Proofs in lectures

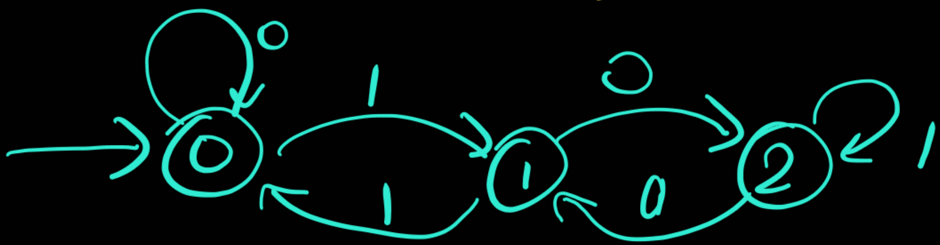
① $\hat{\delta}(q, a) = \delta(q, a)$ $\quad (a = \varepsilon a)$

$\hat{\delta}(q, \varepsilon a) = \delta(\hat{\delta}(q, \varepsilon), a) = \delta(q, a)$ definition

② $L(M) = \{$all strings that end in multiple of 3 in binary$\}$

$M = (Q, \Sigma, \delta, s, F)$ $\quad s = 0$

$F = \{0\}$



$\#\varepsilon = 0$

$Q = \{0, 1, 2\}$ $\quad \Sigma = \{0, 1\}$ $\quad \#x =$ value of the number in binary rep by x

$\delta(q, c) = (2q + c) \bmod 3$

To show: $\hat{\delta}(0, x) = (\#x) \bmod 3$

Induction on length of string x.

Base Case: string $= \varepsilon$

$\hat{\delta}(0, \varepsilon) = 0 = 0 \bmod 3 = \#\varepsilon \bmod 3$

(def^n of $\hat{\delta}$) $\quad$ (def^n of $\#\varepsilon$)

$\hat{\delta}(0, xc) = \delta(\hat{\delta}(0, x), c)$ ($\hat{\delta}$ definition)

$= (2(\hat{\delta}(0, x)) + c) \bmod 3$ (δ def^n)

$$= (2((\#x) \bmod 3) + c) \bmod 3 \quad \text{(IH)}$$

$$= (2((\#x) \bmod 3) + c) \bmod 3$$

$$= (2(\#x) + c) \bmod 3 \quad \text{(mod prop)}$$

$$= (\#xc) \bmod 3 \quad \text{(Lemma)}$$

Lemma $(\#xc) = (2\#x + c)$

$$\left.\begin{aligned} \#x0 &= 2(\#x) + 0 \\ \#x1 &= 2(\#x) + 1 \end{aligned}\right\} \#xc = 2(\#x) + c$$

③ $\hat{\Delta}(A, xy) = \hat{\Delta}(\hat{\Delta}(A, x), y)$

Induction on length of string

Let $y = za$ where a is a sym.

Base Case: $\hat{\Delta}(A, x\varepsilon) = \hat{\Delta}(A, x)$

$$= \hat{\Delta}(\hat{\Delta}(A, x), \varepsilon) \quad (\because \hat{\Delta}(A, \varepsilon) = A)$$

$$\hat{\Delta}(A, xa) = \bigcup_{q \in \hat{\Delta}(A, x)} \Delta(q, a) \quad (\text{def}^n)$$

$$\hat{\Delta}(A, a) = \bigcup_{q \in A} \Delta(q, a) \quad (\text{def}^n)$$

$$\hat{\Delta}(A, xza) = \bigcup_{q \in \hat{\Delta}(A, xz)} \Delta(q, a) \quad (\text{def}^n)$$

$q \in \hat{\Delta}(A, xz)$

$= \bigcup_{q \in B} \Delta(q, a)$ $\quad B = \hat{\Delta}(A, xz)$ (replace)

$B = \hat{\Delta}(\hat{\Delta}(A, x), z)$ (IH)

$= \hat{\Delta}(B, a)$ (defn)

$= \hat{\Delta}(\hat{\Delta}(\hat{\Delta}(A, x), z), a)$ (replace, IH)

$= \bigcup_{q \in \hat{\Delta}(\hat{\Delta}(A, x), z)} \Delta(q, a)$ (defn)

$= \hat{\Delta}(\hat{\Delta}(A, x), za) = \hat{\Delta}(\hat{\Delta}(A, x), y)$ (defn)

④ $\hat{\delta}_M(A, x) = \hat{\Delta}_N(A, x)$

Induction Hypothesis on the length of x.

