Theory of Computation
Chapter 1

Regular Languages Part 1

School of Engineering | Computer Science

Claude Shannon
1916-2001

- Wrote "the most important Master's thesis of all time", proposing that a machine based on Boolean logic could solve problems.
- Later, founder of "information theory" – theory of how to compress, encode and store information.
- Named the "bit"

Regular Languages

- A is a regular language if there is a finite automaton, DFA, M, that recognizes it $L(M) = A$
- Regular languages are the set of languages that are recognizable by deterministic finite automata

Closure of Regular Languages

- Given two regular languages A and B, which operations applied to A and B produce a new regular language C?
- There are three regular operations:
  - Union: $A \cup B = \{ x \mid x \in A \text{ or } x \in B\}$
  - Concatenation: $A \circ B = \{xy \mid x \in A \text{ and } y \in B\}$
  - Star: $A^* = \{x_1x_2\ldots x_k \mid k \geq 0 \text{ and } x_i \in A \text{ for all } 1 \leq i \leq k\}$ ($k = 0$ is allowed → empty string, ε, which is always in A^*)
- Regular operations are helpful and allow us to break up complex languages into smaller, simpler pieces and to combine them using regular operations

Closure of Regular Languages, cont.

- There are three regular operations:
  - Union: $A \cup B = \{ x \mid x \in A \text{ or } x \in B\}$
  - Concatenation: $A \circ B = \{xy \mid x \in A \text{ and } y \in B\}$
  - Star: $A^* = \{x_1x_2\ldots x_k \mid k \geq 0 \text{ and } x_i \in A \text{ for all } 1 \leq i \leq k\}$ (k = 0 is allowed → empty string, ε, which is always in A^*)
- Ex: $S = \{a, b\}$, $T = \{0, 1\}$, $\Sigma = \{0, 1, a, b\}$
  - $S \cup T =$
  - $S \circ T =$
  - $S^* =$

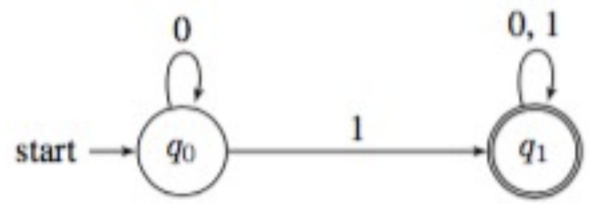
Closure of Regular Languages, cont.

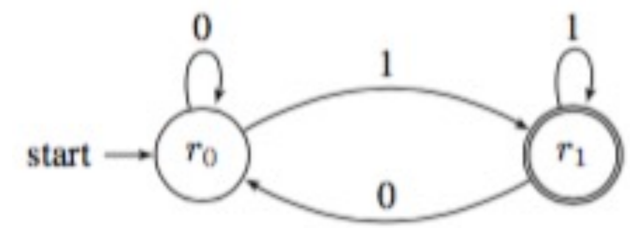
- There are three regular operations:
  - <u>Union</u>: $A \cup B = \{ x \mid x \in A \text{ or } x \in B\}$
  - <u>Concatenation</u>: $A \circ B = \{xy \mid x \in A \text{ and } y \in B\}$
  - <u>Star</u>: $A^* = \{x_1x_2\ldots x_k \mid k \geq 0 \text{ and } x_i \in A \text{ for all } 1 \leq i \leq k\}$ ($k = 0$ is allowed → empty string, ε which is always in A^*)
- Ex: $S = \{a, b\}$, $T = \{0, 1\}$, $\Sigma = \{0, 1, a, b\}$
  - $S \cup T = \{a, b, 0, 1\}$
  - $S \circ T = \{a0, a1, b0, b1\}$
  - $S^* = \{a, b\}^* = \{\varepsilon, a, b, aa, bb, ba, bb, aaa, \ldots \}$

Closure Definition

- Closure:
  - Are regular languages closed under their operations?
  - What is closure?
    - If a class of languages is closed under an operation, then, when that operation is performed on languages in that class, the result is always a language within that class of languages.
      - Ex: Adding two integer numbers, the result is an integer value → integers are closed under addition
      - Ex: Dividing two integer numbers, the result is not always an integer → integers are not closed under division

