CPSC 421: Introduction to Theory of Computing	**Winter Term 1 2018-19**

Lecture 13: October 3

Lecturer: Nicholas Harvey *Scribes: Kaitian Xie*

13.1 Multitape TM

Multitape TM example: $L = \{x\#x : x \in \Sigma^*\}$

Let's do an implementation-level description of a multi-tape TM for L. Our original TM took time $\Theta(n^2)$ for inputs of length n.

1. Scan head 1 to the right until it reads a #. Move Right. (Second head is still at start of tape 2)
2. Repeatedly read symbol from tape 1, write it to tape 2, move both heads right, until seeing a blank on tape 1. Now second half of input is on tape 2.
3. Move both heads left until they reach start of tapes (possibly using \$ back to find start of tape). Replace # by ␣ while doing so.
4. Repeat until ␣ on both tapes. If symbols differ, reject. Else, move both heads right.

The multi-tape TM runs in $\Theta(n)$ time!

13.2 Nondeterministic Turing Machines

Last time: Configuration of TM, aqb, $a, b \in \Gamma^*$, $q \in Q$

Acceptance of a NTM: Input string is accepted if $\exists$ configurations $c_0, c_1, \cdots, c_k$ where:

- $c_0 = q_{start}\ w$
- $c_i \Rightarrow c_{i+1}$ (c_{i+1} is a possible configuration from c_i following the transition function δ)
- c_k is on the accepting state

i.e. in a tree of configs, is there an accepting state:

1. w is accepted: any node in tree is an accepting state
2. w is explicitly rejected: the tree is finite, but yet no node is accepting config i.e. all leaves are rejection configs
3. The NTM runs forever on w: the tree is infinite, but no node is accepting config

Definition 13.1 *A NTM is a decider if for all inputs, case 1 or 2 happens.*

Example of NTMs: Let L_1, L_2 be recognizable languages. Let M_i be a TM that recognizes L_i.

Claim 13.2 *$L_1 \cup L_2$ is recognizable. A cheat! Let's use nondeterminism.*

Definition 13.3 *Define a NTM M as follows:*

- *Nondeterministically choose to do one of the following:*
  - *Run M_1*
  - *Run M_2*

Claim 13.4 *M recognizes $L_1 \cup L_2$.*

Proof: Suppose $w \in L_1 \cup L_2$. Say $w \in L_2$. Then the branch of tree simulating M_i will accept. So M is in case 1, and accepts. If $w \notin L_1 \cup L_2$, then both $M_1 \& M_2$ either run forever or reject. So M is in either case 2 and case 3. So M does not accept w. ■

Theorem 13.5 *Given a NTM M, we can construct a DTM M s.t. $L(M) = L(M')$.*

Theorem 13.6 *If M is a NTM decider, then we can make M' a decider as well.*

Using Theorem, we get a DTM M', completing proof of claim 1.