Base Case: $x = \varepsilon$ $\quad \hat{\Delta}_N(A, \varepsilon) = A$

$\hat{\delta}_M(A, \varepsilon) = A$

$\hat{\delta}_M(A, ya) = \delta_M(\hat{\delta}_M(A, y), a)$ (def^n $\hat{\delta}_M$)

$= \delta_M(\hat{\Delta}_N(A, y), a)$ (IH)

$= \hat{\Delta}_N(\hat{\Delta}_N(A, y), a)$ (def^n of Δ_N)

by lemma ③

$= \hat{\Delta}_N (A, ya)$ (by lemma ③)

5 $L(M) = L(N)$

Let $x \in L(M)$ $\quad \hat{\delta}_M (s_m, x) \in F_M$

Let $y \in L(N)$ $\quad \hat{\Delta}_N (S_N, y) \cap F_N \neq \phi$

$\hat{\Delta}_N (S_N, y) = \hat{\delta}_M (S_N, y)$ (Above lemma)

$= \hat{\delta}_M (s_m, y)$ (def^n of $S_N = S_m$)

$\because y \in L(N) \Rightarrow \hat{\Delta}_N (S_N, y) \cap F_N \neq \phi$

$\Rightarrow \hat{\delta}_M (S_m, y) \cap F_N \neq \phi$

$\because \hat{\delta}_M (S_m, y) \in Q_M = 2^{Q_N} \Rightarrow \hat{\delta}_M (s_m, y) \subseteq Q_N$

$\therefore \hat{\delta}_M (S_m, y) \in F_M \Rightarrow L(N) \subseteq L(M)$

(the arrows are bidirectional)

6 For every NFA N $\exists$ a DFA M st $L(M) = L(N)$.

$N = (Q_N, \Sigma, \Delta_N, S_N, F_N)$ $\quad M = (Q_M, \Sigma, \delta_M, s_m, F_M)$

$Q_M = 2^{Q_N}$ $\quad s_m = S_N$ $\quad F_M = \{A \mid A \in Q_M, A \cap F_N \neq \phi\}$

$\delta_M (q, a) = \hat{\Delta} (q, a)$

$\hat{\delta}_M (A, x) = \delta_M (\hat{\delta}_m (A, y), a)$

$$= \delta_M(\hat{\Delta}_N(A,y), a)$$
$$= \hat{\Delta}_N(\hat{\Delta}_N(A,y), a)$$
$$= \hat{\Delta}_N(A, ya)$$

⑦ A DFA M can easily be converted to an equivalent NFA!

$M = (Q_M, \Sigma, \delta_M, s_M, F_M)$

$N = (Q_N, \Sigma, \Delta_N, S_N, F_N)$

$\Delta_N(q,a) = \{\delta_M(q,a)\}$ $\quad S_N = \{s_M\}$

$F_N = F_M = F \quad Q_N = Q_M = Q$

Lemma: $\hat{\Delta}_N(\{q\}, x) = \{\hat{\delta}_M(q,x)\}$

$\hat{\Delta}_N(\{q\}, \varepsilon) = \{q\} \quad \hat{\delta}_M(q,\varepsilon) = q \quad q \in Q$

Proof by induction on length of string.

Base case: $\hat{\Delta}_N(\{q\}, \varepsilon) = \{q\} = \{\hat{\delta}_M(q,\varepsilon)\}$
(def^n of $\hat{\Delta}_N$) (def^n of $\hat{\delta}_M$)

$$\hat{\Delta}_N(\{q\}, xa) = \bigcup_{p \in \hat{\Delta}_N(\{q\},x)} \Delta_N(p,a) \quad (\text{def}^n \text{ of } \hat{\Delta}_N)$$
$$= \bigcup_{p \in \{\hat{\delta}_M(q,x)\}} \Delta_N(p,a) \quad (\text{by IH})$$
$$= \Delta_N(\hat{\delta}_M(q,x), a) \quad (\text{Singleton set})$$

$= \{ \delta_M(\hat{\delta}_M(q, x), a) \}$ (def^n of Δ_N)

$= \{ \hat{\delta}_M(q, xa) \}$ (def^n of $\hat{\delta}_M$)

