Decision Procedures for Regular Languages
Decidable Questions

1. Set Membership: \( w \in L(M) \)?
   - Run \( M \) on \( w \)

2. Emptiness: \( L(M) = \emptyset \)?
   - Find minimal machine \( Min(M) \) corresponding to \( M \).
     \( L(M) = \emptyset \iff Min(M) \) has only one (nonaccepting) state
   - Test \( M \) on all strings of length < number of states of \( M \)
     \( L(M) = \emptyset \iff M \) does not accept any such string (Pumping Theorem)

3. Totality: \( L(M) = \Sigma^* \)?
   - \( L(M) = \Sigma^* \iff L(M)^c = \emptyset \)
More Decidable Questions

4. **Finiteness:** $|L(M)| < \infty$?

   - Find minimal machine $\text{Min}(M)$ corresponding to $M$.
     $|L(M)| < \infty \iff \text{Min}(M)$ has no Loops

   - Test $M$ on all strings $w$, where 
     \[ \text{number of states} \leq \text{length}(w) < 2(\text{number of states}) - 1 \]  
     (Pumping Thm)

5. **Equivalence:** $L(M_1) = L(M_2)$?

   - Find minimal machines $\text{Min}(M_1)$ and $\text{Min}(M_1)$
     $L(M_1) = L(M_2) \iff \text{Min}(M_1) = \text{Min}(M_2)$

   - $L(M_1) = L(M_2) \iff (L(M_1) - L(M_2)) \cup (L(M_2) - L(M_1)) = \emptyset$
Finite State Automata vs. Turing Machines

Observation

- Such Decision Procedures do NOT Exist for Turing Machines -- Later
  -- Finite State Automata -- Bounded (Finite) Storage
  -- Turing Machines -- Unbounded (Infinite) Storage