Inductive definitions and Induction

- $\mathbb{N}$ = natural numbers = $\{0, 1, 2, \ldots\}$
- Inductive definition:
  i. $0 \in \mathbb{N}$
  ii. $\forall n \in \mathbb{N},\ n+1 \in \mathbb{N}$
- To prove: $\forall n \in \mathbb{N},\ P(n)$
- Base case: Prove $P(0)$
- (Weak) Inductive Hypothesis
  - Assume $P(n)$
- Ind. step: Prove $P(n+1)$ using IH

- A^* = strings over a non-empty finite set A

alphabet

- Inductive definition:
  i. $\varepsilon \in A^*$
  ii. $\forall x \in A^*, \forall a \in A,\ x.a \in A^*$
- To prove: $\forall x \in A^*,\ P(x)$
- Base case: Prove $P(\varepsilon)$
- (Weak) Inductive Hypothesis
  - Assume $P(x)$
- Ind. step: Prove $\forall a \in A$, $P(x.a)$ using IH

Example 1

- For any alphabet A, prove that $\forall a \in A, \forall x \in A^*,\ a.x \in A^*$
- Proof (induction on x):
  - Base case ($x = \varepsilon$): $a.\varepsilon = a = \varepsilon.a$ [property of ε]
    $\in A^*$
  - Inductive step ($x = y.b$ for some $y \in A^*$, $b \in A$):
    $a.(y.b) = (a.y).b$ [property of .]
    By IH, $a.y \in A^*$
    Hence, $(a.y).b \in A^*$ [by definition of A^*]

Example 2

- Let $A = \{0, 1\}$ and inductively define $f : A^* \to \mathbb{N}$ as:
  i. $f(\varepsilon) = 0$
  ii. $\forall x \in A^*,\ f(x.0) = 2f(x) + 1$
  iii. $\forall x \in A^*,\ f(x.1) = 2f(x) + 2$

- Prove that $\forall n \in \mathbb{N},\ f(0^n) = 2^n - 1$

- Prove that $\forall n \in \mathbb{N}, \exists x \in A^*,\ f(x) = n$

Try yourself: Prove that f is injective.
(Hence, $\{0, 1\}^*$ is countably infinite)

Languages

- A language L over the alphabet A is a subset of A^*
  - $\varnothing \subseteq L \subseteq A^*$

- **Claim**: The number of languages over A is uncountably infinite

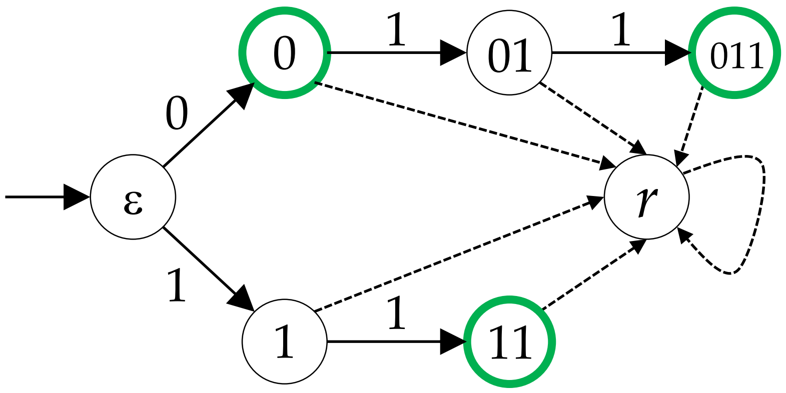
- **Proof**: Suppose not. Then we can enumerate all languages $L_0, L_1, L_2, \ldots$ Also, we can enumerate all strings $x_0, x_1, x_2, \ldots$
- Define $L_d = \{x_i | x_i \notin L_i\}$
- Then, $\forall i \in \mathbb{N},\ L_d \neq L_i$, a contradiction

Diagonalization argument

Deterministic Finite Automata (DFA)

- *Example*: Let $A = \{0, 1\}$ and let $L = \{0, 11, 011\}$

- **Question**: Is there a DFA with fewer states that accepts L?