To prove : $L(M) = L(N)$

$x \in L(M) \Leftrightarrow \hat{\delta}_M(s_M, x) \in F_M$

$\Leftrightarrow \{ \hat{\delta}_M(s_M, x) \} \cap F_M \neq \phi$

$(F_M = F_N)$

because $\hat{\Delta}_N$ is singleton set

$\Leftrightarrow \hat{\Delta}_N(s_N, x) \cap F_N \neq \phi$ and $\hat{\Delta}_N(s_N, x) \subseteq Q_N$ $\Big\}$ $x \in L(N)$

8 Regular sets are closed under union, intersection, complementation, concatenation, kleene star,

---

Homomorphism

9 Let $h : \Sigma^* \to \Gamma^*$ be a homo. If $B \subseteq \Gamma^*$ is regular, then $h^{-1}(B)$ is also regular.

... be a DFA

Let $M = (Q, \Gamma, \delta, s, F)$ be a ...

st $L(M) = B$.

To show: $\exists$ M' over Σ st

$$L(M') = h^{-1}(B)$$

$$M' = (Q, \Sigma, \delta', s, F)$$

$$\delta'(q, a) = \hat{\delta}(q, h(a))$$

Lemma: $\hat{\delta}'(q, x) = \hat{\delta}(q, h(x))$

Proof: $\hat{\delta}'(q, \varepsilon) = q = \hat{\delta}(q, h(\varepsilon))$

$$\begin{aligned}
\hat{\delta}'(q, xa) &= \delta'(\hat{\delta}'(q, x), a) \\
&= \delta'(\hat{\delta}(q, h(x)), a) \\
&= \hat{\delta}(\hat{\delta}(q, h(x)), h(a)) \\
&= \hat{\delta}(q, h(x)\, h(a)) \\
&= \hat{\delta}(q, h(xa))
\end{aligned}$$

To show $L(M') = h^{-1}(B)$

$\because B = L(M) \Rightarrow L(M') = h^{-1}(L(M))$ can be shown equivalently

for any $x \in \Sigma^*$

$x \in L(M') \iff \hat{\delta}'(s, x) \in F$

$x \in L(\ldots)$

$\iff \hat{\delta}(s, h(x)) \in F$

$\iff h(x) \in L(M) \Rightarrow x \in h^{-1}(L(M))$

(10) If $A \subseteq \Sigma^*$ is regular, then $h(A)$ is regular.

Proof: α is a reg exp over Σ st $L(\alpha) = A$.

To show: $\exists\, \alpha'$ st $L(\alpha') = h(A)$
$= h(L(\alpha))$

Replace each letter $a \in \Sigma$ in α by string $h(a) \in \Gamma^*$, to form α'.

$a' = h(a) \quad \forall\, a \in \Sigma$. $\quad (\beta^*)' = (\beta')^*$

$\phi' = \phi \qquad \varepsilon' = \varepsilon$

$(\beta + \gamma)' = \beta' + \gamma' \qquad (\beta \cdot \gamma)' = \beta' \cdot \gamma'$

Claim: For any reg exp β over Σ, β' over Γ^*
$L(\beta') = h(L(\beta))$

- $C, D \subseteq \Sigma^*$, $h(CD) = h(C)\, h(D)$
- $C_i \subseteq \Sigma^*$, $i \in I$, we can say:

$$h\left[\bigcup_{i\in I} C_i\right] = \bigcup_{i\in I} h(C_i)$$

Proof: $L(a') = L(h(a)) = \{h(a)\}$
$= h(\{a\}) = h(L(a))$

$L(\phi') = L(\phi) = \phi = h(\phi) = h(L(\phi))$

<u>Induction on operators</u> $+, \cdot, *$

$$\begin{aligned}
L((\beta+\gamma)') &= L(\beta'+\gamma') && (\text{def}^n \text{ of } ') \\
&= L(\beta') \cup L(\gamma') && (\text{defn of } +) \\
&= h(L(\beta)) \cup h(L(\gamma)) && (IH) \\
&= h(L(\beta) \cup L(\gamma)) && (\text{Lemma}) \\
&= h(L(\beta+\gamma)) && (\text{defn of } +)
\end{aligned}$$

$$\begin{aligned}
L((\beta^*)') &= L((\beta')^*) && (\text{def}^n \text{ of } ') \\
&= (L(\beta'))^* && (\text{def}^n \text{ of } *) \\
&= (h(L(\beta)))^* && (IH) \\
&= \bigcup_{n\geq 0} (h(L(\beta)))^n && (\text{def}^n \text{ of } * \text{ in sets})
\end{aligned}$$

