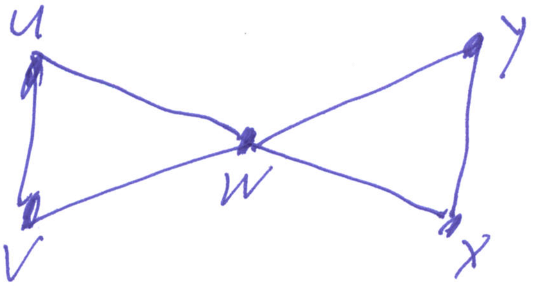
CSE 396 <u></u>

Lecture Tue 2/14

Spr 2017

Example: Suppose a relation $R(x,y)$ on 5 elements has a graph that looks like:



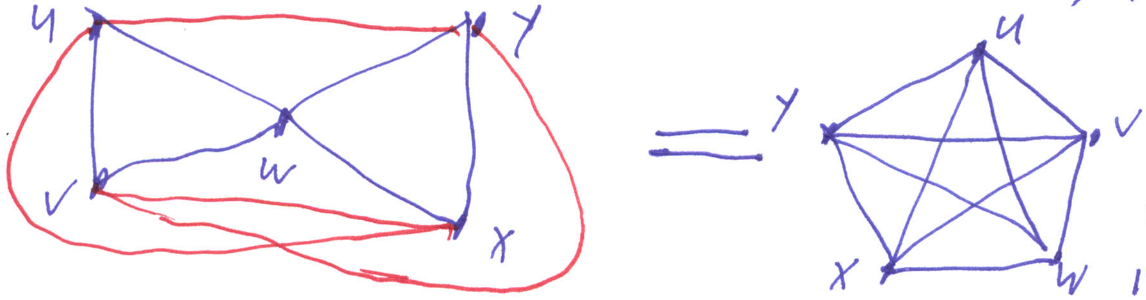
We can do the transitive closure by adding "missing" cases like $R(u,y)$.

$$x \xrightarrow{R(x,y)} y$$

$R(u,v) \wedge R(v,w)$ $R(u,w)$? yes, OK

$R(u,w) \wedge R(w,y)$ $R(u,y)$? no

so this R is not transitive.



The transitive closure of a connected undirected graph is always a complete graph.

Definition: A nondeterministic finite automaton (NFA) is a 5-tuple $N = (Q, \Sigma, \delta, s, F)$ where $\begin{matrix} Q \\ \Sigma \\ s \\ F \end{matrix}$ are like in a DFA but instead of $\delta : Q \times \Sigma \to Q$,

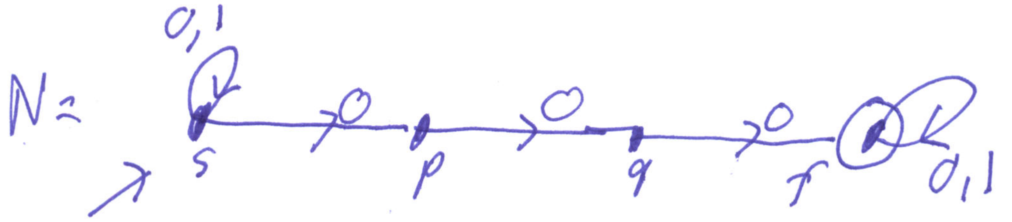
an NFA has $\delta : Q \times \Sigma \to P(Q)$ ("strict NFA")

or in our text $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \to P(Q)$, where $P(Q)$ is the powerset of Q, ie. has all subsets of Q.

②

Example 1: An NFA N such that $L(N) = \{x \in \{0,1\}^*:$ x does have a 000 in it$\}$

$N =$



Nondeterministic because $\delta(s,0) = \{s,p\}$ "two or more" options.

and because $\delta(p,1) = \emptyset$, ditto $\delta(q,1) = \emptyset$

(even though this not count as "actual nondeterminism")

E.g. $x = {}^{s}0\,{}^{s}1\,{}^{s}0\,{}^{s}0\,{}^{s}0\,{}^{s}1\,{}^{s}0\,{}^{p}$ — "too late: processing stops in state p.

