down ($i = 0$) to get

$$w_0 = xz = a^{m^2 - k}.$$

We claim $m^2 - k$ is not a perfect square. Observe

$$(m-1)^2 = m^2 - 2m + 1 < m^2 - m \leq m^2 - k < m^2,$$

where the left inequality holds for $m \geq 2$. Thus

$$(m-1)^2 < m^2 - k < m^2,$$

so $m^2 - k$ lies strictly between two consecutive perfect squares; hence it is not a perfect square. Consequently $w_0 \notin L$, contradicting the Pumping Lemma. Therefore L is not regular. □

Example 6. Show that

$$L = \{\, a^n b^k c^{n+k} : n \geq 0,\ k \geq 0 \,\}$$

is not regular.

Solution (using closure under homomorphism). Define a homomorphism $h : \{a, b, c\}^* \to \{a, c\}^*$ by

$$h(a) = a, \qquad h(b) = a, \qquad h(c) = c.$$

Apply h to L:

$$h(L) = \{\, h(a^n b^k c^{n+k}) = a^{n+k} c^{n+k} : n, k \geq 0 \,\} = \{\, a^i c^i : i \geq 0 \,\}.$$

But $\{a^i c^i : i \geq 0\}$ is known to be non-regular (it is isomorphic to $\{a^n b^n\}$). Since regular languages are closed under homomorphism, if L were regular then $h(L)$ would be regular. This contradiction shows L is not regular. □

Example 7. Show that

$$L = \{a^n b^l : n \neq l\}$$

is not regular.

Solution 1 (pumping-lemma factorial trick). Suppose L is regular and let m be the pumping length. Choose

$$n = m!, \qquad l = (m+1)!,$$

and consider the string

$$w = a^{m!} b^{(m+1)!} \in L \qquad (\text{since } m! \neq (m+1)!).$$

Write $w = xyz$ with $|xy| \leq m$ and $|y| \geq 1$. Then $y = a^k$ for some $1 \leq k \leq m$. Pump by choosing an integer i so that the number of a's becomes $(m+1)!$. We want

$$m! + (i-1)k = (m+1)!.$$

Solve for i:

$$i = 1 + \frac{(m+1)! - m!}{k} = 1 + \frac{m!\, m}{k} = 1 + m! \cdot \frac{m}{k}.$$

Because $1 \leq k \leq m$, the fraction $\frac{m}{k}$ is an integer, so i is an integer ≥ 1. Pumping with this i produces a string with $(m+1)!$ a's and $(m+1)!$ b's, i.e., $a^{(m+1)!} b^{(m+1)!}$ which has equal numbers of a's and b's, so it is not in L. This contradicts the Pumping Lemma, so L is not regular.

Solution 2 (closure properties — simpler). Suppose L were regular. Consider

$$L_1 = L \cap a^* b^*.$$

Regular languages are closed under intersection, so L_1 would be regular. But

$$L_1 = \{a^n b^n : n \geq 0\}^c \cap a^* b^* = a^* b^* \setminus \{a^n b^n : n \geq 0\},$$

and more directly,

$$L_1 = \{a^n b^l : n \neq l\} \cap a^* b^* = \{a^n b^l : n \neq l\}.$$

Intersecting L with the regular language $a^* b^*$ isolates the strings of the simple form $a^n b^l$. If L_1 were regular, then its complement within $a^* b^*$, which is $\{a^n b^n : n \geq 0\}$, would also be regular (regular languages are closed under complement). But we know $\{a^n b^n : n \geq 0\}$ is not regular. This contradiction shows that L is not regular. $\square$

3 Conclusion

In this lecture, we introduced the **Pumping Lemma** and applied it to prove the non-regularity of several important languages. The lemma is a powerful tool for showing that certain languages cannot be expressed by any regular expression or DFA. We also saw how closure properties of regular languages can sometimes provide simpler proofs of non-regularity.

References

[1] Peter Linz, An Introduction to Formal Languages and Automata, 6th Edition, Jones Bartlett Learning, 2016