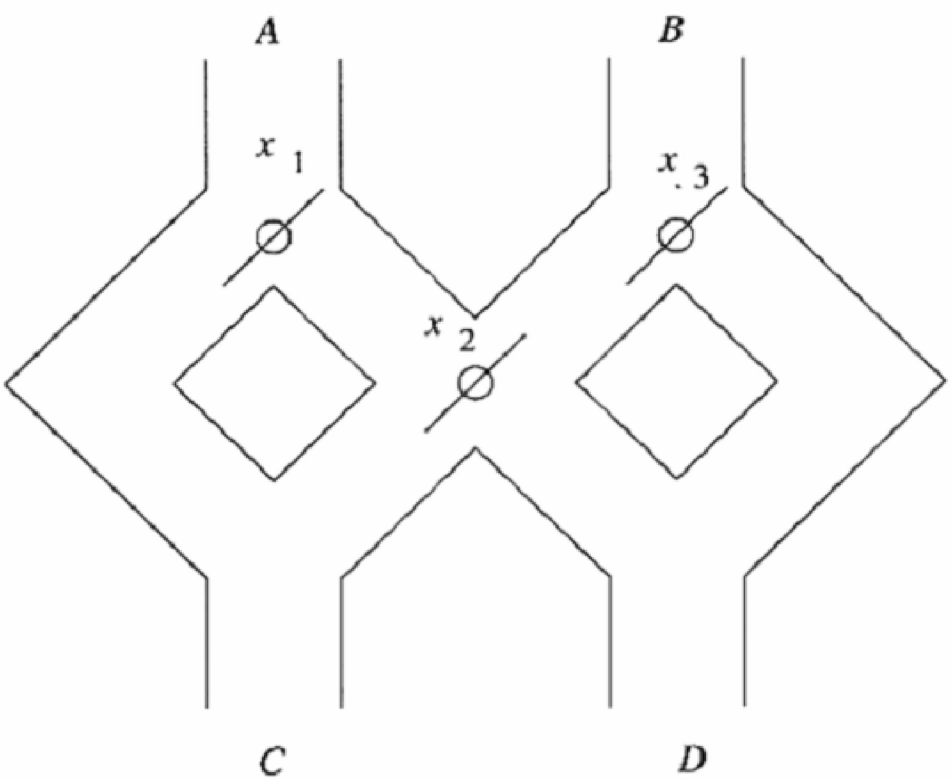
# Formal Languages and Automata Recitation-2

Tacettin Ayar (ayart@itu.edu.tr)

# Problem-1

**Exercise 2.2.1:** In Fig. 2.8 is a marble-rolling toy. A marble is dropped at $A$ or $B$. Levers $x_1$, $x_2$, and $x_3$ cause the marble to fall either to the left or to the right. Whenever a marble encounters a lever, it causes the lever to reverse after the marble passes, so the next marble will take the opposite branch.



# Problem-1(Continued)

- Model this toy by a finite automaton.
- Let the **inputs A** and **B** represent the input into which the marble is dropped.
- Let **acceptance** correspond to the marble exiting at **D**; **nonacceptance** represents a marble exiting at **C**.

# Problem-1(Solution)

- States correspond to the eight combinations of switch positions, and also must indicate whether the previous roll came out at D, i.e., whether the previous input was accepted:
  - Let **L** represent a position to the left (as in the diagram) and **R** a position to the right.
  - Each state can be represented by a sequence of three L's or R's, representing the directions of the three switches, in order from left to right.
  - We follow these three letters by either **a** indicating it is an accepting state or **r** indicating it is a rejection state.
- Of the 16 possible states, it turns out that only 13 are accessible from the initial state, LLLr.

# Problem-1(Solution)

Here is the transition table:

| | A | B |
|---|---|---|
| LLLr | RLLr | LRRr |
| *LLLa | RLLr | LRRr |
| *LLRa | RLRr | LLLa |
| LRLr | RRLr | LLRa |
| *LRLa | RRLr | LLRa |
| LRRr | RRRr | LRLa |
| RLLr | LRLr | RRRr |
| *RLLa | LRLr | RRRr |
| RLRr | LRRr | RLLa |
| *RLRa | LRRr | RLLa |

# Problem-2: ε-NFA Accepting Decimal Numbers

Fig. 2.1 is an $\epsilon$-NFA that accepts decimal numbers consisting of:

1. An optional + or − sign,
2. A string of digits,
3. A decimal point, and
4. Another string of digits. Either this string of digits, or the string (2) can be empty, but at least one of the two strings of digits must be nonempty.