${}_{s}\ {}_{s}\ {}_{s}\ {}_{p}\ {}_{q}\ {}_{f}\ {}_{f}\ {}_{f}$ → this sequence accepts x,

N optically resembles the regular expression $(0 \cup 1)^* 000 (0 \cup 1)^*$.

We can trace the possibilities deterministically as follows.

$x = \{s\}\ 0\ \{s,p\}\ 1\ \{s\}\ 0\ \{s,p\}\ 0\ \{s,p,q\}\ 0\ \{s,p,q,f\}\ 1\ \{s,f\}\ 0\ \{s,p,f\}$

died

The final $\{s,p,f\}$ is good because it includes the original final state f.

Now consider the complement $\tilde{L} = \{x : x$ does not have 000 in it$\}$.

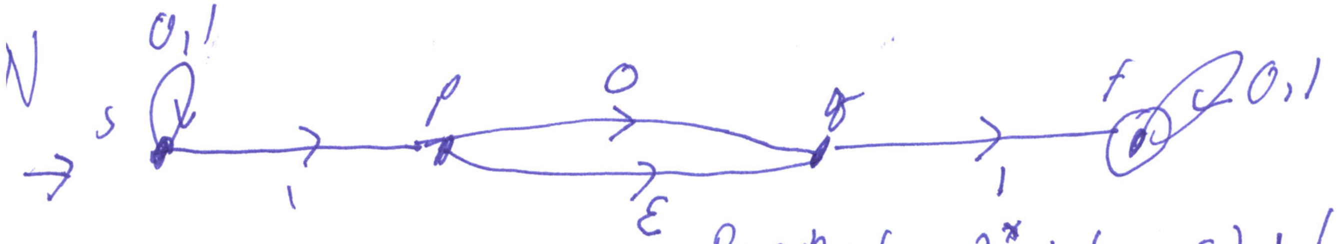
I don't know any NFA simpler than complementing the previous DFA.

$N' =$

Complementing states doesn't work because N' still accepts x

The previous "chicken" trace of staying in s until the last char now becomes an accepting one.

We can do $L_2 = \{x : x \text{ has a } 11 \text{ in it}\}$ similarly.
Text example $L_3 = \{x : x \text{ has a } 11 \text{ or a } 101 \text{ in it}\}$.



In text's style

Regexp: $(0 \cup 1)^* 1 (0 \cup \varepsilon) 1 (0 \cup 1)^*$

$\delta(s,0) = \{s\}$ $\quad \delta(p,0) = \{q\}$ $\quad \delta(q,0) = \emptyset$ (no saying ε option either)

$\delta(s,1) = \{s,p\}$ $\quad \delta(p,\varepsilon) = \{q\}$ $\quad \delta(q,1) = \{f\}$ $\quad \delta(f,0) = \delta(f,1) = \{f\}$.

$\delta(s,\varepsilon) =$ what? $\quad \delta(p,1) = \emptyset$ even though you can process 1 by first taking ε to q and 1 to f.

Note that N can process 0110 by the trace

$$\begin{array}{ccccccccccc} s & & s & & p & & q & & f & & f \\ & 0 & & 1 & & \varepsilon & & 1 & & 0 & \end{array}$$

$\boxed{\begin{array}{l} 5 \text{ steps for} \\ 4 \text{ chars} \end{array}}$

I prefer to think of δ as a set of instructions:

$$\delta = \{(s,0,s), (s,1,s), (s,1,p), \underline{(p,0,q)}, \underline{(p,\varepsilon,q)}, (q,1,f), (f,0,f), (f,1,f)\}.$$

$$\underline{\delta \subseteq Q \times (\Sigma \cup \{\varepsilon\}) \times Q}.$$

Advantages:

- No need to mention "$\mathcal{P}(Q)$" (yet).
- No need to worry about ε-cases that don't arise.

ef^n: An NFA can process a string x from state q to state r if there is a computation trace $\boxed{(q_0, w_1, q_1, w_2, q_2, \ldots, q_{m-1}, w_m, q_m)}$ .t. $q_0 = q$, $q_m = r$, $x = w_1 \cdot w_2 \cdots w_m$ (some w_i's can be ε), and $(\forall i)(q_{i-1}, w_i, q_i) \in \delta$.