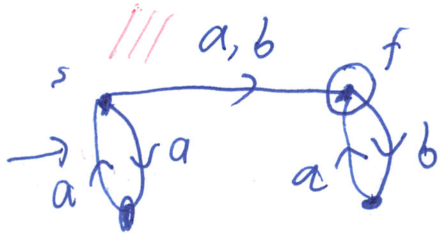
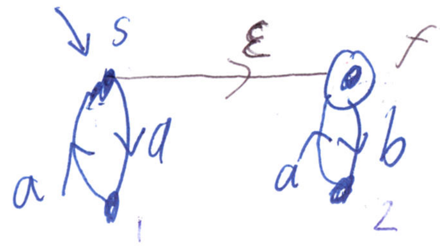
CSE 396 Lecture Tue 2/19 Spr 2019

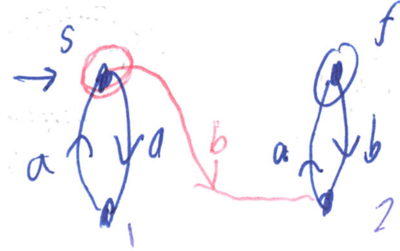
Top Hat # 9795 Often one does not need (so many) ε-transitions

Regexp $r = (aa)^*(a+b)(ba)^*$ $r' = (aa)^*(ba)^*$

$N_r =$ 

$N' =$ 

What happens with r', N', and N'' when $x = bb$, or $x = aabb$?
Can we process x, let alone accept it?

N'' 

Trial computation paths: $(s, a, 1, a, s, \varepsilon, f, b, 2, b$ die. (in N_1)

in N' also $[s, \varepsilon, f, a$ crash: die

in N'' only: $[s, a, 1, a, b, 2,$ die.

$\therefore x$ cannot be processed by either machine

N' can process ε from s to either s or f.
N'' can process ε from s only to s.

Defn: For any states p, q ($p = q$ allowed) of an NFA $N = (Q, \Sigma, \delta, s, F)$ and string $x \in \Sigma^*$, letting $n = |x|$, say that N can process x from p to q if there is a computation path

$$[p, w_1, q_1, w_2, q_2, w_3, q_3, \ldots\ldots q_{m-1}, w_m, q_m]$$

Such that:
- each w_j is either a char or ε, and $m \geq n$.
- For each j, $1 \leq j \leq m$, (q_{j-1}, w_j, q_j) is an instruction in δ of N.
- $x = w_1 \circ w_2 \circ \cdots \circ w_m$. (implies $m \geq n$).

$q_0 = p$]
$q_m = q$]

And define $L_{pq}(N) = \{x \in \Sigma^* : N$ can process x from p to $q\}$. (Or just L_{pq}.)

Then we can formally define $L(N) = \bigcup_{f \in F} L_{sf}$.

Works for a DFA M too since a DFA "Is-A" NFA.

Given x and i, $0 \le i \le n$, define

$R_i = \{q : N$ can process $x_1 \cdots x_i$ from s to $q\}$

$R_0 = \{q : \varepsilon \in L_{sq}\}$. In N_r and N'', $R_0 = \{s\}$. In N', $R_0 = \{s, f\}$

<u>Theorem</u>: For every NFA $N = (Q, \Sigma, \delta, s, F)$, we can build a DFA $M = (\mathcal{Q}, \Sigma, \Delta, S, \mathcal{F})$ such that $L(M) = L(N)$.

<u>Proof</u>: We will maintain the induction invariant that for any $x \in \Sigma^*$ and all i, $0 \le i \le n$, R_i equals the state M is in upon reading $x_1 \cdots x_i$. We define $\mathcal{F} = \{R \subseteq Q : R$ includes a $q \in F\}$.

<u>Basis</u> $(i=0)$: Make $S = R_0 = \{q : \varepsilon \in L_{sq}\}$ Text calls this $E(s)$, the ε-closure of $\{s\}$.