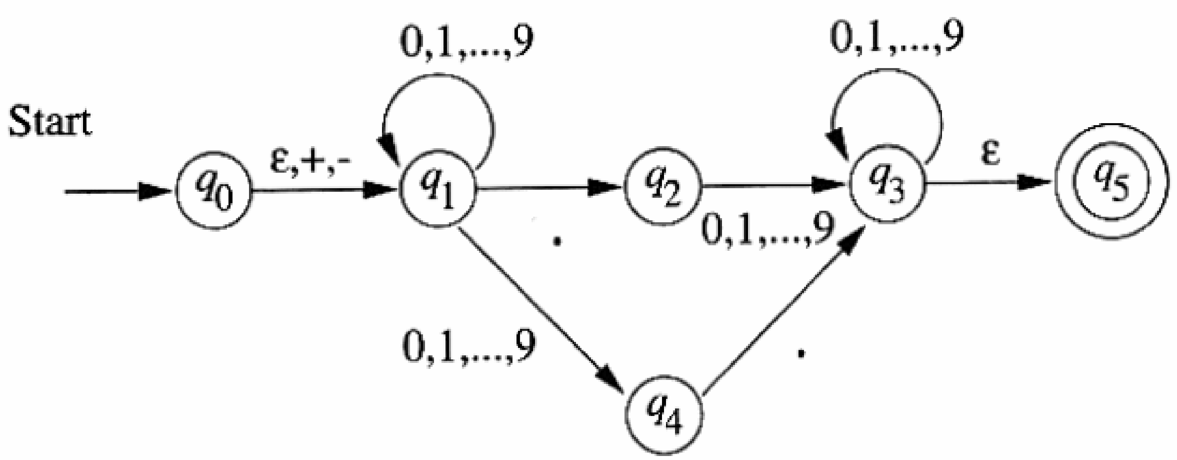



Figure 2.1: An $\epsilon$-NFA accepting decimal numbers

# Problem-2: ε-NFA to DFA

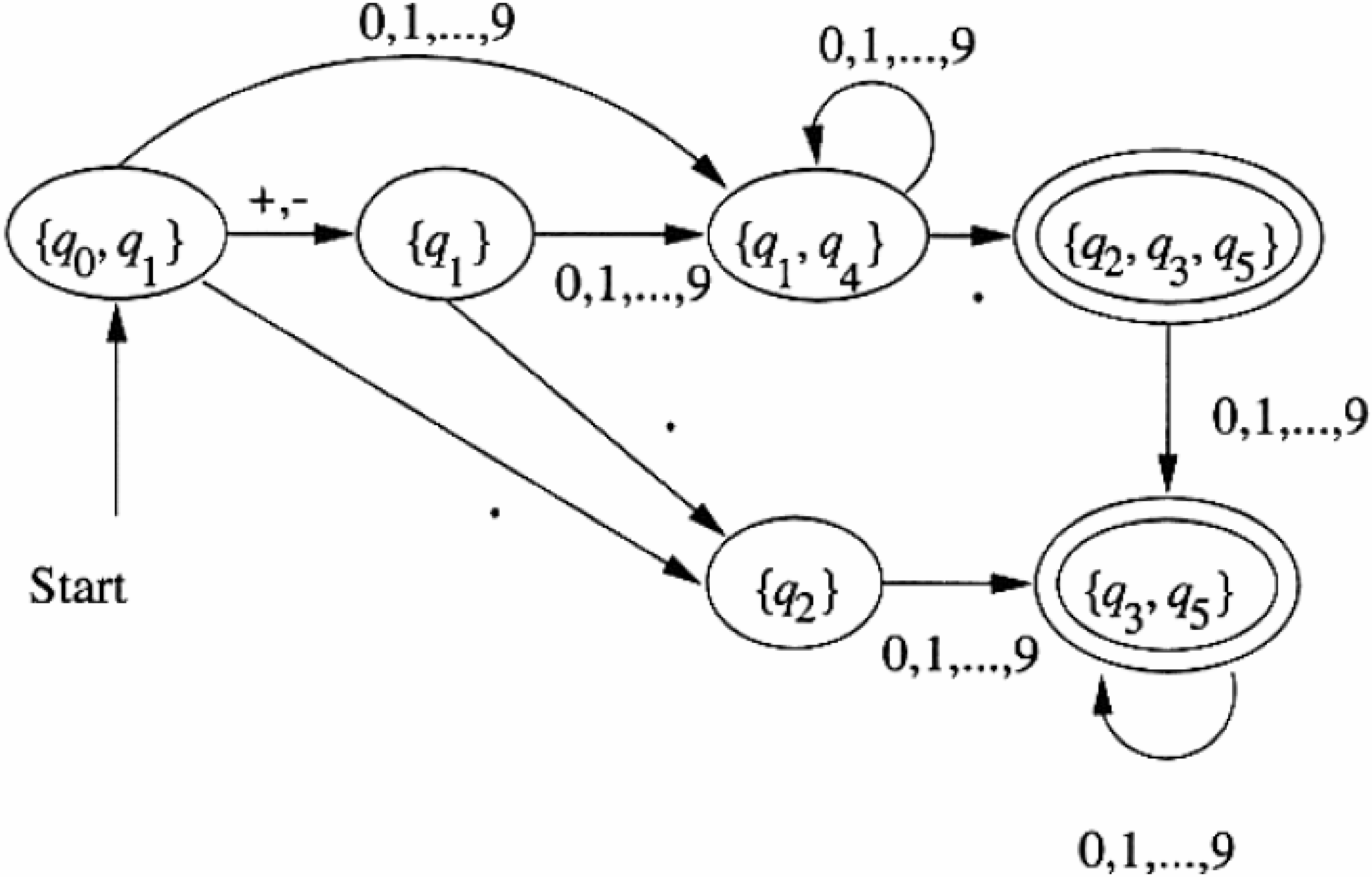