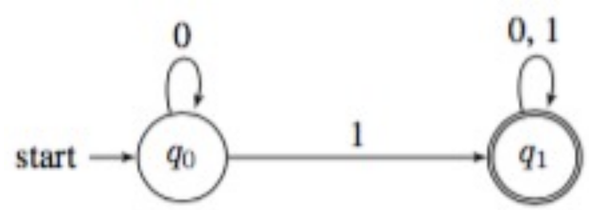
Closure of Regular Languages Definition

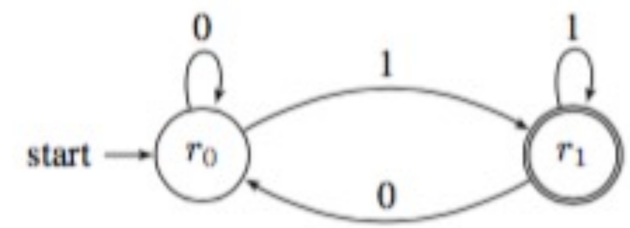
- Are regular languages closed under the union operation?
    - Proof Ex: M_1: 

      M_2: 

    - M_3 recognizes $L(M_1) \cup L(M_2)$:
        - Combine the states in every combination
        - Show the transitions simultaneously

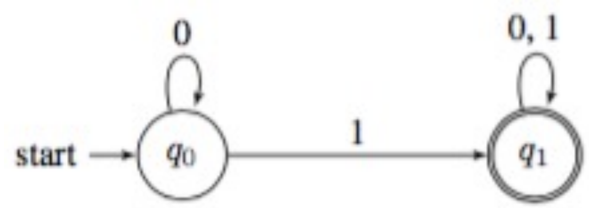
Closure of Regular Languages Definition

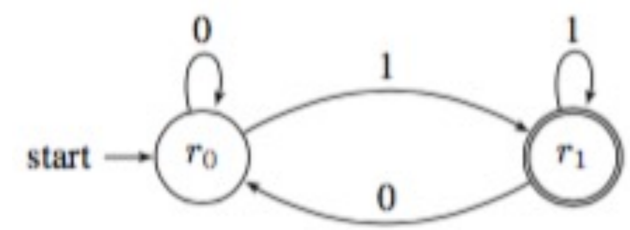
- Are regular languages closed under the union operation?
  - Proof Ex: M_1: 

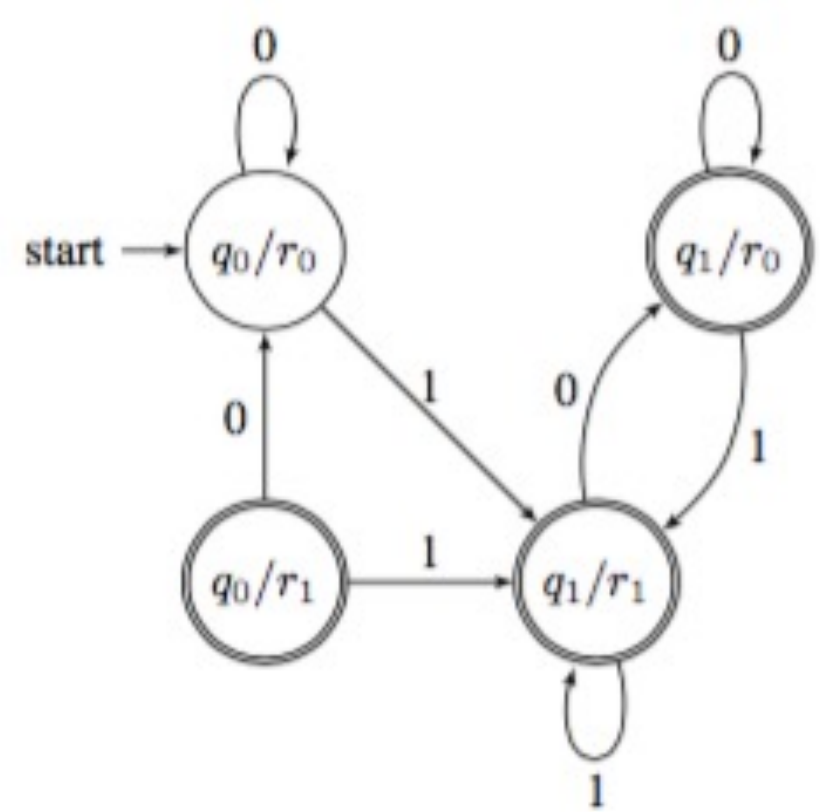
    M_2: 

  - M_3 recognizes $L(M_1) \cup L(M_2)$:

Closure of Regular Languages Definition

- Are regular languages closed under the union operation?
  - Proof Ex: M_1: 

  M_2: 

  - M_3 recognizes $L(M_1) \cup L(M_2)$: 

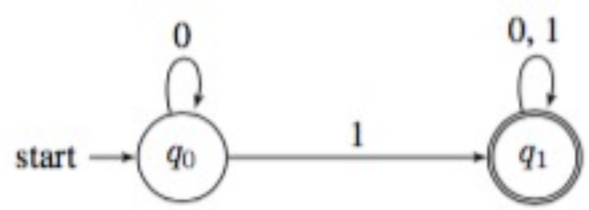
Closure of Regular Languages Definition, cont.