A Deterministic Finite Automaton (DFA) over alphabet A is a tuple $M = (Q, s, \delta, F)$ where:

- $Q \neq \emptyset$ is a **finite** set of states, $s \in Q$, $F \subseteq Q$
- $\delta : Q \times A \rightarrow Q$

Question: Define $\hat{\delta}: Q \times A^* \rightarrow Q$ inductively

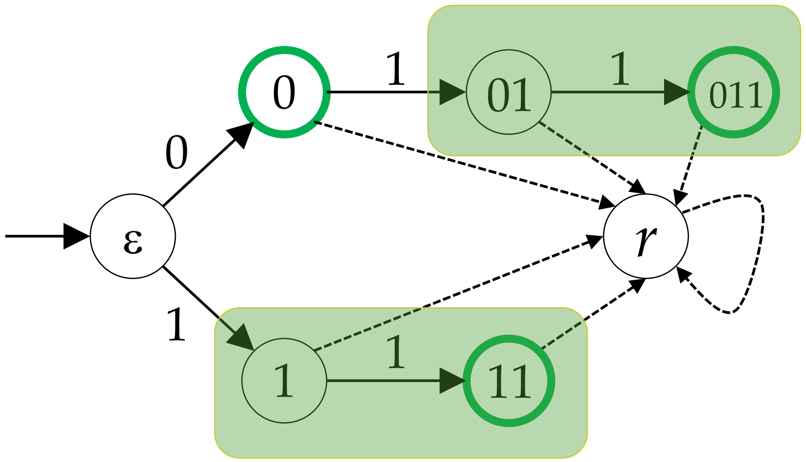
Definition: $L(M) = \{x \in A^* | \hat{\delta}(s, x) \in F\}$

Definition: L is regular if $\exists$DFA M s.t. $L = L(M)$

Deterministic Finite Automata (DFA)

- *Example*: Let $A = \{0, 1\}$ and let $L = \{0, 11, 011\}$

- **Question**: Is there a DFA with fewer states that accepts L?



A Deterministic Finite Automaton (DFA) over alphabet A is a tuple $M = (Q, s, \delta, F)$ where:

- $Q \neq \emptyset$ is a **finite** set of states, $s \in Q$, $F \subseteq Q$
- $\delta : Q \times A \to Q$

Question: Define $\hat{\delta}: Q \times A^* \to Q$ inductively

Definition: $L(M) = \{x \in A^* | \hat{\delta}(s, x) \in F\}$

Definition: L is regular if $\exists$DFA M s.t. $L = L(M)$

All finite languages are regular

- **Idea**: Generalize the construction in the example. Define

$$pre(L) = \{x \in A^* | \exists y \in A^*, x.y \in L\}$$

- **Claim**: If L is finite then $pre(L)$ is finite. Short proof?
- Let $Q = pre(L) \cup \{r\}$, let $F = L$ and $\forall a \in A$, let

$$\delta(r, a) = r$$

$$\forall x \in pre(L), \qquad \delta(x, a) = \begin{cases} x.a & \text{if } x.a \in pre(L) \\ r & \text{otherwise} \end{cases}$$

- What about s?
- **Main claim**: $\forall x \in A^*,\ x \in L \Leftrightarrow \hat{\delta}(s, x) \in F$

Observation 1:
$L \subseteq pre(L)$

Observation 2:
$L = \emptyset \Rightarrow pre(L) = \emptyset$

Even if L is not finite, this defines a deterministic automaton (DA), called the "free DA" for L

Lemma

$$\forall x \in A^*, \qquad \hat{\delta}(s,x) = \begin{cases} x & \text{if } x \in pre(L) \\ r & \text{otherwise} \end{cases}$$

