That is, if given an NFA, how can we convert it to a regular expression?

Example:

Quickly we see that this corresponds to

$$a^* b(a \cup b)$$
Example 1.68
Consider the following DFA
It is not that easy to directly see what the regular expression is

We need a procedure shown below

First, add a start and an accept states
This generates a generalized NFA.

Our procedure is:

\[ \text{DFA} \rightarrow \text{GNFA} \rightarrow \text{regular expression} \]

Remove state 1.
Regular ⇒ a regular expression $V$

Example: the link $3 \rightarrow 2$
Thus $ba \cup a$

Remove state 2
Example:

\[ s \rightarrow 3 \]
Thus $a(aa \cup b)^* ab \cup b$
Regular \Rightarrow a regular expression IX

- Here we need to handle

$$2 \xrightarrow{aa \cup b} 2$$

- Remove state 3

$$(a(aa \cup b)^*ab \cup b)((ba \cup a)(aa \cup b)^*ab \cup bb)^*$$

$$(ba \cup a)(aa \cup b)^* \cup \epsilon \cup a(aa \cup b)^*$$

- We will formally explain the procedure
Definition of GNFA I

- Between any two states: a regular expression
- \((Q, \Sigma, \delta, q_{\text{start}}, q_{\text{accept}})\)

\[\delta : (Q - \{q_{\text{accept}}\}) \times (Q - \{q_{\text{start}}\}) \rightarrow R\]

- \(R\): all regular expressions over \(\Sigma\)
- DFA \(\rightarrow\) GNFA
  - Two new states: \(q_{\text{start}}, q_{\text{accept}}\)
  - \(q_{\text{start}} \rightarrow q_0\) with \(\epsilon\)
  - any \(q \in F \rightarrow q_{\text{accept}}\) with \(\epsilon\)
In the definition, between any two states there is an expression.

But what if in the graph two states are not connected?

\( \emptyset \in R \) so if no connection, we simply consider \( \emptyset \) as the expression between two states.
GNFA $\rightarrow$ regular expression I

- Fig 1.63

![Diagram]

- $q_i$
- $q_j$
- $q_{rip}$
- $R_1$
- $R_2$
- $R_3$
- $R_4$
GNFA $\rightarrow$ regular expression II

$q_i (R_1)(R_2)^* (R_3) \cup (R_4) \rightarrow q_j$

- $q_{rip}$ is the state being removed
- In the procedure
  - 3-state DFA $\rightarrow$ 5-state GNFA $\rightarrow$ 4-state $\cdots \rightarrow$
  - 2-state GNFA $\rightarrow$ regular expression
- In the procedure any any $(i, j)$ related to $q_{rip}$ considered
- Algorithm: convert(G)
  - $k$: # of G
If $k = 2$

return $R$

If $k > 2$, choose any $q_{\text{rip}} \in Q \setminus \{q_s, q_a\}$ for removal

$$Q' = Q \setminus \{q_{\text{rip}}\}$$

$$\forall q_i \in Q' \setminus \{q_{\text{accept}}\}, q_j \in Q' \setminus \{q_{\text{start}}\}$$

$$\delta'(q_i, q_j) = R_1 R_2^* R_3 \cup R_4,$$

where

$$R_1 = \delta(q_i, q_{\text{rip}}), \quad R_2 = \delta(q_{\text{rip}}, q_{\text{rip}}), \quad R_3 = \delta(q_{\text{rip}}, q_j), \quad R_4 = \delta(q_i, q_j)$$
Run convert($G'$), where

$$G' = (Q', \Sigma, \delta', q_s, q_a)$$

Why in the textbook we modify DFA to GNFA?
Is it ok to use NFA?
Seems ok??