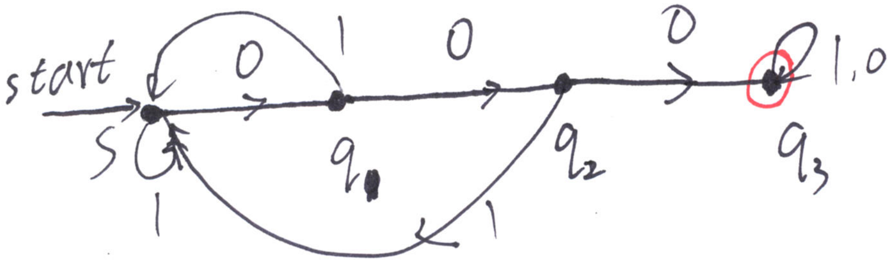
Let $A = \{x \in \Sigma^* : x \text{ has 3 consecutive } 0\text{s in it}\}$.

$\Sigma = \{0, 1\}$



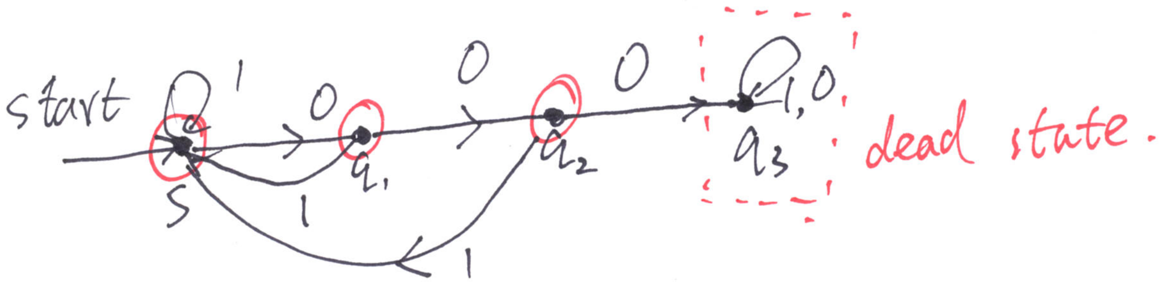
$Q = \{s, q_1, q_2, q_3\}$.

$F = \{q_3\}$.

The complement of A, (denoted by $\sim A$ or $\hat{A}$),
(Text $\bar{A}$)

$\sim A = \{x \in \Sigma^* : x$ ~~ha~~ does not ~~have~~ 3 consecutive 0s $\}$

$\sim F = Q \setminus F = \{s, q_1, q_2\}$.



Theorem: Given any DFA $M = (Q, \Sigma, s, \delta, F)$ accepting a language A, we can build a DFA $M' = (Q', \Sigma', s', \delta', F')$ s.t. $L(M') = \hat{A}$.

Proof: Design M' by taking $Q' = Q$, $s' = s$, and $\delta' = \delta$ but define $F' = Q \setminus F$.

then $L(M') = \{x \in \Sigma^*: M'$ on input x ends up in a state in $F'\}$.

$= \{x \in \Sigma^*: M'$ on x does not end up in a state in $F\}$.

$= \sim\{x \in \Sigma^*: \underbrace{M' \text{ on } x}_{M'(x)}$ ends up in a state in $F\}$

