CS310 Automata Theory – 2016-2017

Nutan Limaye

Indian Institute of Technology, Bombay
nutan@cse.iitb.ac.in

Lecture 13: Extensions of DFA/NFAs
January 31, 2017

Last class

Decision problems on DFA/NFAs.

The minimization problem for DFAs.

End of Module - I

Diffrent models of computation: 2DFA.

Module - II: Different models of computation

What do we plan to do in this module?

2DFA, a variant of a DFA where the input head moves right/left.

Chapter 18, from the text of Dexter Kozen

Pushdown automata, context-free languages(CFLs), context-free grammar(CFG), closure properties of CFLs.

Module - II: Different models of computation

2DFA: Two-way deterministic finite state automata.

$$\boxed{\# \;\; w_1 \;\; w_2 \;\; \ldots \;\; \ldots \;\; \ldots \;\; \ldots \;\; \ldots \;\; \ldots \;\; w_n \;\; \$}$$

Input head moves left/right on this tape.

It does not go to the left of $\#$.

It does not go to the right of \$.

Can potentially get stuck in an infinite loop!

Formal definition of 2DFA

Definition

A 2DFA $A = (Q, \Sigma \cup \{\#, \$\}, \delta, q_0, q_{acc}, q_{rej})$, where

Q :	set of states,	Σ:	input alphabet
$\#$:	left endmarker	$\$$:	right endmarker
q_0:	start state		
q_{acc}:	accept state	q_{rej}:	reject state

$$\delta : Q \times (\Sigma \cup \{\#, \$\} \to Q \times \{L, R\}$$

The following conditions are forced:

$\forall q \in Q$, $\exists q', q'' \in Q$ s.t. $\delta(q, \#) = (q', R)$ and $\delta(q, \$) = (q'', L)$.

2DFA: Two-way deterministic finite state automata

Examples

Let $\Sigma = \{a, b\}$ and L be a regular language.

$L_1 = \{w \in \Sigma^* \mid \text{second letter from the end if } a\}$.

$L_2 = \{w \in \Sigma^* \mid w \cdot w \in L\}$

$L_2 = \{w \in \Sigma^* \mid w^{\leq |w|} \in L\}$

Acceptance by 2DFA

Definition

Let A be a 2DFA.

A word w is said to be accepted by A if A reaches q_{acc} on w.

A word w is said to be rejected by A if A reaches q_{rej} on w.

A is said to recognize a language L if $\forall w \in L$, A reaches q_{acc}.

2DFA may loop forever if $w \notin L$ or may enter q_{rej}.