CS 360
Naomi Nishimura

State elimination

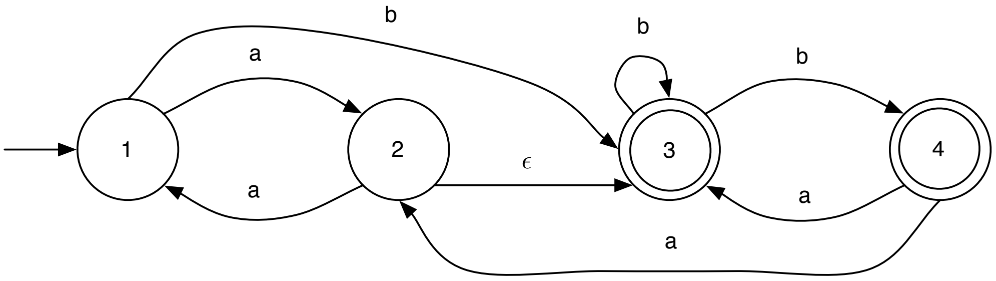
Note: this information is meant to cover some material used in class but absent from the textbook. This is not intended to be comprehensive or a replacement for attending lecture.

This handout is based on material developed by Jeff Shallit for CS 360, in turn based on material developed by Eric Bach of the University of Wisconsin.

In Section 3.2.2 of the textbook, an algorithm is given for constructing a regular expression from a DFA. The algorithm presented here (and in class) is simpler to understand, and applies to NFA's and ϵ-NFA's as well.

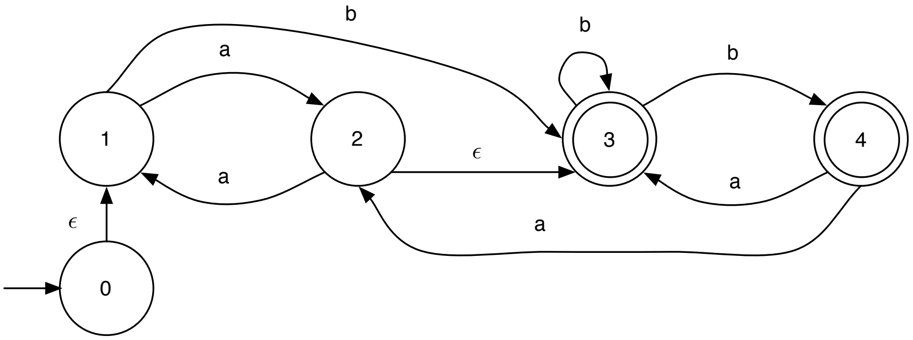
As in the textbook, we will remove states from the automaton, replacing labels of arcs, so that in the end a single regular expression is formed. The single regular expression will be the label on an arc that goes from the start state to the accepting state, and this will be the only arc in the automaton.

The algorithm forms the simpler automaton as follows. In step 1, we modify the automaton to have a start state that is not an accepting state and has no transitions in (either self-loops or from other states). In step 2, we create an equivalent automaton that has a single accepting state with no transitions out. These will be the two states that remain at the end of the algorithm. In step 3, the other states are eliminated, in any order. Details of the algorithm follow, along with a running example, illustrated below.



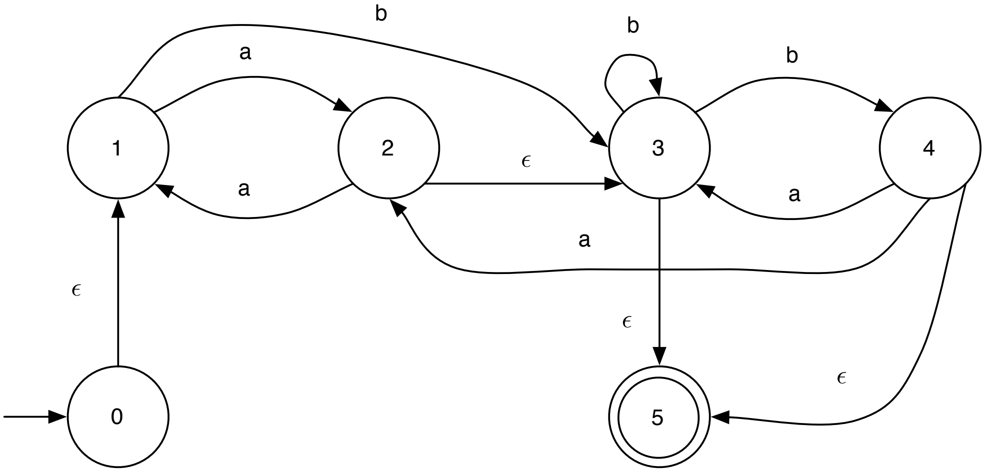
Step 1

If the start state is an accepting state or has transitions in, add a new non-accepting start state and add an ϵ-transition between the new start state and the former start state.



