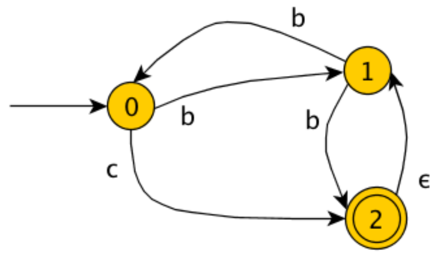
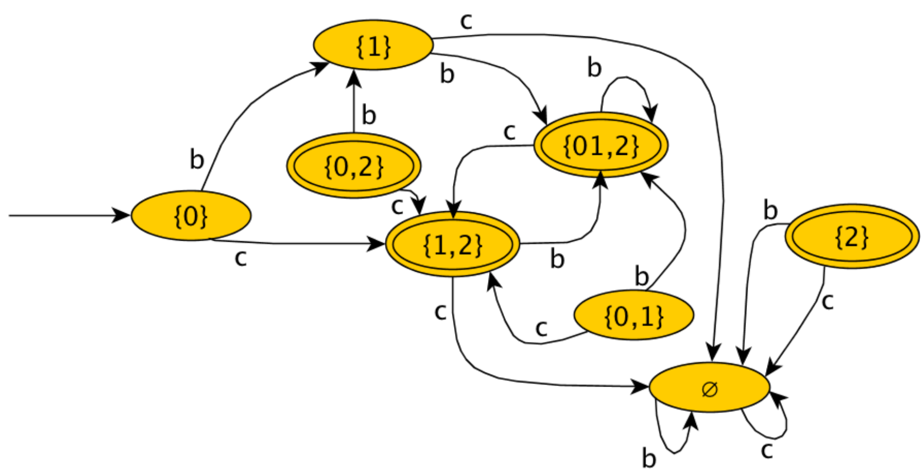
Naive subset construction

We start with this machine and a symbol set of {'b', 'c'}



The naive subset construction algorithm makes a machine with 8 states and 16 transitions like this



In general the number of states is 2^n where n is the number of states in the DFR and the number of transitions is $m2^n$ where m is the size of the symbol set.
Note that 3 of the 8 states can not be reached from the start state.
Compare the recognition algorithm on these two machines for string "bbc".

NDFR. We start with $R = \epsilon$-closure$(q_{\text{start}}) = \epsilon$-closure$(0) = \{0\}$

R	a	$\delta(R,a)$	ϵ-closure$(\delta(R,a))$	Accept
$\{0\}$	b	$\{1\}$	$\{1\}$	
$\{1\}$	b	$\{0,2\}$	$\{0,1,2\}$	
$\{0,1,2\}$	c	$\{2\}$	$\{1,2\}$	
$\{1,2\}$				Yes

Now consider the second machine and the deterministic version of the recognition algorithm.

r	a	$\delta(r,a)$	Accept
$\{0\}$	b	$\{\{1\}\}$	
$\{1\}$	b	$\{\{0,1,2\}\}$	
$\{0,1,2\}$	c	$\{\{1,2\}\}$	
$\{1,2\}$			Yes

Now consider the string "cc". First the NDFR.

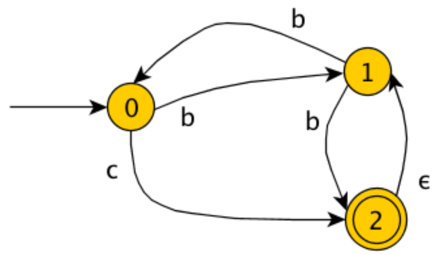
R	a	$\delta(R,a)$	ϵ-closure$(\delta(R,a))$	Accept
$\{0\}$	c	$\{2\}$	$\{1,2\}$	
$\{1,2\}$	c	$\emptyset$	$\emptyset$	
$\emptyset$				No

