531 - 2013-02-12

Note Title 2013-02-12

Conversion of NFAs to DFAs. [Mogensen, Ch. 2]

ε-closure as a special case of solving set equations

$\varepsilon\text{-closure}(M) = M \cup \{t \mid s \in \varepsilon\text{-closure}(M) \text{ and } s^{\varepsilon}t \in T\}$

$F_M(X) = M \cup \{t \mid s \in X \text{ and } s^{\varepsilon}t \in T\}$. With this defn to solve the ε-closure set equation, we just need to solve

$$X = F_M(X)$$

The F_M function is monotonic.

$$F_M(X) \subseteq F_M(Y) \text{ if } X \subseteq Y.$$

The least solution S to the equation $X = F(X)$, when F is monotonic satisfies $S = F(S)$.

(We say that the solution S is a fixpoint of the set equation)

Since $\emptyset \subseteq S$ (where S is the solution),

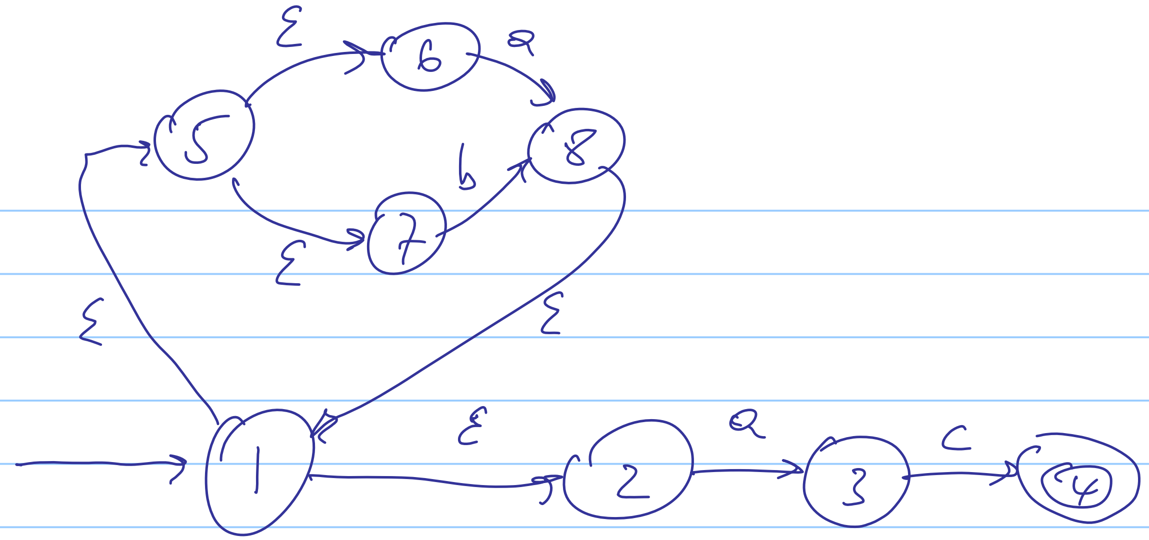
$$F(\emptyset) \subseteq F(S) = S.$$

We therefore can start by guessing that $\emptyset$ is a solution of $X = F(X)$. If it is, we are done; otherwise, we try $F(\emptyset)$ as a new guess; and then we continue, Building the chain

$\emptyset \subseteq F(\emptyset) \subseteq F(F(\emptyset)) \subseteq \cdots$ until two successive elements of the chain are equal, i.e. until a fixpoint is reached.

Example (Fig. 2.5 in Mogensen's)

NFA that recognizes $(a|b)^* ac$



We want ε-closure $(\{1\})$, so $M = \{1\}$, $F_M = F_{\{1\}}$

We start by guessing $F_M(\phi) = F_{\{1\}}(\phi)$

$$F_{\{1\}}(\phi) = \{1\} \cup \{t \mid s \in \phi \text{ and } s \xrightarrow{\varepsilon} t\} = \{1\}$$

So, ϕ is not a solution, and we continue

$$F_{\{1\}}(\{1\}) = \{1\} \cup \{t \mid s \in \{1\} \text{ and } s^{\varepsilon}t \in T\} =$$

$$= \{1\} \cup \{2, 5\} = \{1, 2, 5\}$$

So, $\{1\}$ is not a solution, and we continue

$$F_{\{1\}}(\{1, 2, 5\}) = \{1\} \cup \{t \mid s \in \{1, 2, 5\} \text{ and } s^{\varepsilon}t \in T\} =$$

$$\{1\} \cup \{2, 5, 6, 7\} = \{1, 2, 5, 6, 7\}$$

So, $\{1, 2, 5\}$ is not a solution, and we continue

$F_{\{1\}}(\{1,2,5,6,7\}) = \{1\} \cup \{t \mid s \in \{1,2,5,6,7\} \text{ and } s \xrightarrow{\varepsilon} t \in T\} =$

$= \{1\} \cup \{2,5,6,7\} = \{1,2,5,6,7\}$

So, $\{1,2,5,6,7\}$ is the solution, i.e., the ε-closure$(\{1\})$.