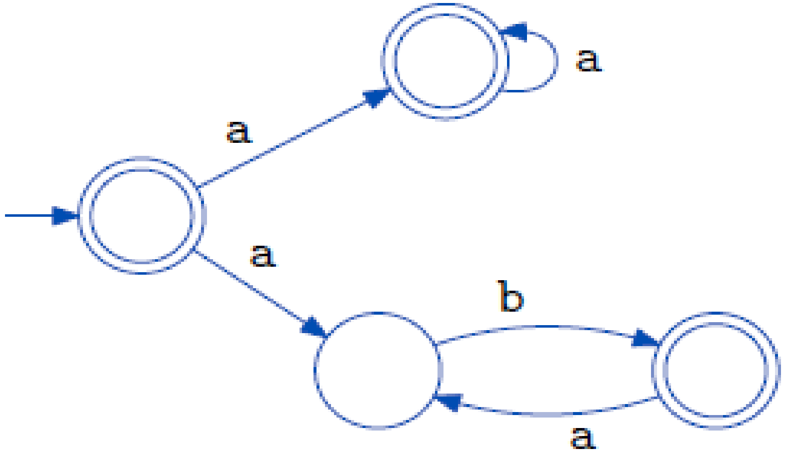



Figure 2.2: The DFA $D$ that eliminates $\epsilon$-transitions from Fig. 2.1

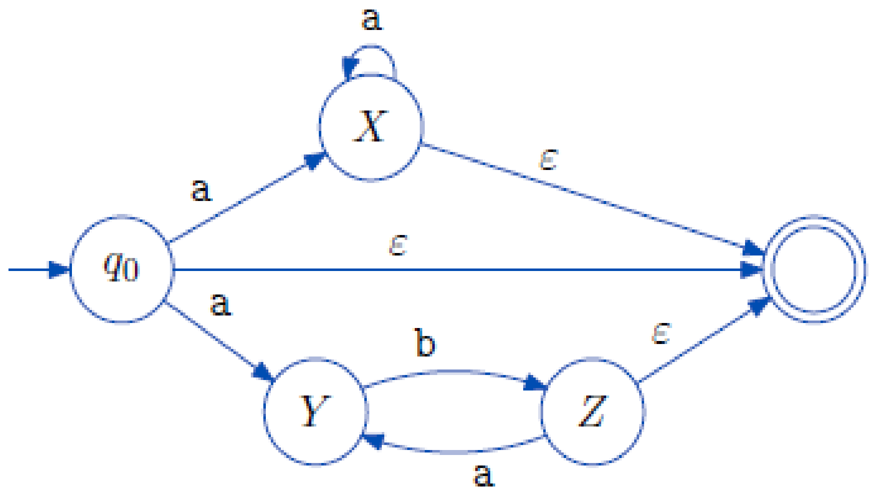
# Problem-3: REs and FAs (1)