- **Base case** ($x = \varepsilon$): True by definition of $\hat{\delta}$ and s
- **Inductive step** ($x = y.a$ for some $y \in A^*$ and some $a \in A$)
  - *Case 1* ($y \in pre(L)$). By IH, $\hat{\delta}(s,y) = y$. Now apply definition of δ.
  - *Case 2* ($y \notin pre(L)$). By IH, $\hat{\delta}(s,y) = r$. Now apply definition of δ.
- Back to main claim: $\forall x \in A^*,\ x \in L \Leftrightarrow \hat{\delta}(s,x) \in F$
  - If $x \in pre(L)$ then $\hat{\delta}(s,x) = x$ and hence $x \in L \Leftrightarrow \hat{\delta}(s,x) \in F$
  - If $x \notin pre(L)$ then $\hat{\delta}(s,x) = r$ and both LHS and RHS are false

Not all regular languages are finite

- *Examples*: Strings over $\{a, b\}$ that: contain an odd number of a's, contain the substring *abb*, (at least one property/both/exactly one/neither), …

- For any language $L \subseteq A^*$, define the following relation over A^*

$$x \equiv_L y \quad \text{iff} \quad \forall z \in A^*, \qquad x.z \in L \Leftrightarrow y.z \in L$$

- **Claim**: $\equiv_L$ is an equivalence relation

Example: For $L = \{0, 11, 011\}$,
$01 \equiv_L 1,\ 011 \equiv_L 11$ and $0 \not\equiv_L 11$

- **Theorem** (*Myhill-Nerode*): L is regular iff $\equiv_L$ has finite index

Regular $\Rightarrow$ finite index

- Suppose L is regular i.e., $\exists M = (Q, s, \delta, F)$ such that $L = L(M)$

- **Claim 1**: This is an equivalence relation over A^*

$$x \equiv_M y \quad \text{iff} \quad \hat{\delta}(s, x) = \hat{\delta}(s, y)$$

- **Claim 2**: The index of $\equiv_M$ is at most $|Q|$
- **Claim 3**: $\equiv_M$ refines $\equiv_L$ (and hence the index of $\equiv_L$ is at most $|Q|$)
  - *Need to show*: If $\hat{\delta}(s, x) = \hat{\delta}(s, y)$ then $\forall z \in A^*,\ x.z \in L \Leftrightarrow y.z \in L$
  - Instead, show: If $\hat{\delta}(s, x) = \hat{\delta}(s, y)$ then $\forall z \in A^*,\ \hat{\delta}(s, x.z) = \hat{\delta}(s, y.z)$

Finite index ⇒ regular

- Define the DFA $M^* = (Q, s, \delta, F)$ where
  $Q = \{[x] \mid x \in A^*\}$, $s = [\varepsilon]$, $F = \{[x] \mid x \in L\}$
  and $\forall [x] \in Q, \forall a \in A,\ \delta([x], a) = [x.a]$

Note: $[x]$ denotes the equivalence class of x for the relation $\equiv_L$

- **Claim 4**: δ is well-defined i.e., $\forall x, y \in A^*$,
  $$x \equiv_L y \Rightarrow \forall a \in A, \qquad x.a \equiv_L y.a$$
- **Claim 5**: $\forall x \in A^*,\ \hat{\delta}(s, x) = [x]$
  - From this, it follows that $L(M^*) = L$ and hence L is regular
- *Note*: From Claim 3, it follows that M^* is a smallest DFA for L

Applications of Myhill-Nerode Thm.

- Let L be any language over alphabet A and let $S \subseteq A^*$ such that
$$\forall x, y \in S, \qquad x \equiv_L y \Rightarrow x = y$$

- **Application 1**: If L is regular, then any DFA for L has at least $|S|$ states
  - *Example*: Let $A = \{a, b\}$ and let L_k be the set of strings whose k^{th} last letter is b. Then any DFA for L_k has at least 2^k states.
- **Application 2**: If $|S|$ is infinite, then L is not regular
  - *Example*: Let $A = \{a, b\}$ and let L be the set of strings with an unequal number of a's and b's. Then L is not regular.

