Undergrad Complexity Theory: Notes 4

Ben Chaplin

Contents

1 Complexity

1.1 Complexity classes

Definition. Let $t : \mathbb{N} \to \mathbb{R}^+$. The **complexity class TIME** is defined:

$$\text{TIME}(t(n)) = \{L : L \text{ is a language decided by some TM in } O(t(n)) \text{ time}\}.$$

Example. PALINDROMES $\in$ TIME(n^2). Because there is a Turing Machine which decides PALINDROMES in $O(n^2)$ time.

An interesting fact: if there exists a TM deciding some language L in $5n^3$ time, then there also exists a TM deciding L in n^3 time, and $\frac{1}{1000}n^3$ time... etc. Basically, we can make constants arbitrarily small with Turing Machines by adding states and tape symbols. However the big-O classes remain the same.

Definition. The **complexity class P** is defined P$= \bigcup_{k\in\mathbb{N}}$TIME$(n^k)$. In other words, P is the class of languages decidable in polynomial time.

Note that the previous definition is model-agnostic. Even multitape Turing Machines are just a polynomial factor faster than single-tape Turing Machines. So P is the same no matter which model we choose.

1.2 Halting problem

Definition. ACCEPTS $= \{\langle M, w\rangle : M$ is a TM with input alphabet $\Sigma, w \in \Sigma^*, M(w)$accepts$\}$.

Theorem 1. *There exists a* **Universal TM** $\mathcal{U}$ *that takes as an input* $\langle M, w\rangle$ *and simulates* $M(w)$*.*

Theorem 2 (Turing's Theorem)**.** *There exists no TM* H *that decides ACCEPTS.*

Proof. Assume to the contrary that H exists. We define another TM D that takes an input encoding of a Turing machine. D prepares the input $\langle M, \langle M\rangle\rangle$ and runs H on it. Then, if H accepts, D rejects, and if H rejects, D accepts.

Note that D is a decider, because H is a decider. But now, take $D(\langle D\rangle)$. The computation will run:

$$H(\langle D, \langle D\rangle\rangle)$$

Say D accepts $\langle D\rangle$. That means H rejects $\langle D, \langle D\rangle\rangle$, which means that D rejects $\langle D\rangle$. We have a contradiction. $\square$

This argument is known as a diagonalization argument.