Pakistan Institute of Engineering & Applied Sciences Fall 2008
CIS 317: Theory of Automata

Equivalence and Minimization of DFAs

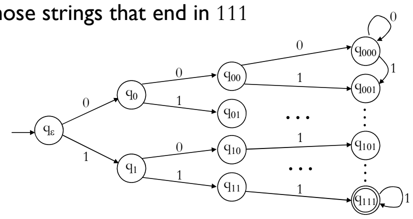
Umar Faiz
http://www.pieas.edu.pk/umarfaiz/cis317

DFA minimization

- Questions of DFA size:
  - Given a DFA, can we find one with fewer states that accepts the same language?
  - What is the smallest DFA for a given language?
  - Is the smallest DFA unique, or can there be more than one "smallest" DFA for the same language?

Example

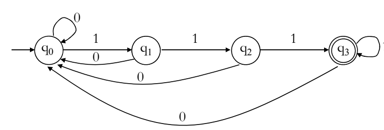
- Construct a DFA over alphabet $\{0, 1\}$ that accepts those strings that end in 111



- This is big, isn't there a smaller DFA for this?

Smaller DFA

- Yes, we can do it with 4 states:

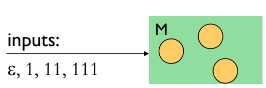


  - The state remembers the number of consecutive 1s at the end of the string (up to 3)
- Can we do it with 3 states?

Even smaller DFA?

- Suppose we had a 3 state DFA M for L
- We do not know what this M looks like

… but let's imagine what happens when:



- By the pigeonhole principle, on two of these inputs M ends in the same state

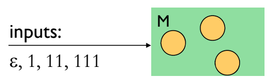
Pigeonhole principle

Suppose you are tossing m balls into n bins, and $m > n$. Then two balls end up in the same bin.

- Here, balls are inputs, bins are states:

If you have a DFA with n states and you run it on m inputs, and m > n, then two inputs end up in same state.

A smaller DFA?



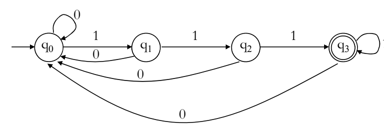
- What goes wrong if…
  - M ends up in same state on input 1 and input 111?
  - M ends up in same state on input ε and input 11?

No smaller DFA!

- Conclusion

  There is no DFA with 3 states for L

- So, this DFA is minimal

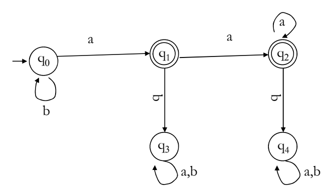


  - In fact, it is the unique minimal DFA for L

DFA minimization

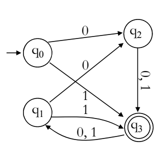
- There is an algorithm to start with any DFA and reduce it to the smallest possible DFA
- The algorithm attempts to identify classes of equivalent states
- These are states that can be merged together without affecting the answer of the computation

Examples of equivalent states

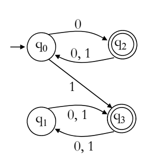


In both q_3 and q_4, the machine rejects, no matter what the rest of the input string contains
They're equivalent and can be combined…

Examples of equivalent states



q_0, q_1 equivalent



q_0, q_1 equivalent

q_2, q_3 also equivalent

Equivalent and distinguishable states

- Two states q, q' are equivalent if

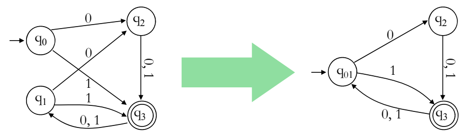
  For every string w, the states $\hat{d}(q, w)$ and $\hat{d}(q', w)$ are either both accepting or both rejecting

  - Here, $\hat{\delta}(q, w)$ is the state that the machine is in if it starts at q and reads the string w

- q, q' are distinguishable if they are not equivalent:

  For some string w, one of the states d(q, w), d(q', w) is accepting and the other is rejecting

Examples of distinguishable states



q_3 distinguishable from q_0, q_1, q_2

q_3 is accepting, others are rejecting

(q_0, q_2) and (q_1, q_2) distinguishable

they behave differently on input 0

q_0, q_1 equivalent

DFA minimization algorithm

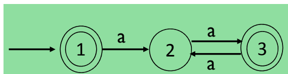
- Find all pairs of distinguishable states as follows:

① For any pair of states q, q':
If q is accepting and q' is rejecting
Mark (q, q') as distinguishable

② Repeat until nothing is marked:
For any pair of states (q, q'):
For every alphabet symbol a:
If (δ(q, a), δ(q', a)) are marked as distinguishable
Mark (q, q') as distinguishable

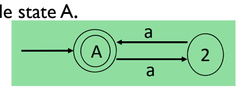
③ For any pair of states (q, q'):
If (q, q') is not marked as distinguishable
Merge q and q' into a single state

Example (1) of DFA minimization [Method 1]

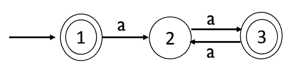


2	1	
3		1
	1	2

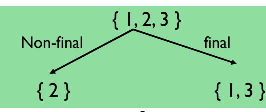
- The boxes (1,2) and (2,3) are marked in the first pass.
- In pass 2 no boxes are marked because, δ (1,a) → 2 and δ (3,a) → 2. That is (1,3) → (2,2), where 2 and 2 are not distinguishable.
- δ(1,b) → ϕ and δ (3,b) → ϕ. That is (1,3) → (ϕ, ϕ), where ϕ is a non final state and not distinguishable. It implies that (1,3) are equivalent and can replaced by a single state A.