$n \geq 0$

$= \bigcup h((L(\beta))^n)$ (Lemma 2)

$= h(\bigcup L(\beta)^n)$ (Lemma 3)

$= h(L(\beta)^*)$ (def^n of set $*$)

$= h(L(\beta^*))$ (def^n of regex $*$)

Regex $\Longleftrightarrow$ M

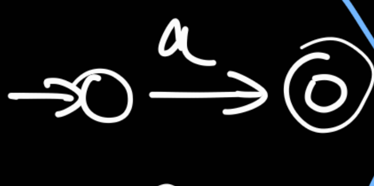
(11) $A = L(\alpha)$ for some pattern α.
Then A is regular.

redundant

Atomic pattern $a \in \Sigma$, ϵ, ϕ, $\boxed{\#, @}$

Compound $+$, $\boxed{\cap}$, $*$, $\boxed{+}$, $\cdot$, $\neg$

Base cases :

i) $a \in \Sigma$ $L(a) = \{a\}$ 

ii) ϵ : $L(\epsilon) = \epsilon$ 

iii) ϕ $L(\phi) = \phi$ 

all are regular

Compound :

Compound:

$\beta + \gamma$: $L(\beta R \gamma) = L(\beta) R L(\gamma)$ (defⁿ of +)

By IH, $L(\beta)$ & $L(\gamma)$ are regular.
Reg. sets are closed under union, intersection, complement, concat, Kleene star. ⇒ all proved.

(12) If A is regular, then $\exists$ a reg exp α st $L(\alpha) = A$.

Let $M = (Q, \Sigma, \Delta, S, F)$ be an NFA without ε transitions, st $L(M) = A$.
for all $Y \subseteq Q$, and $u, v \in Q$, we construct a reg exp α_{uv}^{Y}

α_{uv}^{Y} : set of strings x st there is a path from u to v in M labelled x ($v \in \hat{\Delta}(u, x)$) and all states along the path with possible exception of u & v lie in Y.

Proof: $Y = \phi$ Base case $\quad a_1, a_2, \ldots a_k \in \Sigma$ st $v \in \Delta(u, a_i)$

i) $u \neq v$

$$\alpha_{uv}^{\phi} = \begin{cases} \phi & k = 0 \\ a_1 + a_2 + \ldots + a_k & k > 0 \end{cases}$$

ii) $u = v$

$$\alpha_{uv}^{\phi} = \begin{cases} \varepsilon & k = 0 \\ a_1 + a_2 + \ldots + a_k & k > 0 \end{cases}$$

Induction Step:

$$\alpha_{uv}^{Y} = \alpha_{uv}^{Y - \{q\}} + \alpha_{uq}^{Y-\{q\}} \left(\alpha_{qq}^{Y-\{q\}}\right)^* \alpha_{qv}^{Y-\{q\}}$$

Choose an arbitrary state $q \in Y$.

α = Sum (union) of all expr of the form α_{sf}^{Q} st $s \in S$ $f \in F$

Reverse of A

$x = a_1 a_2 \ldots a_n \qquad rev(x) = a_n \ldots a_2 a_1$

If A is regular $\Rightarrow$ $rev(A)$ also regular.

$M = (Q, \Sigma, \delta, s, F) \quad L(M) = A \quad L(N) = rev(A)$

$N = (Q \cup \{s'\}, \Sigma, \Delta, s', \{s\})$

$\Delta(s', a) = \ldots \quad \Delta(s', a) = \phi \quad \forall a \in \Sigma$

$\Delta(s', \varepsilon) = F \qquad \Delta'(x, a) = \varphi$

$\Delta(p,a) = \{q \mid \delta(q,a) = p\} \quad \forall\, p \in Q,\ a \in \Sigma.$

Quotient Automata

① δ' is well defined.

if $[p] = [q]$ then $[\delta(p,a)] = [\delta(q,a)]$

i.e. if $p \approx q$ then $\delta(p,a) \approx \delta(q,a)$

Proof: $p \approx q \iff \hat{\delta}(p,a) \in F \iff \hat{\delta}(q,a) \in F$
$\forall\, a \in \Sigma$