$= \sim L(M)$ since M and M' ~~not~~ work the sam

$= \sim A$ $\boxed{\times}$

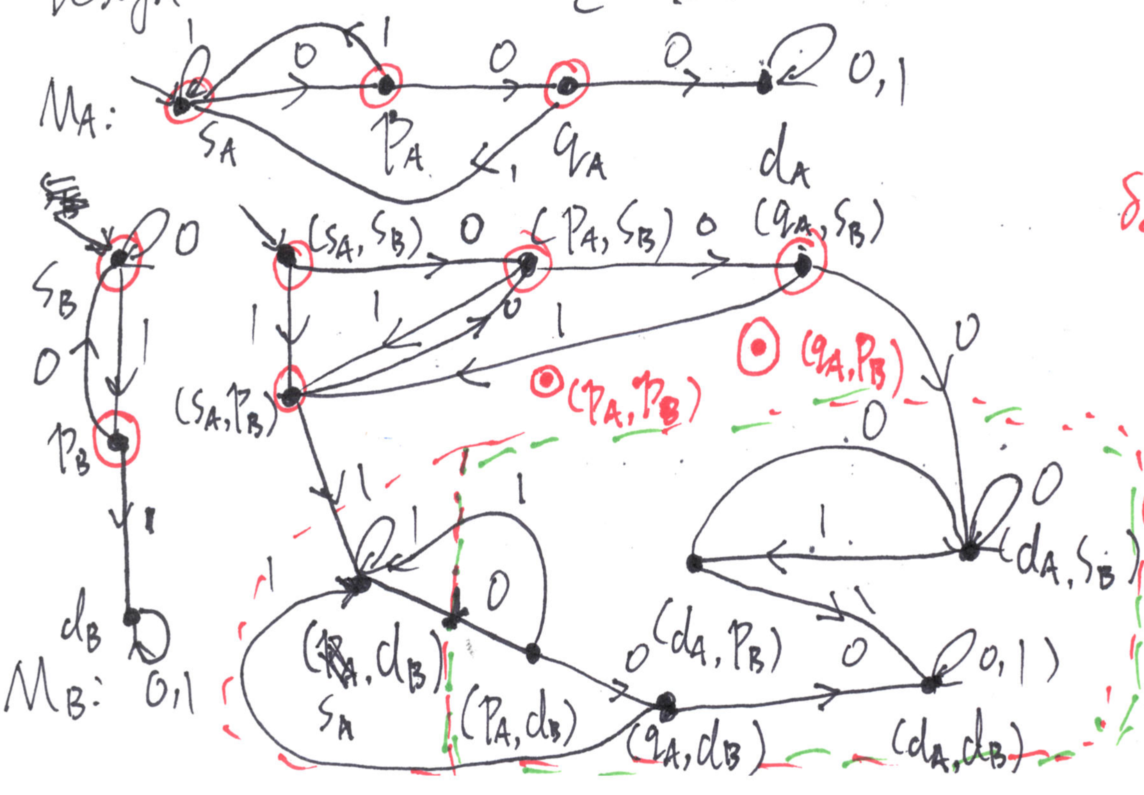
---

consider Cartesian Product Example: $\Sigma = \{0, 1\}$

$A = \{x \in \Sigma^*: 000$ is not a substring in $x\}$. M_A

$B = \{x \in \Sigma^*: 11$ is not a substring of $x\}$. M_B

Design a DFA M_C s.t. $L(M_C) = A \cap B$



$Q_C = Q_A \times Q_B$

$s_C = (s_A, s_B)$

$\delta_C((q_A, q_B), c) = (\delta_A(q_A, c), \delta_B(q_B, c))$

$\delta_C((q_A, s_B), 1) = (s_A, p_B)$

13

$L(M_C) = A \cap B$

$= \{x \in \Sigma^* : 000$ is not a substring of x and 11 is not a substring of $x\}$.

$A \setminus B$

$F_C = \{(q_A, q_B) : q_A \in F_A$ and $q_B \in F_B\}$.

---

more generally, consider $L_3 = L_1 \ \underline{op}\ L_2$

$op \in \{\cap, \cup, \setminus, \Delta, \text{etc}\}$.

boolean operation

$A \setminus B = \{x : x \in A$ but $x \notin B\}$.

$Q_3 = Q_1 \times Q_2$

$s_3 = (s_1, s_2)$

$\delta_3((q_1, q_2), c) = (\delta_1(q_1, c), \delta_2(q_2, c))$

$F_3 = \{(q_1, q_2) : q_1 \in F_A \ \underline{\underline{op}}\ q_2 \in F_B\}$

$op \in \{$ and, or, $\uparrow$

$A \setminus B \Longleftrightarrow \{(q_1, q_2) : q_1 \in F_A$ and $q_2 \notin F_B\}$.

$A \Delta B = (A \setminus B) \cup (B \setminus A)$

$\Updownarrow$

$\{(q_1, q_2) : q_1 \in F_A$ and $q_2 \notin F_B$ or $q_1 \notin F_A$ and $q_2 \in F_B\}$.

$q_1 \in F_A$ XOR $q_2 \in F_B$

$\Rightarrow$

Theorem: For any two languages A, B accepted by DFAs M_A and M_B, we can build a DFA M_C s.t.

$$L(M_C) = L(M_A) \overset{op}{\cancel{\ }} L(M_B), \text{ where } op = \{\cap, \cup, \setminus, \sim, \Delta, etc.$$

$$L(M_C) = L(M_A) \cap L(\overset{M_B}{\cancel{B_B}})$$

$$L(\hat{A}) = \sim L(A)$$

since $\cap, \sim$ generate all Boolean operations, ~~a~~ we can get $A \cup B$, $A \Delta B$, etc. as well.