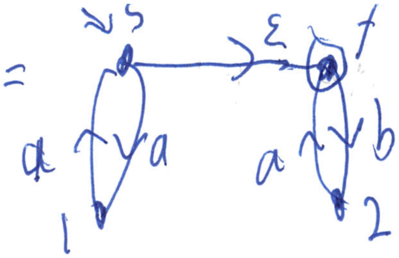
For all $P \subseteq Q$ and $c \in \Sigma$ define $\Delta(P, c) = \bigcup_{p \in P} \{q : N$ can process c from p to q **by first reading c**$\}$

Suppose M is in state R_{i-1} after processing $x_1 \cdots x_{i-1}$. Take $c = x_i$. Say $P' = \Delta(P, c)$. Need to show $P' = R_i$ which is defined in terms of the NFA

$R_{i-1} = \{p : N$ can process $x_1 \cdots x_{i-1}$ from s to $p\}$

[Proof conveyed in words] (will write out later) that $P' = R_i$. Thus the induction "goes through" and $L(M) = L(N)$. ⊠

$Q = \{s, f, 1, 2\}$

Example $N' =$ 

"Whenever s, then also f."

$$\Delta(P,c) = \bigcup_{p \in P} \underline{\delta}(p,c)$$

where $\underline{\delta}(p,c)$ is the RHS of the def^n of Δ after the $\bigcup$.

$\mathcal{F} = \{R \subseteq Q : R \text{ includes } f\}$.

$S = \{s, \underline{f}\}$, not just $\{s\}$, so $S \in \mathcal{F}$.

$\underline{\delta}$	a	b
s	1	$\emptyset$
f	$\emptyset$	2
1	s, f	$\emptyset$
2	f	$\emptyset$

Do not make a separate column for ε as in the text and other sources. We've already handled ε.

Lecture did not finish the example, but once you get the $\underline{\delta}$ table correct it's automatic:

$\Delta(S,a) = \underline{\delta}(s,a) \cup \underline{\delta}(f,a)$
$= \{1\} \cup \emptyset = \{1\}$

$\Delta(S,b) = \underline{\delta}(s,b) \cup \underline{\delta}(f,b) = \emptyset \cup \{2\} = \{2\}$

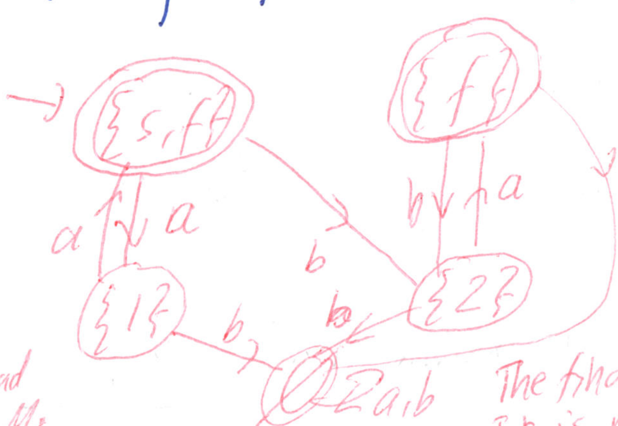
$\Delta(\{1\},a) = \underline{\delta}(1,a) = \{s,f\}$.

$\Delta(\{1\},b) = \underline{\delta}(1,b) = \emptyset$. which is always a dead state if you get it in M.

$\Delta(\{2\},a) = \underline{\delta}(2,a) = \{f\}$. New state, so expand it.

$\Delta(\{2\},b) = \underline{\delta}(2,b) = \emptyset$. $\Delta(\emptyset,a) = \Delta(\emptyset,b) = \emptyset$.

$\Delta(\{f\},a) = \underline{\delta}(f,a) = \emptyset$ $\Delta(\{f\},b) = \underline{\delta}(f,b) = \{2\}$. No more new states, so done.



The final DFA. It is minimal.

Note that we could have had as many as $2^{|Q|} = 16$ states, but we only needed 5. This way of economically going about the states is an example of breadth-first search (BFS). [CSE33

Writing Out what was said orally in the NFA-to-DFA Proof:

To show $P' = R_i$, we need to show that any $q \in P'$ belongs to R_i and vice-versa. So first suppose $q \in P'$. By $P' = \Delta(P,c)$, where P was the state of M after reading $x_1 \cdots x_{i-1}$, we have that N can process c from p to q (by first reading c). By inductive hypothesis, we have $P = R_{i-1}$, which means that since $p \in P$, we have $p \in R_{i-1}$, so N can process $x_1 \cdots x_{i-1}$ from s to p. Putting those together, N can process $x_1 \cdots x_{i-1} \cdot c = x_1 \cdots x_i$ from s to q (via p), which means $q \in R_i$. So the first half is done. Now suppose $q \in R_i$. This means N can process x from s to q via a path $(s, \ldots, p, c, \ldots q)$ where p is the state it was in just before N read c. So $p \in R_{i-1}$. By Ind-Hyp., $p \in P$. And N can process c from p to q by first reading c. So $q \in \Delta(P,c) = P'$. ☒