For $a \in \Sigma$ and $y \in \Sigma^*$

$\hat{\delta}(\delta(p,a), y) \in F$ iff $\hat{\delta}(p, ay) \in F$

iff $\hat{\delta}(q, ay) \in F$

iff $\hat{\delta}(\delta(q,a), y) \in F$

if $p \approx q$

$\Rightarrow \delta(p,a) \approx \delta(q,a) \quad \forall\, y \in \Sigma^*$

② $p \in F$ iff $[p] \in F'$

$F' = \{[p] \mid p \in F\} \implies$ Fwd dir easy

$p \approx q \iff \hat{\delta}(p,\varepsilon) \in F$ iff $\hat{\delta}(q,\varepsilon) \in F$

$\iff p \in F$ iff $q \in F$.

So if $p \approx q$ and $p \in F \implies q \in F$

every $\approx$ eq. class is either a subset of F or completely disjoint from F.

③ $\forall x \in \Sigma^* \quad \hat{\delta}'([p], x) = [\hat{\delta}(p, x)]$

By induction on $|x|$.

<u>Base</u>: $x = \varepsilon \Rightarrow \hat{\delta}'([p], \varepsilon) = [p] = [\hat{\delta}(p, \varepsilon)]$

$$\begin{aligned}
\hat{\delta}'([p], xa) &= \delta'(\hat{\delta}'([p], x), a) && (\text{def}^n \text{ of } \hat{\delta}') \\
&= \delta'([\hat{\delta}(p, x)], a) && (\text{IH}) \\
&= [\delta(\hat{\delta}(p, x), a)] && (\text{defn of } \delta') \\
&= [\hat{\delta}(p, xa)] && (\text{defn of } \hat{\delta})
\end{aligned}$$

④ $L(M/\approx) = L(M)$.

$x \in \Sigma^* \quad \Leftrightarrow x \in L(M/\approx) \Leftrightarrow \hat{\delta}'(s', x) \in F'$

$\Leftrightarrow \hat{\delta}'([s], x) \in F' \Leftrightarrow [\hat{\delta}(s, x)] \in F'$

$\Leftrightarrow \hat{\delta}(s, x) \in F \Leftrightarrow x \in L(M)$

Double Quotient Construction:

$[p] \sim [q] \Leftrightarrow \forall x \in \Sigma^*$

$\quad (\hat{\delta}'([p], x) \in F' \text{ iff } \hat{\delta}'([q], x) \in F')$

$\Leftrightarrow \forall x \in \Sigma^*$

$\quad ([\hat{\delta}(p, x)] \in F' \text{ iff } [\hat{\delta}(q, x)] \in F')$

$\Leftrightarrow \forall x \in \Sigma^* \quad \hat{\delta}(p, x) \in F \text{ iff } \hat{\delta}(q, x) \in F$

$\Leftrightarrow p \approx q \quad \Leftrightarrow \quad [p] = [q]$

State Minimisation Algo

Theorem: The pair $\{p,q\}$ is marked by the Algo iff $\exists x \in \Sigma^*$ st $\hat{\delta}(p,x) \in F$ and $\hat{\delta}(q,x) \notin F$ or vice versa.

That is, iff $p \not\approx q$.

Proof: if $\{p,q\}$ gets marked.

for some $a \in \Sigma$, $\{\delta(p,a), \delta(q,a)\}$ marked.

$\exists y \in \Sigma^*$ st

$\{\hat{\delta}(\delta(p,a),y), \hat{\delta}(\delta(q,a),y)\}$ marked.

$\Rightarrow \{\hat{\delta}(p,ay), \hat{\delta}(q,ay)\}$ marked.

Then $x = ay$ is the string.

if $\{p,q\}$ is unmarked, then $\forall x \in \Sigma^*$ $\hat{\delta}(p,x) \in F$ iff $\hat{\delta}(q,x) \in F$

Induction on $|x|$. $x = ay$.