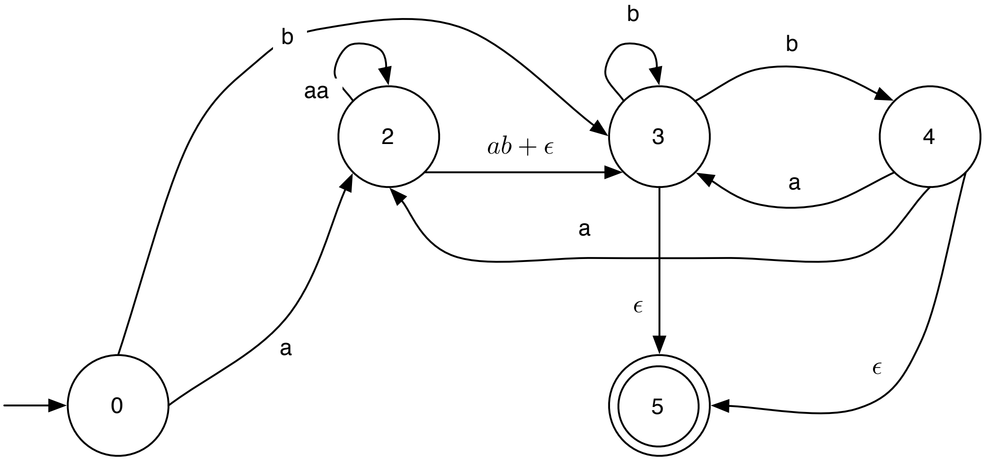
Step 2

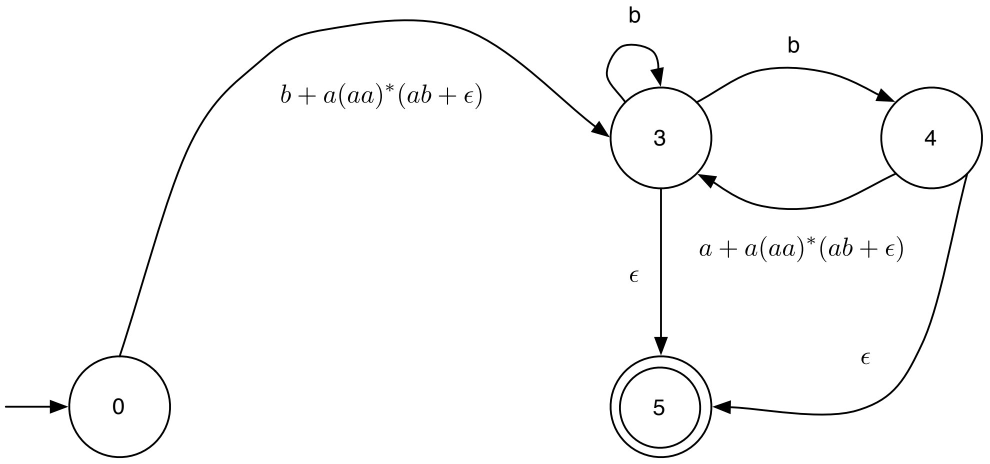
If there is more than one accepting state or if the single accepting state has transitions out, add a new accepting state, make all other states non-accepting, and add an ϵ-transition from each former accepting state to the new accepting state.

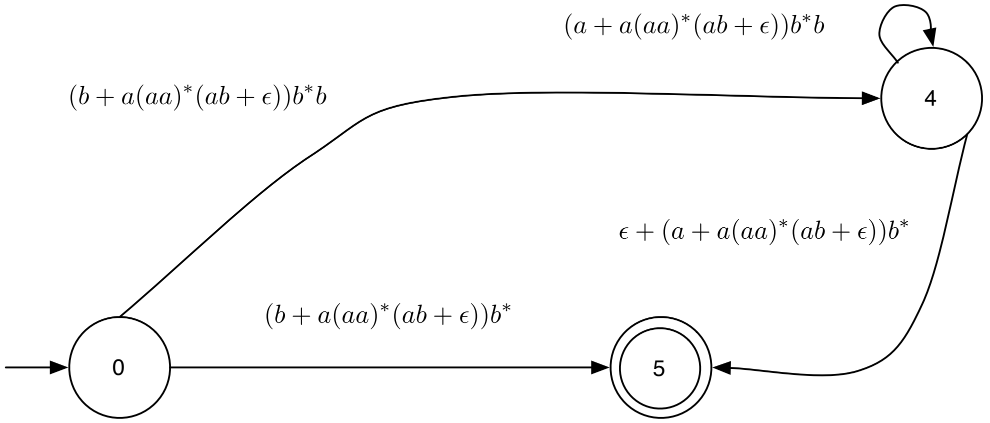


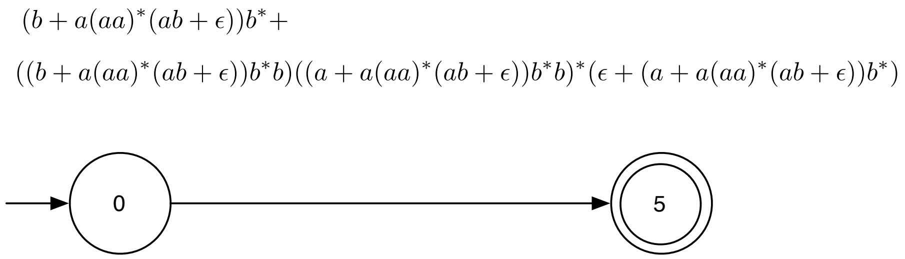
Step 3

For each non-start non-accepting state in turn, eliminate the state and update transitions according to the procedure given on page 99 of the textbook, Figures 3.7 and 3.8. The following illustrations depict the removal of states 1, 2, 3, and 4 in that order.




b
b
$b + a(aa)^*(ab + \epsilon)$
3
4
$a + a(aa)^*(ab + \epsilon)$
ϵ
ϵ
0
5


$(a + a(aa)^*(ab + \epsilon))b^*b$
$(b + a(aa)^*(ab + \epsilon))b^*b$
4
$\epsilon + (a + a(aa)^*(ab + \epsilon))b^*$
$(b + a(aa)^*(ab + \epsilon))b^*$
0
5


$(b + a(aa)^*(ab + \epsilon))b^* +$
$((b + a(aa)^*(ab + \epsilon))b^*b)((a + a(aa)^*(ab + \epsilon))b^*b)^*(\epsilon + (a + a(aa)^*(ab + \epsilon))b^*)$
0
5