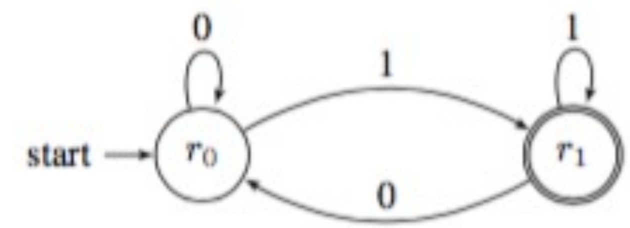
- Are regular languages closed under the union operation?
  - Put the idea into words:
    - Let Q_1 and Q_2 be sets of states for the finite automata M_1 and M_2 recognizing the languages A and B, respectively.

1. States for $M_1 \cup M_2$ are $Q_1 \times Q_2$
2. Simultaneously track the transitions for both M_1 and M_2
3. Accept only if string ends in state (q_i, r_j) such that either q_i or r_j is an accepting state in M_1 or M_2 respectively

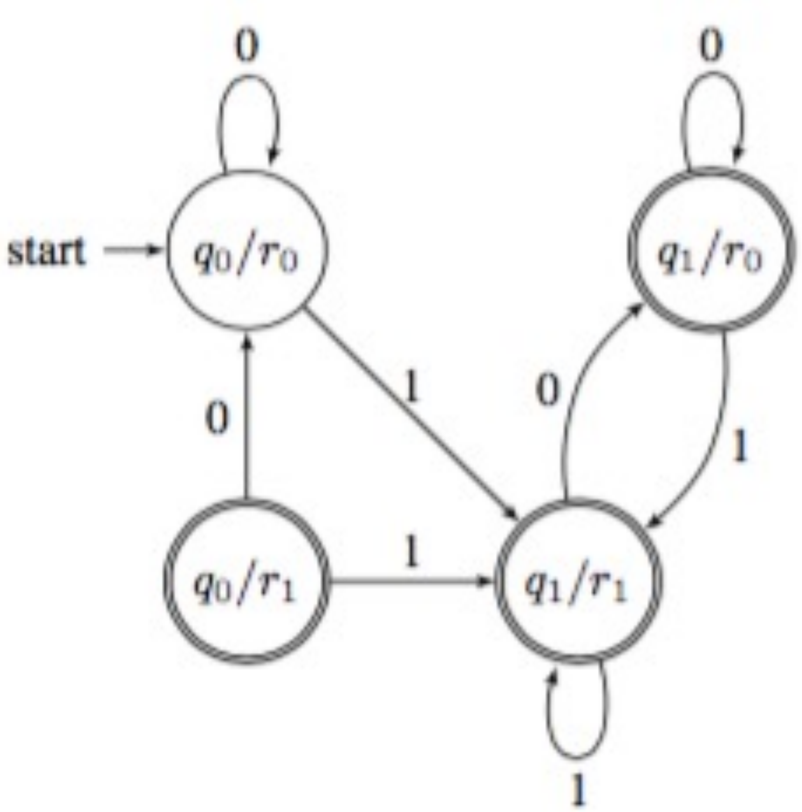
School of Engineering | Computer Science

Closure of Regular Languages Definition

- Are regular languages closed under the union operation?
  - Proof Ex: M_1: 

M_2: 

  - M_3 recognizes $L(M_1) \cup L(M_2)$:



School of Engineering | Computer Science

Closure of Regular Languages Definition, cont.

- Formal Proof that regular languages are closed under the union operation
    - Let M_1 recognize A_1 where $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and M_2 recognize A_2 where $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$, construct $M_3 = (Q_3, \Sigma, \delta_3, q_3, F_3)$ such that $L(M_3) = A_1 \cup A_2$ (M_3 recognizes $A_1 \cup A_2$)

1. $Q_3 = Q_1 \times Q_2 = \{(r_1, r_2) \mid r_1 \in Q_1, r_2 \in Q_2\}$
2. Σ is the same
3. For all $(r_1, r_2) \in Q_3$, and all $a \in \Sigma$, let $\delta_3((r_1, r_2), a) = (\delta_1(r_1, a), \delta_2(r_2, a))$
4. $q_3 = (q_1, q_2)$
5. $F_3 = \{(r_1, r_2) \mid r_1 \in F_1 \text{ or } r_2 \in F_2\}$ (same as $F_3 = (F_1 \times Q_2) \cup (Q_1 \times F_2)$)

Summary Closure of Regular Languages

- We just showed that we can formally prove that regular languages are closed under the union operation.
- The same is true for applying the concatenation and star operations, but the proofs will be easier once we study non-deterministic finite automata (NFAs).
- We will learn about NFAs soon.