Regular Languages

- **Closure properties:** Regularity preserved under union, choice, concatenation.
  - Kleene Closure (by definition):
    - \( K_1, K_2 \text{ regular} \Rightarrow K_1 \cap K_2 \text{ regular} \) (consider \( G_1, G_2 \))
    - \( K \text{ regular} \Rightarrow K^c \text{ regular} \) (consider \( (G)^c \))

- **Pumping Lemma:** \( K \text{ regular} \Rightarrow \exists m \text{ such that } \forall w \in K, |w| \geq m, w = uv^iww \in K \) for each \( i \).

**Proof:** DFSA \( G \) s.t. \( L(G) = K \). Set \( m = 1 \times 1 \).
- Pick \( s \in K \) with \( |s| \geq m \).
- Let \( t \leq s \) be such that \( |t| = m \).
- \( s \) is execution of \( t \) visits at least one state twice. \( t \) be 1st such state.

- **Application of Pumping Lemma:** \( K = \{a^i b^j | i \geq j \} \) is not regular.

**Proof:** Suppose for contradiction \( K \) is regular. Pick \( s = a^m b^m \in K \), where \( m \) as in pumping lemma.
- So \( s = a^m b^m = uvw \) with \( |w| \geq 1 \), \( u \), \( v \), \( w \in K \)
- \( |uv| \leq m \Rightarrow u = a^i \), \( v = a^k \), \( j + k \leq m \), \( k \geq 1 \)
  - So \( w = a^{m-(j+k)} b^m \)
- Choose \( i = 2m \). Then \( uv^i w = a^j a^{2km} a^m b^m = a^{-2km-m-k} b^m \).
- Since \( m - k \geq 2 \) & \( k \geq 1 \), \( 2km + m - k \geq 2m \), a contradiction.
Review of equivalence relation

Given a set \( X \), a relation \( R \) on \( X \) is a subset \( R \subseteq X \times X \)

\((a, y) \in R\), then \( a \) is related to \( y \), written \( a \, R \, y \)

Example: \( X = \mathbb{N} \), set of naturals, \( a \, R \, y \) if \( a \mod 5 = y \mod 5 \)

\( 4R9, \ 9R14, \ 2R0 \)

\( R \) is equivalence relation if

i) reflexive: \( a \, R \, a \)

ii) symmetric: \( a \, R \, y \Rightarrow y \, R \, a \)

iii) transitive: \( a \, R \, y, y \, R \, z \Rightarrow a \, R \, z \)

Examples: \( \mod 5 \) relation is an equivalence

"brother" relation is transitive but not symmetric

An equivalence relation \( R \) denoted by \( \equiv_R \).

Equivalence class or coset \([a]_R \subseteq X \) if \( x \in [a]_R \) if \( x \equiv_R a \)

Example: \([4]_{\mod 5} = \{ 4+5k \mid k \in \mathbb{N} \} = \{ 4, 9, 14, 19, 24, \ldots \} \)

Then: The set of all equivalence classes \( \{ [x]_R \mid x \in X \} \) form a partition of \( X \)

\( X = \bigcup [a]_R = X \) covers \( X \) (obvious)

\( a \in X \)

(i) \([a]_R \neq [b]_R \Rightarrow [a]_R \cap [b]_R = \emptyset \) different \( \Rightarrow \) pairwise disjoint

Suppose for contradiction, \([a]_R \cap [b]_R \neq \emptyset \).

Pick \( x \in [a]_R \cap [b]_R \) \( \Rightarrow a \equiv_R x \) \( \land \) \( b \equiv_R x \) \( \Rightarrow a \equiv_R b \) \( \Rightarrow [a]_R = [b]_R \).

Example:

\([0]_{\mod 3} = \{ 0, 3, 6, 9, \ldots \} \) \( \Rightarrow \) pairwise disjoint

\([1]_{\mod 3} = \{ 1, 4, 7, 10, \ldots \} \)

\([2]_{\mod 3} = \{ 2, 5, 8, 11, \ldots \} \)

\([0]_{\mod 3} \cup [1]_{\mod 3} \cup [2]_{\mod 3} = \mathbb{N} \) \( \Rightarrow \) covers \( \mathbb{N} \)

Index of \( \equiv_R \) is \( n \). \( n \) distinct equivalence classes of \( \equiv_R \).
Myhill-Nerode Characterization

- Another characterization of regular languages

- Equivalence relation on $\Sigma^*$ induced by $K$:
  \[ a \cong t (R_K) \iff [K \setminus a^* t = K \setminus \epsilon] \]
  \[ \iff [\forall u \in \Sigma^* : a u \not\in K \iff t u \in K] \]

  Note: $a \cong t (R_K) \implies \forall u : a u \not\in t u (R_K)$
  $R_K$ is "right invariant" (wrt concatenation)

- Equivalence relation on $\Sigma^*$ induced by $G$:
  \[ a \cong t (R_G) \iff [\exists (x_0, t) \in \Sigma^* \times \Sigma^* : x (x_0, t) \in \Delta] \lor [\Delta \times \Sigma^* \text{ undefined}] \]

  Note: $a \cong t (R_G) \implies [b \cong t (R_L (a))] \land [a \cong t (R_L (a))]$
  $R_G$ "refines" $R_L (a)$ and $R_L (a)$

- Example: $K = (ad)^*$
  \[ [a] (R_K) = (ad)^* \]
  \[ [a] (R_K) = a (da)^* \]
  \[ [a] (R_K) = \Sigma^* - (a (da)^*) \]

(Example continued on another page)
Thm: The following are equivalent:

1. $K$ is regular
2. $K$ is union of some equivalence classes of a right-invariant equivalence relation of finite index
3. $R_k$ is finite index

Pf: ($1 \Rightarrow 2$) Let $G$ be a DFSM with $Lm(G) = K$.

$$K = \{ [s](R_a) \mid s \in K \} \Rightarrow R_a \text{ is the required equivalence relation}$$

($2 \Rightarrow 3$) Suppose $R$ is the given equivalence relation.

