Problem vs. Languages

1. \[ a + b = ? \]

2. Given \( w \) and a DFA \( M \), can \( M \) accept \( w \)?

3. Given \( w \) and a NFA \( M \), can \( M \) accept \( w \)?

4. Given DFA \( M \), is \( L(M) = \emptyset \)?

Algorithm vs. Turing Machines

A1: compute \( a + b \).

A2: test if \( q_0 \) reaches \( q_f \)
via path/computation
that consumes \( w \).

A3: transform NFA to DFA

A4: reachability test from \( q_0 \) to \( q_f \).
Problem

Languages

\[ L \]

\[ \forall w \in \Sigma^* \exists M. \text{ if } w \in L \text{ then } M \text{ either accepts or rejects } w \]

undecidable

("undecidable")

"recognizable" maybe

("recognizable")

"decidable"
Problem # 2:

\[ L_{DFA} = \{ <B, w> | B \text{ is a DFA that accepts } w, w \in \Sigma^* \} \]

Theorem: \( L_{DFA} \) is decidable.

- Encoding of a DFA \( <X_1, X_2 \ldots X_m> \):
  - \( <Q, \Sigma, Q_0, Q_0, F> \)
  - \( Q_0, q_1, \ldots q_n > \)
  - \( q_a \) : accept (\( q_{a1}, \ldots q_{an} \)) : find.
  - \( S : (q_i, X_j) \rightarrow q_k \)
    - \( (q_1, X_1) \rightarrow q_4 \)
    - \( 00 \ 01 \ 0000 \)

- "Algorithm".

M : TM

Read/move \( W \)

B \quad W
Problem #3 "emptiness"

\[ E_{DFA} = \{ <A> \mid A \text{ is a DFA and } L(A) = \emptyset \} \]

Theorem: \( E_{DFA} \) is decidable.

DFA:

\[
\begin{array}{c}
\text{0} \\
\longrightarrow \\
\text{1} \\
\text{2} \\
\text{3} \\
\text{4} \\
\end{array}
\]

\[
\{ (1, 2, 3, 4), (1, 2), (2, 3), (3, 1), (3, 4) \}
\]
Recursive languages: closure properties

Let language \( L \) be a recursive language (decidable).

Then:

\[ L^c \]

\[ \overline{L} \]

\[ \emptyset \]

\[ \emptyset \]

\[ \ast \]

\[ \ast \]

\[ \{ \} \]

\[ \{ \} \]

\[ \leq \]

\[ \leq \]

\[ L \]

\[ L \]

\[ M \]

\[ M \]

---

\[ M_1 \]

\[ M_1 \]

\[ M_2 \]

\[ M_2 \]

---

OR

---

Accept

---

Reject

---

Accept

---

Reject

---

Reject

---