Cross-product construction

- Let $M_1 = (Q_1, s_1, \delta_1, F_1)$ and $M_2 = (Q_2, s_2, \delta_2, F_2)$ be two DFAs
- $\forall F \subseteq Q_1 \times Q_2$, define $M_1 \times M_2(F) = (Q_1 \times Q_2, (s_1, s_2), \delta_1 \times \delta_2, F)$ where
  $\forall (q_1, q_2) \in Q_1 \times Q_2, \forall a \in A, \qquad \delta_1 \times \delta_2\big((q_1, q_2), a\big) = \big(\delta_1(q_1, a), \delta_2(q_2, a)\big)$
- **Claim**: $\forall x \in A^*, \widehat{\delta_1 \times \delta_2}\big((s_1, s_2), x\big) = \Big(\widehat{\delta_1}(s_1, x), \widehat{\delta_2}(s_2, x)\Big)$
- **Consequence 1**: $L(M_1 \times M_2(F_1 \times F_2)) = L(M_1) \cap L(M_2)$, and hence regular languages are closed under intersection
- **Claim**: For $M = (Q, s, \delta, F)$ let $\bar{M} = (Q, s, \delta, Q - F)$. Then $L(\bar{M}) = \overline{L(M)}$ and hence regular languages are closed under complement
- **Consequence 2**: Regular languages are closed under union

Cross-product construction

- Let $M_1 = (Q_1, s_1, \delta_1, F_1)$ and $M_2 = (Q_2, s_2, \delta_2, F_2)$ be two DFAs
- $\forall F \subseteq Q_1 \times Q_2$, define $M_1 \times M_2(F) = (Q_1 \times Q_2, (s_1, s_2), \delta_1 \times \delta_2, F)$ where
  $\forall (q_1, q_2) \in Q_1 \times Q_2, \forall a \in A, \qquad \delta_1 \times \delta_2\big((q_1, q_2), a\big) = \big(\delta_1(q_1, a), \delta_2(q_2, a)\big)$
- **Claim**: $\forall x \in A^*, \widehat{\delta_1 \times \delta_2}\big((s_1, s_2), x\big) = \Big(\widehat{\delta_1}(s_1, x), \widehat{\delta_2}(s_2, x)\Big)$
- **Consequence 1**: $L(M_1 \times M_2(F_1 \times F_2)) = L(M_1) \cap L(M_2)$, and hence regular languages are closed under intersection
- **Claim**: For $M = (Q, s, \delta, F)$ let $\bar{M} = (Q, s, \delta, Q - F)$. Then $L(\bar{M}) = \overline{L(M)}$ and hence regular languages are closed under complement
- **Consequence 2**: Regular languages are closed under union

Direct proof: Choose $F = F_1 \times Q_2 \cup Q_1 \times F_2$

Non-deterministic FA (NFA)

- Replace the transition function $\delta : Q \times A \rightarrow Q$ by either

i. A transition function $\Delta : Q \times (A \cup \{\varepsilon\}) \rightarrow 2^Q$ (power set of Q)

  - If $q \in \Delta(p, e)$ then the automaton can go from p to q on input e

ii. A transition relation $\Delta \subseteq Q \times (A \cup \{\varepsilon\}) \times Q$

  - If $(p, e, q) \in \Delta$ then the automaton can go from p to q on input e

- Inductive definition of $\stackrel{x}{\rightarrow}$ (can go to on input x)

  i. $\forall q \in Q,\ q \stackrel{\varepsilon}{\rightarrow} q$

  ii. $\forall x \in A^*, \forall e \in A \cup \{\varepsilon\}$, if $p \stackrel{x}{\rightarrow} q$ and q can go to r on input e then $p \stackrel{x.e}{\rightarrow} r$

Non-deterministic FA (NFA)

- Replace the transition function $\delta : Q \times A \to Q$ by either

i. A transition function $\Delta : Q \times (A \cup \{\varepsilon\}) \to 2^Q$ (power set of Q)
   - If $q \in \Delta(p, e)$ then the automaton can go from p to q on input e

ii. A transition relation $\Delta \subseteq Q \times (A \cup \{\varepsilon\}) \times Q$
   - If $(p, e, q) \in \Delta$ then the automaton can go from p to q on input e