It suffices to show that $R$ refines $R_k$, i.e., $s \equiv_t (R) \Rightarrow s \equiv_t (R_k)$

$$s \equiv_t (R) \Rightarrow \exists u: su \equiv tu (R)$$

$$\exists u \in u \in K \Rightarrow tu \in K$$

$$\Rightarrow s \equiv_t (R_k)$$

($3 \Rightarrow 1$) Construct a DFSM $G$:

$$X = \{ [s](R_k) \mid s \in \Sigma^* \}$$

$$X_m = \{ [s](R_k) \mid s \in K \}$$

$$x_0 = [\varepsilon](R_k)$$

$$\alpha([s](R_k), \varepsilon) = [as](R_k)$$

This implies: 

$\text{# of states in minimal DFSM } \leq |R_k| - 1$

Example: $K = (ad)^*$

[2] $[e](R_w) = (ad)^*$

[5] $[a](R_k) = a(ad)^*$

[4] $[d](R_k) = \Sigma^* - pr(K)$. 

Remark: 

$\text{# of states in minimal DFSM for } K = |R_k| - 1$. 

($\text{Since } R_a \text{ refines } R_{Lm(G)} \Rightarrow \text{# of states in } G \geq |R_k| \Rightarrow \text{# of states in minimal DFSM } \geq |R_k| - 1$)
**Algorithm for Regular languages.**

**Emptiness:** \( K = \emptyset \)? Construct DFSA \( G \) such that \( L(G) = K \).

\[ [K = \emptyset ] \iff [R_G(x_0) \land \neg m = \emptyset ] , \text{ where} \]
\( R_G(x_0) = \text{set of reachable states from } x_0. \)

**Containment:** \([K_1 \subseteq K_2 \iff K_1 \cap K_2^c = \emptyset \).\]

---

**Algorithm for Computing Reachability:**

1. **Initialization Step:** \( R_G^0(x_0) = \emptyset ; \ R_G^0(x_0) = \{ x_0 \}; \ k = 0 \)

2. **Iteration Step:**

\( R_G^{k+1}(x_0) = R_G^k(x_0) \cup \left\{ x \in X - R_G^k(x_0) \mid \exists x' \in R_G^k(x_0) - R_G^k(x_0), r \in \Sigma : \right\}
\( \alpha(x', r) = x \}

3. **Termination step:** \( R_G^{k+1}(x_0) = R_G^k(x_0) \), then \( R_G(x_0) = R_G^k(x_0) \), stop;

else \( k = k + 1 \), goto step 2.

**Complexity:** \( G \) has \( m \) states \( \Rightarrow O(m) \) steps to compute \( R_G(x_0) \).

\( f = O(g) \iff f(n) \leq Cg(n) \)

**Definitions:** \( G \) is called accessible if \( R_G(x_0) = X \)

\( G \) is called co-accessible if \( L_G(x) \land \neg m = \emptyset \) \( \forall x \in X \).

\( \text{trim} = \text{accessible} + \text{co-accessible}. \)

- It is always possible to find a lang-equivalent trim SM for \( G \).

- minimal SM must be trim.
Algorithm for SM: minimization.

- Follows from Myhill Nerode Characterization that a trim DFSM is minimal iff
  \[ \forall \delta: (x, x', y) \in X \times X \times \Sigma \Rightarrow [x = x'] \]

- Aggregate \( x \) and \( x' \) if they are unequal.

- Consider a trim DFSM \( G \) with \( L_m(c) \neq \Sigma^* \) (otherwise minimal is trivial).

  Construct \( \bar{G} = (\bar{X}, \bar{I}, \bar{A}, \bar{X}_0, \bar{X}_m) \)

  Then a pair of states \( (\bar{x}, \bar{x'}) \in \bar{X}_m \times (\bar{X} - \bar{X}_m) \) must not be aggregated.

Initialization:
- \( A_0 := (X \times X) \cup (\bar{X} - \bar{X}_m) \times (\bar{X} - \bar{X}_m) \)
- \( B_0 := (\bar{X} \times \bar{X}) - A_0 \) pairs which cannot be aggregated
- \( k := 0 \)

Iteration:
- \( A_{k+1} := A_k \setminus \{ (x, x') \mid \exists \sigma \text{ s.t. } (x(x, \sigma), x'(x', \sigma)) \in B_k \} \)
- \( B_{k+1} := (\bar{X} \times \bar{X}) - A_{k+1} \)

Termination:
- Stop if \( A_{k+1} = A_k \); else \( k := k+1 \) and iterate.
"Button tie-down problem"

- Safety feature provided in potentially hazardous machines (saw mill, die press) by requiring machine be only allowed to start when both hands are pressing a push-button (can't start machine by one hand)

\[\text{Step 1:} \quad \text{p}_1 = \text{push; } \quad \text{r}_i = \text{release; } \quad \text{u} = \text{"up"; } \quad \text{d} = \text{"down"} \]

- Spec: Successive start events must be blocked unless separated by execution of both the push-button events (can't start the machine by tapping down one button, and pressing the other thus by using only one hand)

Source: "Basic Control Systems Eng.," Prentice Hall, 1997