An NFA for $a^* + (ab)^*$



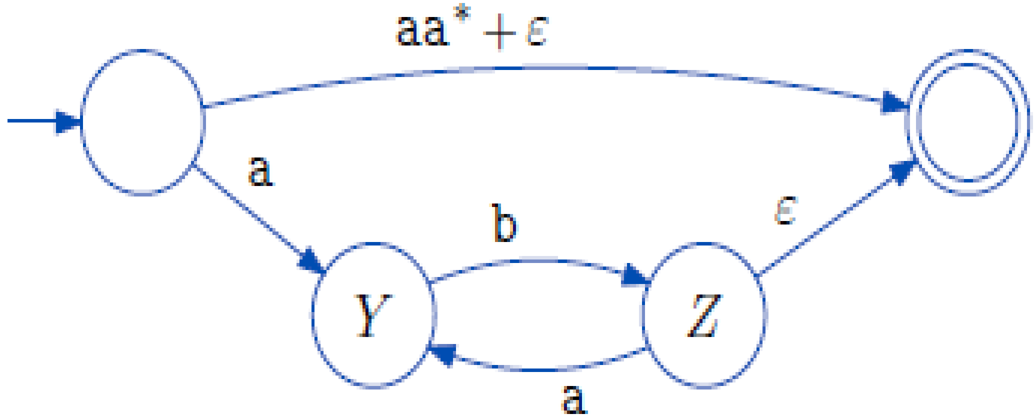
# Problem-3: REs and FAs (2)

Here is the NFA for $a^* + (ab)^*$ adjusted to have a unique accept state.



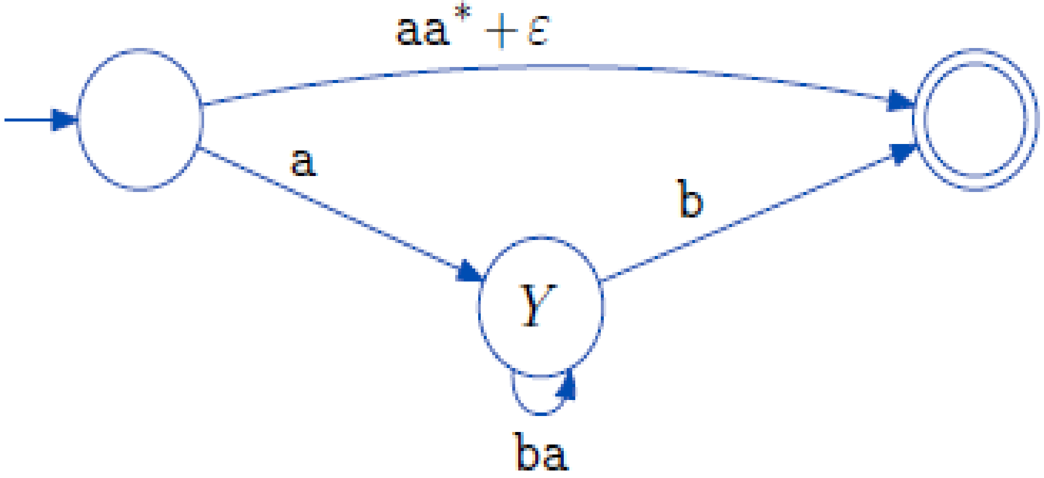
# Problem-3: REs and FAs (3)

If we eliminate state $X$ we get



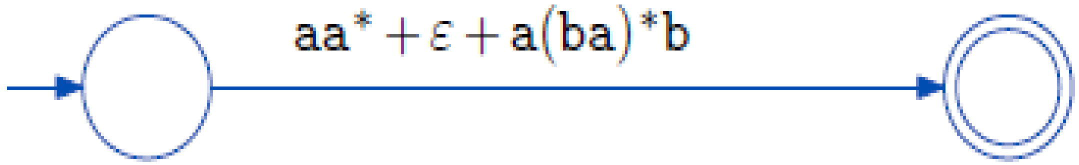
# Problem-3: REs and FAs (4)

If we eliminate state $Z$ we get



# Problem-3: REs and FAs (5)

If we eliminate state $Y$ we get



# REFERENCES


- “Introduction to Automata Theory, Languages, and Computation“
  by J. E. Hopcroft, R. Motwani, and J. D. Ullman

- “Introducing The Theory Of Computation”
  by W. Goddard.