And the DFR using the deterministic algorithm

r	a	$\delta(r,a)$	Accept
$\{0\}$	c	$\{\{1,2\}\}$	
$\{1,2\}$	c	$\{\emptyset\}$	
$\emptyset$			No

Subset construction

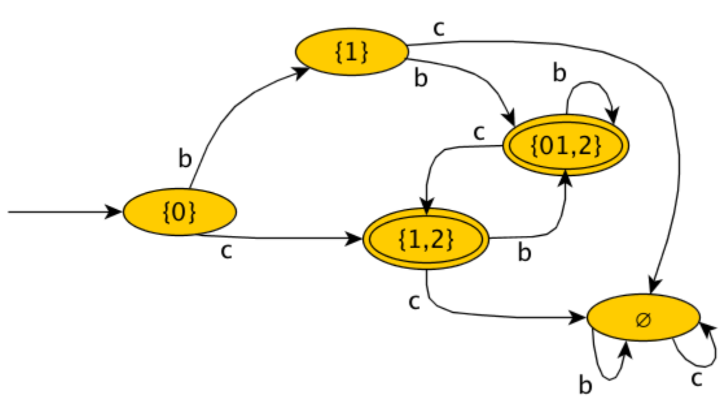
Start with this NDFR



We trace the algorithm. The workset W is the set of states discovered to be reachable but not yet processed. The last column indicates whether the destination state is a new state (not yet processed and not already in W).

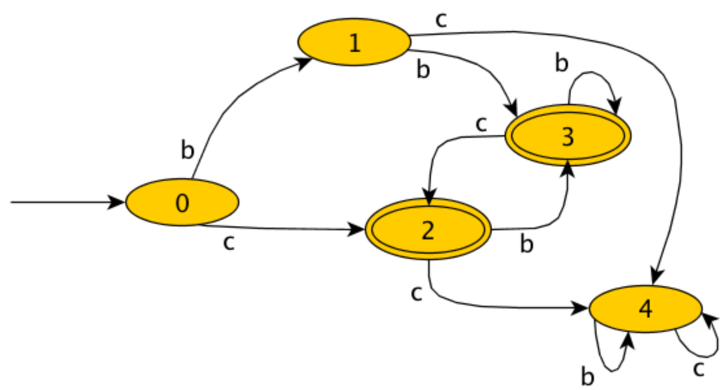
W	$\dot{q}$	a	Transitions	Final?	New?
$\{\{0\}\}$	$\{0\}$			no	
		‘b’	$(\{0\}, \text{‘b’}, \{1\})$		yes
		‘c’	$(\{0\}, \text{‘c’}, \{1,2\})$		yes
$\{\{1\}, \{1,2\}\}$	$\{1\}$			no	
		‘b’	$(\{1\}, \text{‘b’}, \{0,1,2\})$		yes
		‘c’	$(\{1\}, \text{‘c’}, \emptyset)$		yes
$\{\{1,2\},$	$\{1,2\}$			yes	
$\{0,1,2\}, \emptyset\}$		‘b’	$(\{1,2\}, \text{‘b’}, \{0,1,2\})$		no
		‘c’	$(\{1,2\}, \text{‘c’}, \emptyset)$		no
$\{\{0,1,2\}, \emptyset\}$	$\{0,1,2\}$			yes	
		‘b’	$(\{0,1,2\}, \text{‘b’}, \{0,1,2\})$		no
		‘c’	$(\{0,1,2\}, \text{‘c’}, \{1,2\})$		no
$\{\emptyset\}$	$\emptyset$			no	
		‘b’	$(\emptyset, \text{‘b’}, \emptyset)$		no
		‘c’	$(\emptyset, \text{‘c’}, \emptyset)$		no
$\emptyset$					

In a picture, the DFR machine is



As you can see this is the same as the machine we got from

the naive subset construction algorithm, restricted to the states that are reachable from the start state.
After renaming the states to numbers we get



We can represent the transitions with a state transition matrix

	b	c
0	1	2
1	3	4
2	3	4
3	3	2
4	4	4