Equivalence of DFA & NFA

Nondeterministic Finite Automata (NFA)

- A NFA has the power to be in several states at once
- This ability is often expressed as an ability to "guess" something about its input
- Each NFA accepts a language that is also accepted by some DFA
- NFA are often more succinct and easier than DFAs
- We can always convert an NFA to a DFA, but the latter may have exponentially more states than the NFA (a rare case)
- The difference between the DFA and the NFA is the type of transition function δ
  - For a NFA δ is a function that takes a state and input symbol as arguments (like the DFA transition function), but returns a set of zero or more states (rather than returning exactly one state, as the DFA must)

Equivalence of Deterministic and Nondeterministic Finite Automata

- Every language that can be described by some NFA can also be described by some DFA.
- The DFA in practice has about as many states as the NFA, although it has more transitions.
- In the worst case, the smallest DFA can have 2^n (for a smallest NFA with n state).

Proof: DFA can do whatever NFA can do

The proof involves an important construction called subset construction because it involves constructing all subsets of the set of stages of NFA.

From NFA to DFA

- We have a NFA $N = (Q_N, \Sigma, \delta_N, q_0, F_N)$
- The goal is the construction of a DFA $D = (Q_D, \Sigma, \delta_D, \{q_0\}, F_D)$ such that $L(D) = L(N)$.

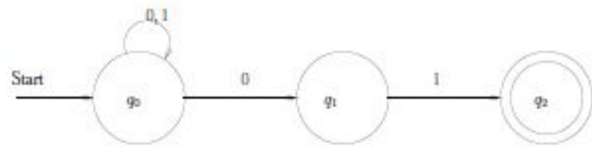
Subset Construction

- Input alphabets are the same.
- The start set in D is the set containing only the start state of N.
- Q_D is the set of subsets of Q_N, *i.e.*, Q_D is the power set of Q_N. If Q_N has n states Q_D will have 2^n states. Often, not all of these states are accessible from the start state.
- F_D is the set of subsets S of Q_N such that $S \cap F_N \neq \emptyset$. That is, F_D is all sets of N's states that include at least one accepting state of N.
- For each set $S \subseteq Q_N$ and for each input symbol $a \in \Sigma$

$$\delta_D(S, a) = \bigcup_{p \in S} \delta_N(p, a)$$

  To compute $\delta_D(S, a)$, we look at all the states p in S, see what states N goes from p on input a, and take the union of all those states.

Example



$Q_N = \{q_0, q_1, q_2\}$ then $Q_D = \{\emptyset, \{q_0\}, \{q_1\}, \{q_2\}, \{q_0, q_1\} \ldots\}$, *i.e.*, Q_D has 8 states (each one corresponding to a subset of Q_N)

		0	1
	$\emptyset$	$\emptyset$	$\emptyset$
$\rightarrow$	$\{q_0\}$	$\{q_0\ q_1\}$	$\{q_0\}$
	$\{q_1\}$	$\emptyset$	$\{q_2\}$
$\star$	$\{q_2\}$	$\emptyset$	$\emptyset$
	$\{q_0, q_1\}$	$\{q_0\ q_1\}$	$\{q_0, q_2\}$
$\star$	$\{q_0, q_2\}$	$\{q_0\ q_1\}$	$\{q_0\}$
$\star$	$\{q_1, q_2\}$	$\emptyset$	$\{q_2\}$
$\star$	$\{q_0, q_1, q_2\}$	$\{q_0\ q_1\}$	$\{q_0, q_2\}$

Example