$\hat{\delta}(p,ay) \in F \Leftrightarrow \hat{\delta}(\delta(p,a),y) \in F$

$\{\delta(p,a), \delta(q,a)\} \rightarrow$ unmarked

$\Leftrightarrow \hat{\delta}(\delta(q,a),y) \in F$ (IH)

$\Leftrightarrow \hat{\delta}(q,ay) \in F$

Myhill Nerode Relation

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

(x, y may or may not belong to R)

① RC: $\forall x, y \in \Sigma^* \quad \forall a \in \Sigma$

$x \equiv_M y \Rightarrow xa \equiv_M ya$

$\hat{\delta}(s, xa) = \delta(\hat{\delta}(s,x), a)$ ($\hat{\delta}$ defn)

$= \delta(\hat{\delta}(s,y), a)$ ($\equiv_M$)

$= \hat{\delta}(s, ya)$ ($\hat{\delta}$ defn)

② Ref: $\forall x, y \in \Sigma^* \quad x \equiv y \Rightarrow (x \in R \text{ iff } y \in R)$

$\because \hat{\delta}(s,x) = \hat{\delta}(s,y)$

③ F Id: Since finite no. of states in a DFA.

so there is exactly one eq. class corresponding to each $q \in Q$.

$\{x \in \Sigma^* \mid \hat{\delta}(s,x) = q\}$

MN Reln to DFA:

$[x] = \{ y \mid y \equiv x \}$

$M_{\equiv} = (Q, \Sigma, \delta, s, F)$

$Q = \{ [x] \mid x \in \Sigma^* \}$

$F = \{ [x] \mid x \in R \}$

$s = [\epsilon]$

$\delta([x], a) = [xa]$

(P1) $x \in R$ iff $[x] \in F$

$x \in R \Rightarrow [x] \in F$ by defn of F

if $x \equiv y \Rightarrow x \in R$ iff $y \in R$.

$\therefore [x] \in F \Rightarrow x \in F \Rightarrow y \in F$.

Lemma: $\hat{\delta}([x], y) = [xy]$

Induction on $|y|$.

Base: $\hat{\delta}([x], \epsilon) = [x] = [x\epsilon]$.

Ind: $\hat{\delta}([x], ya) = \delta(\hat{\delta}([x], y), a)$

$= \delta([xy], a) = [xya]$

$L(M_\equiv) = R$

$x \in R \iff [x] \in F \iff [\epsilon x] \in F$

$\iff \hat{\delta}([\epsilon], x) \in F \iff \hat{\delta}(s, x) \in F$

$\iff x \in L(M_\equiv)$

---

(1) $\equiv \longrightarrow M_\equiv \longrightarrow \equiv_{M_\equiv}$ (Identical)

Proof: $x \equiv_{M_\equiv} y \iff \hat{\delta}(s, x) = \hat{\delta}(s, y)$

$\iff \hat{\delta}([\epsilon], x) = \hat{\delta}([\epsilon], y)$

$\iff [\epsilon x] = [\epsilon y] \iff [x] = [y]$

$\iff x \equiv y$

(2) $M \rightarrow \equiv_M \longrightarrow M_{\equiv_M}$ (Isomorphic)

Proof: $M = (Q, \Sigma, \delta, s, F)$

$M_{\equiv_M} = (Q', \Sigma, \delta', s', F')$

$[x] = \{ y \mid y \equiv_M x \} = \{ y \mid \hat{\delta}(s,x) = \hat{\delta}(s,y) \}$

$Q' = \{ [x] \mid x \in \Sigma^* \}$ $\quad s' = [\varepsilon]$

$F' = \{ [x] \mid x \in L(M) \}$ $\quad \delta'([x], a) = [xa]$

Consider $F : Q' \to Q$

$F([x]) = \hat{\delta}(s, x)$

$[x] = [y]$ iff $\hat{\delta}(s,x) = \hat{\delta}(s,y)$

so F is well defined, and one-one.

F is onto as M has no inaccessible states.

$\forall q \in Q, \exists x \in \Sigma^*$ st $\hat{\delta}(s,x) = q$.

or $\forall q \in Q, \exists q' = [x] \in Q'$ st

$F([x]) = \hat{\delta}(s,x) = q = F(q')$

$M_{\equiv_M}$ and M are isomorphic under F.

i) $F(s') = s$

ii) $F(\delta'([x], a)) = \delta(F([x]), a)$

iii) $[x] \in F'$ iff $F([x]) \in F$

① $F(s') = F([\varepsilon]) = \hat{\delta}(s, \varepsilon) = s$.

