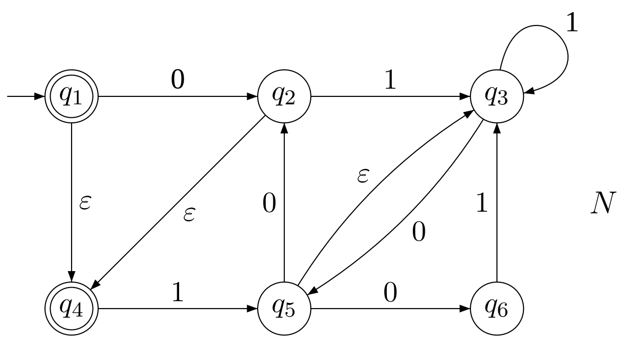
Handout 2: converting NFAs to DFAs

This handout explains the construction for obtaining an equivalent DFA from any given NFA. The most important thing to take away from this handout is the *idea* behind the construction—it is both easier and more important to understand the idea than to learn how to perform the construction as a mechanical procedure. If you focus on the idea, you should be able to reconstruct the method for yourself without difficulty.

Suppose N is a given NFA having alphabet Σ and set of states Q. Let us assume that the initial state of N is q_1. In order to use an example to aid in explaining the construction, let us assume that N is as follows:



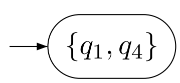
Our goal is to construct a DFA M such that $L(M) = L(N)$.

The main idea: the DFA M will need to keep track of *every possible state* that the NFA N could be in after every possible sequence of symbols. In order to do this, the states of M will correspond to *subsets of states* of N.

First step: The first step is to determine the starting state of M.

> *Draw an oval representing the start state of M. Label it with the subset of all states of N that you could get to from q_1 by following zero or more ε-transitions.*

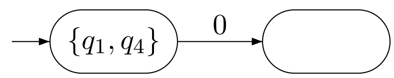
In our example, you can stay at q_1 or take an ε-transition to q_4, and that is it. The set $\{q_1, q_4\}$ therefore represents the set of all states of N you could possibly be in without reading any symbols. The diagram for M therefore starts out like this:



Middle steps: You need to repeat this step until your diagram for M represents a complete DFA, meaning that every state has a transition leading from it for each possible input symbol. Most of the work is in repeating this step. One iteration of this step works as follows.

> *Pick any symbol $\sigma \in \Sigma$ and any state in your diagram for M so that this state is missing a transition labeled by σ. Draw a transition from the state you chose to a new state, and label the transition with σ.*

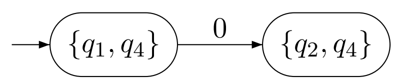
In our example, we have just one state of M so far, and it has no transitions leading out of it, so let's draw a transition for the symbol 0:



We still need to figure out how to label this new state with a subset of Q.

> *The subset you label this new state with includes every state that you can get to in N by following* **exactly one** *transition labeled by σ, and* **zero or more** *transitions labeled ε, starting from any state in N that is in the subset labeling the state of M you are starting from.*

In our example, this means that our new state will be labeled by the subset of Q containing all states in N that you can get to from either q_1 or q_4 by following a transition labeled 0 and any number of ε-transitions. By studying the diagram for N for a minute, we see that this subset is $\{q_2, q_4\}$, so we label the new state accordingly:

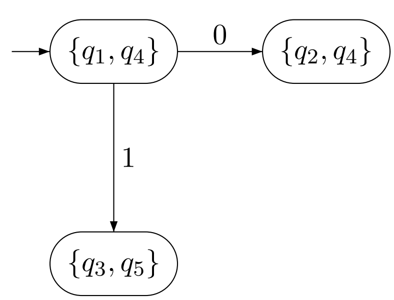


Figuring out exactly which states should label your new state can be annoying sometimes, especially for complicated examples. What you should keep in mind is (i) this is easy for a computer to do, and (ii) I won't ask you to perform this construction for complicated examples (because it is important that you understand the idea, not that you can behave like a computer and perform the construction by hand for complicated NFAs).

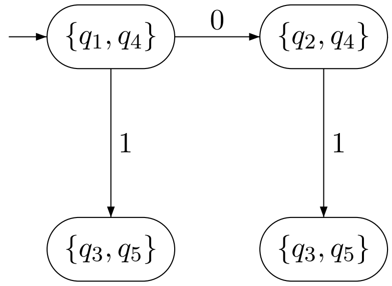
Obviously the process has to stop somehow—when you get a state labeled by a subset that is already in your diagram, you can simplify:

> *If the subset labeling your new state is already in your diagram, erase the state and instead draw the new transition to the existing state.*

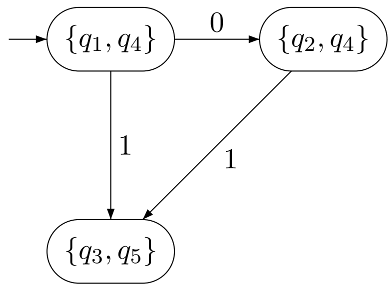
Let us continue with our example for a few steps to see how this goes. Suppose we now pick the same state labeled $\{q_1, q_4\}$, and deal with the transition out of it labeled with the symbol 1. We add a state as before, and label the state appropriately:



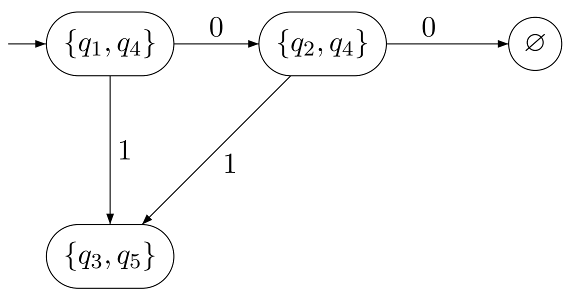
Let us repeat the process for another transition. To illustrate the process of simplifying, suppose we look at the transition leading from the state $\{q_2, q_4\}$ labeled by 1. We would obtain this diagram:



We need to simplify because there are now two states labeled the same way. We erase the new state and draw the transition instead to the old state:

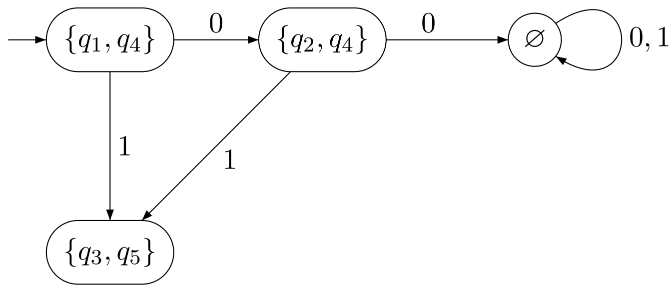


Sometimes you may get the empty set as a state in M. This is not a problem—you just treat it like any other subset. For example, if we look at the transition out of $\{q_2, q_4\}$ labeled by 0, we get this:

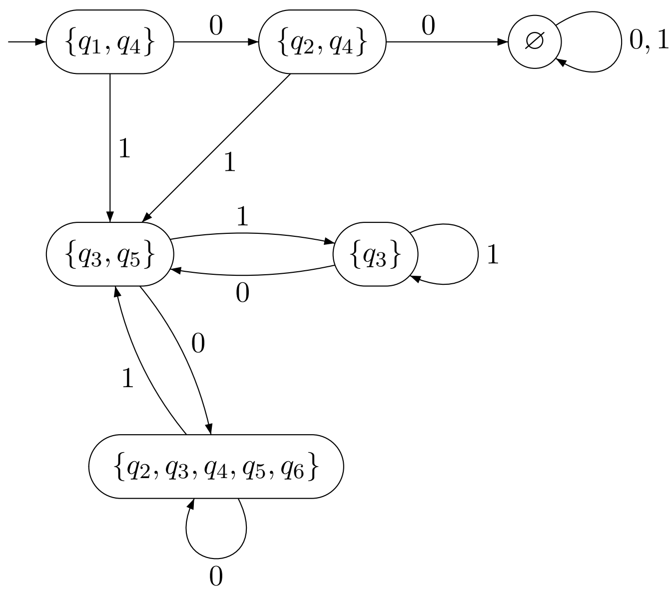


This is because you cannot get to any states if you start at q_2 or q_4 and follow a 0 transition.

It is easy to deal with the transitions out of a state labeled by the empty set—it is always a self-loop. That's because you can't get anywhere if you can't start anywhere:



If you keep repeating this process, you will eventually get this diagram:

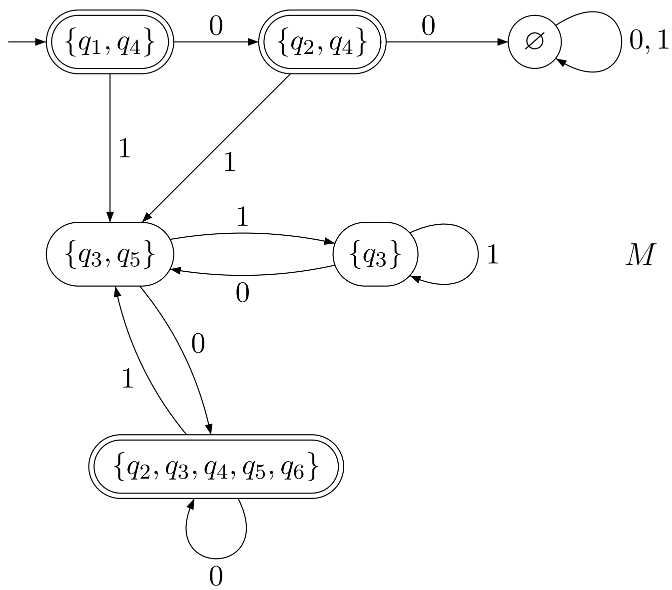


Every state has a transition for each symbol, implying that we cannot continue the process. So, we proceed to the last step.

Final step: It remains to determine which states of our new DFA will be accept states, which is fortunately very simple.

> *Each state in M is an accept state if and only if it is labeled by a subset that includes an accept state of N.*

In our example, the accept states of N are q_1 and q_4, so the accept states of our new DFA M are as indicated in the final diagram:



Unless we have made errors, it will be the case that $L(M) = L(N)$. (It is always a good idea to try the DFA on some sample input strings to decrease the chance of letting an error slip by.)

Don't forget the idea! To illustrate the idea one more time, consider a sample input string, such as 1110. Following the transitions for 1110 in the above diagram, we reach the state labeled $\{q_3, q_5\}$. This means that if you were to follow any transitions you wanted in N that were legal given the input 1110, the possible states you could be in would be precisely q_3 or q_5. This is a reject state, given that neither of q_3 or q_5 is an accept state—there is no way you could land on an accept state of N after reading 1110. On the other hand, after reading the string 0100 for instance, it would be possible for you to be in any of the states $\{q_2, q_3, q_4, q_5, q_6\}$ in N. In particular, you could be on state q_4, so this string must be accepted by M.

In our example, we were lucky—the DFA M only has 6 states. In general, the number of states can grow exponentially with the number of states of N. Such is life: although the DFA you get might not be the smallest possible DFA for $L(N)$, it can be proved that this sort of blow-up in the number of states is unavoidable in some situations.