Example (1) of DFA minimization [Method 2]

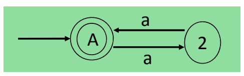


2	1	
3		1
	1	2

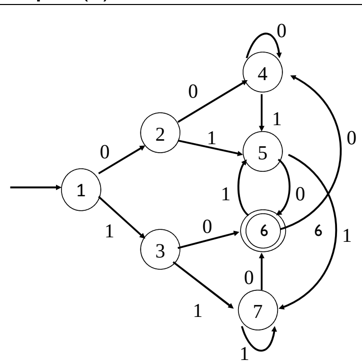


Consider set {1,3}. (1,3) $\xrightarrow{a}$ (2,2) and (1,3) $\xrightarrow{b}$ (ϕ, ϕ).

- This implies state 1 and 3 are equivalent and can not be divided further. This gives us two states 2,A.



Example (2) of DFA minimization [Method 1]



Example (2) of DFA minimization [Method 1]

2						
3						
4						
5						
6	x	x	x	x	x	
7						x
	1	2	3	4	5	6

In pass 1,
(1,6) (2,6)
(3,6) (4,6)
(5,6) (6,7)
are distinguishable

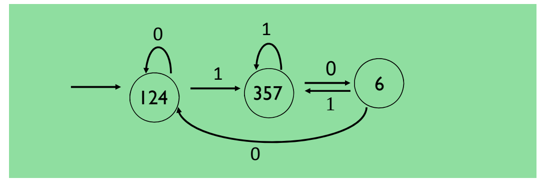
Example (2) of DFA minimization [Method 1]

2						
3	x	x				
4			x			
5	x	x		x		
6	x	x	x	x	x	
7	x	x		x		x
	1	2	3	4	5	6

In pass 2,
(3,1) $\xrightarrow{0}$ (6,2)
(3,2) $\xrightarrow{0}$ (6,4)
(5,1) $\xrightarrow{0}$ (6,2)
(5,2) $\xrightarrow{0}$ (6,4)
(7,4) $\xrightarrow{0}$ (6,4)
(7,2) $\xrightarrow{0}$ (6,4)
and so on.

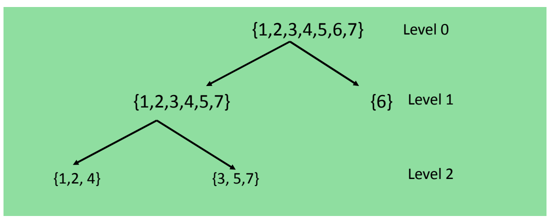
The pairs marked 2 are those marked on the second pass.

Example (2) of DFA minimization [Method 1]



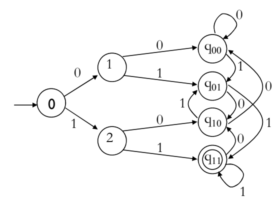
The states 1, 2, and 4 are equivalent and can be replaced by a single state 124. The states 3, 5, and 7 are equivalent can be replaced by the single state 357. The state is distinguishable. The resultant minimal FA is shown in the figure

Example (2) of DFA minimization [Method 2]



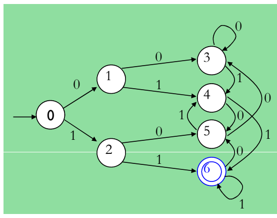
(2,3) $\xrightarrow{0}$ (4,6) this implies that 2 and 3 belongs to different group hence they are split in level 2. The same hold for the pairs (4,5) (1,7) and (2,5) and so on.

Example (3) of DFA minimization



1						
2						
q_{00}						
q_{01}						
q_{10}						
q_{11}						
	0	1	2	q_{00}	q_{01}	q_{10}

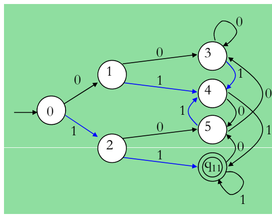
Example (3) of DFA minimization



1						
2						
3						
4						
5						
6	x	x	x	x	x	x
	0	1	2	3	4	5

6 is distinguishable from all other states

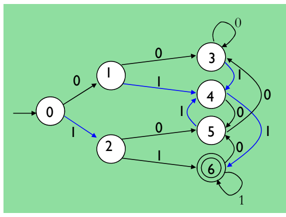
Example (3) of DFA minimization



1						
2	x	x				
3			x			
4						
5			x			
q_{11}	x	x	x	x	x	x
	0	1	2	3	4	5

2 is distinguishable from $0, 1, 3, 5$
On transition 1, they go to distinguishable states

Example (3) of DFA minimization

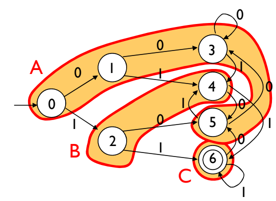


1						
2	x	x				
3			x			
4	x	x		x		
5			x		x	
6	x	x	x	x	x	x
	0	1	2	3	4	5

4 is distinguishable from 0, 1, 3, 5

On transition 1, they go to distinguishable states

Example (3) of DFA minimization



1						
2	x	x				
3			x			
4	x	x		x		
5			x		x	
5	x	x	x	x	x	x
	0	1	2	3	4	5

Merge states not marked distinguishable

0, 1, 3, 5 are equivalent → group A

2, 4 are equivalent → group B

6 cannot be merged → group C