② $F(\delta'([x], a)) = F([xa])$

$= \hat{\delta}(s, xa) = \delta(\hat{\delta}(s,x), a)$

$= \delta(F([x]), a)$

③ $[x] \in F \iff x \in R \iff \hat{\delta}(s,x) \in F$

$\Leftrightarrow F([x]) \in F$

$\equiv_R$: Coarsest

$x \equiv_R y$ iff $\forall z \in \Sigma^* (xz \in R$ iff $yz \in R)$

① $\equiv_R$ is right congruent

Take $z = aw$. $a \in \Sigma$ $w \in \Sigma^*$

$x \equiv_R y \Leftrightarrow \forall a \in \Sigma \ \forall w \in \Sigma^*$
$(xaw \in R$ iff $yaw \in R)$

$\Leftrightarrow \forall a \in \Sigma \ (xa \in R \Leftrightarrow ya \in R)$

② $\equiv_R$ refines R

take $z = \varepsilon$. $x \in R$ iff $y \in R$.

③ Coarsest.

Let $\equiv$ be any other rel n (satisfying P_1 and P_2)

$x \equiv y \Rightarrow \forall z \ (xz \equiv yz)$ (induction on $|z|$)

$\Rightarrow \forall z \ (xz \in R$ iff $yz \in R)$

$\Rightarrow x \equiv_R y$.

Myhill Nerode Theorem

$\equiv_R$ of finite index then $\equiv_R$ is a MNR for R.

$x \equiv_R y \Leftrightarrow \forall z \in \Sigma^* (xz \in R$ iff $yz \in R)$

$x \equiv_R y \iff \forall z \in \Sigma^* \; (xz \in R \text{ iff } yz \in R)$

$\iff \forall z \in \Sigma^* \left(\hat{\delta}(s, xz) \in F \text{ iff } \hat{\delta}(s, yz) \in F \right)$

$\iff \forall z \in \Sigma^* \left(\hat{\delta}(\hat{\delta}(s, x), z) \in F \iff \hat{\delta}(\hat{\delta}(s, y), z) \in F \right)$

$\iff \hat{\delta}(s, x) \approx \hat{\delta}(s, y)$

$\iff \hat{\delta}(s, x) = \hat{\delta}(s, y)$ (since M is collapsed)

$\iff x \equiv_M y$

Context Free Languages

$Z = \{a^n b^n \mid n \geq 0\}$

$G = (N, \Sigma, P, S)$ $\quad N = \{S\}$ $\quad \Sigma = \{a, b\}$

$P = \{S \rightarrow aSb, \; S \rightarrow \varepsilon\}$ $\quad L(G) = Z.$

$S \rightarrow aSb \mid \varepsilon$

$S \xrightarrow[G]{n+1} a^n b^n$ prove using induction on n.

Base Case: $n = 0 \Rightarrow a^n b^n = \varepsilon.$

and $S \xrightarrow[G]{1} \varepsilon$ (by productions)

Induction case: In one step of derivation,

$S \xrightarrow[G]{1} aSb \xrightarrow[G]{1} aaSbb \xrightarrow[G]{2} \cdots \xrightarrow[G]{n} a^n S b^n$

Now in the $(n+1)^{th}$ step, $S \xrightarrow[G]{1} \varepsilon$ which will give $S \xrightarrow[G]{n} a^n S b^n \xrightarrow[G]{1} a^n b^n$

$\Rightarrow$ all strings of the form $a^n b^n \in L(G)$

only strings in $L(G)$ are of the form $a^n b^n$.

(Induction on the length of the derivation)

The last step of derivation must be $S \to \epsilon$
otherwise we will have S present in the string, and L(G) must not have any non terminal in it by definition.

Now, in every step of derivation the number of non-terminals either dec by 1 or remains the same. Since we have only one non terminal at start, once we use the epsilon production, we cannot derive any further, as we would have 0 Non-terminals left.

Now, if we have an n-step derivation, the n^{th} step must be $S \to \epsilon$. So all steps from 1 to n-2 must be $S \to aSb$.

So $S \xrightarrow[G]{1} aSb \xrightarrow[G]{1} a^2Sb^2 \to \cdots \xrightarrow[G]{n-1} a^{n-1}Sb^{n-1} \xrightarrow{n} a^{n-1}b^{n-1}$

Therefore, any string derivable from S having only terminal symbols, must be of the form $a^n b^n$.

L(G) = all strings $\in \Sigma^*$ and derivable from S in G in finitely many steps.