- Inductive definition of $\overset{x}{\to}$ (can go to on input x)

  i. $\forall q \in Q,\ q \overset{\varepsilon}{\to} q$

  ii. $\forall x \in A^*, \forall e \in A \cup \{\varepsilon\}$, if $p \overset{x}{\to} q$ and q can go

For an NFA $N = (Q, s, \Delta, F)$

$L(N) = \left\{x \in A^* \middle| \exists q \in F,\ s \overset{x}{\to} q\right\}$

NFA-DFA equivalence

- **Theorem**: For any NFA $N = (Q, s, \Delta, F)$ with n states, there is a DFA M with at most 2^n states such that $L(M) = L(N)$
  - **Trivial**: For any DFA M there is an NFA N such that $L(M) = L(N)$
- **Construction**: Let $Q_M = 2^Q$, $S_M = \left\{q \in Q \middle| s \overset{\varepsilon}{\to} q\right\}$

$$\forall P \in Q_M, \forall a \in A, \qquad \delta_M(P, a) = \left\{q \in Q \middle| \exists p \in P, p \overset{a}{\to} q\right\}$$

$$F_M = \{P \in Q_M | P \cap F \neq \emptyset\}$$

- **Claim**: $\forall x \in A^*, \widehat{\delta_M}(S_M, x) = \left\{q \in Q \middle| s \overset{x}{\to} q\right\}$

Recall: L_k defined over $\{a, b\}$ as strings whose k^{th} last letter is b.
Any DFA for L_k has at least 2^k states, but there is a k+1 state NFA for L_k

Additional closure properties (1/2)

- For any two languages L_1 and L_2 over a common alphabet A, define

$$L_1.L_2 = \{x.y | x \in L_1 \text{ and } y \in L_2\}$$

- **Claim**: If L_1 and L_2 are regular then $L_1.L_2$ is regular

- **Proof sketch**: Suppose $L_1 = L(N_1)$ and $L_2 = L(N_2)$ where

$$N_1 = (Q_1, s_1, \Delta_1, F_1), N_2 = (Q_2, s_2, \Delta_2, F_2) \text{ and } Q_1 \cap Q_2 = \emptyset$$

- Define $N = (Q_1 \cup Q_2, s_1, \Delta, F_2)$ where $\Delta = \Delta_1 \cup \Delta_2 \cup \{(p, \varepsilon, s_2) | p \in F_1\}$

Additional closure properties (2/2)

- For any languages L over an alphabet A, define L^* inductively as:
  i. $\varepsilon \in L^*$
  ii. $\forall x \in L^*, \forall y \in L,\ x.y \in L^*$

- **Claim**: If L is regular then L^* is regular

- **Proof sketch**: Suppose $L = L(N)$ where $N = (Q, s, \Delta, F)$
- Define $N' = (Q \cup \{s_0\}, s_0, \Delta', F \cup \{s_0\})$ where

$$\Delta' = \Delta \cup \{(s_0, \varepsilon, s)\} \cup \{(p, \varepsilon, s) | p \in F\}$$

Regular expressions

- Define the set R_A of regular expressions over an alphabet A as:
  1. $\varnothing \in R_A$; $\varepsilon \in R_A$; $\forall a \in A,\ a \in R_A$
  2. $\forall r_1, r_2 \in R_A,\quad r_1 + r_2 \in R_A$ and $r_1.r_2 \in R_A$
  3. $\forall r \in R_A,\quad r^* \in R_A$
- Define language $L(r)$ of regular expressions r as:
  1. $L(\varnothing) = \varnothing$; $L(\varepsilon) = \{\varepsilon\}$; $\forall a \in A,\ L(a) = \{a\}$
  2. $\forall r_1, r_2 \in R_A,\quad L(r_1 + r_2) = L(r_1) \cup L(r_2)$ and $L(r_1.r_2) = L(r_1).L(r_2)$
  3. $\forall r \in R_A,\quad L(r^*) = (L(r))^*$
- **Claim**: $\forall r \in R_A,\ L(r)$ is regular