Quick Review

• 5 tuple (Q, Σ, δ, q₀, F)
  \[ \Sigma_ε = \Sigma \cup \{\varepsilon\} \]
  \[ \delta : Q \times \Sigma_ε \rightarrow P(Q) \]

Every NFA has an equivalent DFA (Th 1.39)

We can convert from NFA to DFA
  NFAs are often easier to build
  DFAs are easier to code
Let’s define the \( \delta' \)

\[
\delta : Q \times \Sigma \rightarrow P(Q) \text{ in NFA}
\]

\[
\delta' : Q' \times \Sigma \rightarrow Q' \text{ in DFA}
\]

For \( R \in Q' \), \( a \in \Sigma \)

\[
\delta'(R, a) = \{ q \in Q | q \in \delta(r, a) \text{ some } r \in R \}
\]

Union of all sets that can be reached from a state in set \( R \) using the \( \delta \) with input \( a \)

\[
\delta'(R, a) = \bigcup_{r \in R} \delta(r, a)
\]
Converting NFA to DFA - \( \varepsilon \) Transitions

- Define start state and \( \delta' \) to include all states that can be reached from a given state by 0 or more \( \varepsilon \) transitions

\[
E(R) = \{ \; q \mid q \; \text{can be reached from } R \; \text{by using 0 or more } \varepsilon \; \text{transitions} \; \}
\]

\[
\delta'(R,a) = \{ \; q \in Q \mid q \in E(\delta(r,a)) \; \text{for some } r \in R \; \}
\]

\[
\delta'(R, a) = \{ \; q \in Q \mid q \in \delta(r, a) \; \text{some } r \in R \; \}
\]
Conversion Example (with δ)

NFA

DFA

Q = \{1, 2, 3\}
Σ = \{a, b\}
Q₀ = 1
F = \{1\}

Q' = \{Ø, 1\}
Σ' = \{a, b\}
Q₀' =
F' = 

NFA-DFA equivalence

• Th 1.25: Every NFA has an equivalent DFA
Corollary: A language is regular if and only if there exists an NFA that recognizes it

Proof:
If the language is regular, there exists a DFA that recognizes it. Each DFA is an NFA. Conversely, if there exists an NFA that recognizes the language, convert the NFA to a DFA.
Proof with NFAs

• Theorem 1.25: The class of regular languages is closed under the union operation.
  – We proved this using DFAs
    • What was the computation the new DFA simulated?
  – Is it any easier to prove using NFAs?
Proof with NFAs

• Regular languages are closed under concatenation
  this is where we stopped using DFAs
  what made this hard for DFAs?
Practice

• Construct an NFA to recognize concatenation of DFAs
  \[ A = \{ w \mid w \text{ contains at least three 1s} \} \]
  \[ B = \{ \varepsilon \} \]
Proof with NFAs

• Regular languages are closed under Kleene star

What is Kleene star?
Practice

• Construct an NFA to recognize Kleene star of $A$ if $A = \{ w \mid w$ contains at least two 0s and at most one 1 $\}$