We can make this algorithm more efficient by noticing that the ε-closure function is distributive, i.e., it has the property that $F(X \cup Y) = F(X) \cup F(Y)$.

$$F_{\{1\}}(\phi) = \{1\} \cup \{t \cdots\} = \{1\}$$

$$F_{\{1\}}(\{1\}) = \{1\} \cup \{t \cdots\} = \{1\} \cup \{2,5\} = \{1,2,5\}$$

$$F_{\{1\}}(\{1,2,5\}) = F(\{1\}) \cup F(\{2,5\}) = \{1,2,5\} \cup \{1\} \cup$$

$$\{t \mid s \in \{1,2,5\} \text{ and } s \xrightarrow{\varepsilon} t \in T\} = \{1,2,5\} \cup \{1\} \cup$$

$$\cup \{6,7\} = \{1,2,5,6,7\}$$

$$F_{\{1\}}(\{1,2,5,6,7\}) = F_{\{1\}}(\{1,2,5\}) \cup F_{\{1\}}(\{6,7\}) =$$

$$= \{1,2,5,6,7\} \cup \{1\} \cup \{t \mid s \in \{6,7\} \text{ and } s \xrightarrow{\varepsilon} t \in T\} =$$

$= \{1,2,5,6,7\} \cup \{1\} \cup \{\} = \{1,2,5,6,7\}.$

Distributivity allows to refine the algorithm by using a worklist.

The worklist for the previous example (i.e. ε-closure$(\{1\})$),

$\{1\}$

$\{\check{1}, 2, 5\}$

$\{\check{1}, \check{2}, 5\}$

$\{1, 2, 5, 6, 7\}$

$\{1, 2, 5, 6, 7\}$

$\{1, 2, 5, 6, 7\}$ done

Now the algorithm to construct a DFA from an NFA.

Algorithm 2.3 (The subset construction).

NFA	N	$\overset{\text{equiv.}}{\Longleftrightarrow}$	DFA	D
states	S			S'
starting state	s_0			s_0'
accepting states	$F \subseteq S$			F'
alphabet	Σ		alphabet	Σ (same)

(partial)

transition relation   move function

a DFA state is a set of NFA states

$s_0' = \varepsilon\text{-closure}(\{s_0\})$

$\text{move}(s', c) = \varepsilon\text{-closure}(\{t \mid s \in s' \text{ and } s \xrightarrow{c} t \in T\})$

$S' = \{s_0'\} \cup \{\text{move}(s', c) \mid s' \in S', c \in \Sigma\}$

a set function, to be solved in the same way as the ε-closure function

$$F' = \{ s' \in S' \mid s' \cap F \neq \emptyset \}$$

$\square$

Example: again, we use the NFA of Figure 2.5 (see above) that recognizes $(a|b)^* ac$

The initial state of the DFA is:

$$s_0' = \varepsilon\text{-closure}(\{s_0\}) = \varepsilon\text{-closure}(\{1\}) = \{1, 2, 5, 6, 7\}$$

To compute move, we use the worklist procedure

Start with the worklist (i.e., the uncompleted set of states of the DFA) $S' = \{s_0'\}$

$$\text{move}(s_0', a) = \varepsilon\text{-closure}(\{t \mid s \in \{1,2,5,6,7\} \text{ and } s \xrightarrow{a} t \in T\}) =$$

$$= \varepsilon\text{-closure}(\{3,8\}) =$$

$$= \{3,8,1,2,5,6,7\}$$

$$= s_1'$$

$move(S_0', b) = \ldots = \{8, 1, 2, 5, 6, 7\} = S_2'$

$move(S_0', c) = \varepsilon\text{-}closure(\{t \mid s \in \{1,2,5,6,7\} \text{ and } s \xrightarrow{c} t \in T\}) =$

$= \varepsilon\text{-}closure(\{\}) = \{\}$

The empty set of NFA states is not a DFA state, so there is no transition from S_0' on c.

Now, our worklist (incomplete set of DFA states), S', is

$\checkmark$

$\{ s_0', s_1', s_2' \}$

We pick s_1' and calculate its transitions:

$move(s_1', a) = - - - - = s_1'$

$move(s_1', b) = \cdot \cdot \cdot -- = s_2'$

$move(s_1', c) = \varepsilon\text{-}closure(\{t \mid s \in \{3, 8, 1, 2, 5, 6, 7\} \text{ and } s \xrightarrow{c} t \in T\} =$

$\{4\} = s_3'$

The work list changes to

$\{ \checkmark S_0', \checkmark S_1', S_2', S_3' \}$

We pick S_2' and compute

$move(S_2', a) = \cdots = S_1'$

$move(S_2', b) = \cdots = S_2'$

$move(S_2', c) = \cdots = \{\}$

The worklist is $\{ \checkmark S_0', \checkmark S_1', \checkmark S_2', S_3' \}$.

We pick s_3' and compute

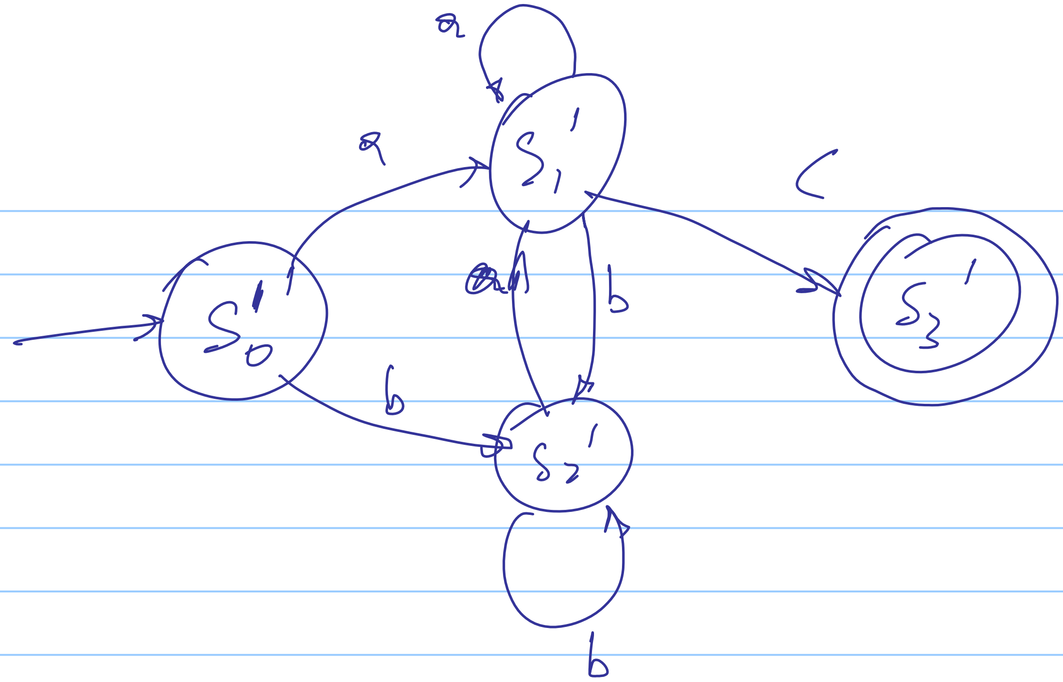
$move(s_3', a) = \{\}$

$move(s_3', b) = \{\}$

$move(s_3', c) = \{\}$

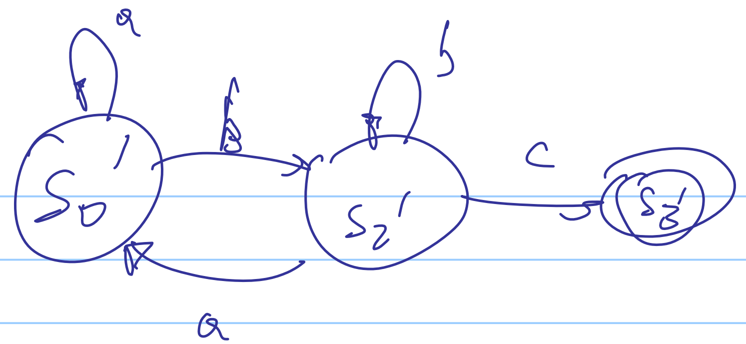
No new state to the added to the worklist.

All elements of the worklist are checked - Done.



$F' = \{s_3'\}$

$\Downarrow$ s_0' and s_1' are equivalent



Minimality!