Ch. 4 Computation

1 Definition of a Program

Program P is defined as an ordered set of logical operations s_i

$$P \equiv \{s_1, s_2, ..., s_N\} \quad (Definition3)$$

2 Definition of a Decision Problem

A decision problem is a program whos output is True/False

$$d_i \equiv \{s_1, s_2, ..., return \quad 0/1, ..., s_N - 1, return \quad 0/1\} \quad (Definition4)$$

2.1 Define D the set of Decision Problems

Define D as the set of all Decision Problems d_i

$$D \equiv \{d_1, d_2, ...\}(Theorem1)$$

2.2 Conjecture: D is a finite, never decreasing Set (bounded by language)

2.3 Time Complexity of a Decision Problem

The Time Complexity $O_T[n]$ of Decision Problem d

$$O_T[n] <= n(D)(Theorem1)$$

3 Definition Proper Decision Problem

$$D \equiv \{s_1, s_2, , ..., s_N - 1, return \quad 0/1\}$$

3.1 Time Complexity of a Proper Decision Problem

The Time Complexity $O_T[n]$ of Proper Decision Problem D

$$O_T[n] = n(d)(Theorem2)$$

4 Definition of Complexity

Define Complexity $O[n]$ as a Tensor of dimension N

$$\mathbf{O}[n] \equiv \; < O_T[n], O_S[n], O_3[n], O_4, ..., O_N[n] > \quad (Definition1)$$

4.1 Total Complexity

The Total Complexity of a Decision problem d

$$O[n] \equiv O_T[n] + O_S[n] + \sum_{n=3}^{N} O_i[n](Definition2)$$

4.2 Time Complexity

Define Time Complexity O_T as the maximum number of logical operations in a Program P

$$O_T[n] \equiv n(P) \quad (Definition 2)$$

4.3 Space Complexity

Define Time Complexity O_T as the maximum number of bits required to complete Program P

4.4 Conjecture: Complexity is a 2D Tensor in Space and Time

Conjecture

$$\mathbf{O}[n] \equiv < O_T[n], O_S[n] > \quad (Conjecture 1)$$

5 Definition of Polynomial Time Complexity

A proper decision problem D with Time Complexity $O_T[n]$ can be solved with Polynomial Time Complexity if

$$\exists K, C \ \ \ni \ \ O[n] < n^K + C, \quad \forall n$$

5.1 Definition of Polynomial Problems

Define P, the set of Proper Decision Problems that can be solved with Polynomial Time Complexity

$$P \equiv \{d_1, d_2, .., d_N\} \in D$$
$$\exists K, C \ \ \ni \ \ O[n] < n^K + C, \quad \forall d_i \in P$$

5.2 Proof of the existence of P

Trivial

5.3 Definition of Non-Polynomial Problems

Define $\mathcal{N}$, the set of Proper Decision Problems that cannot be solved with Polynomial Time Complexity

$$\mathcal{N} \equiv P \perp$$

$$\implies \nexists K, C \ \ \ni \ \ O[n] < n^K + C, \quad \forall d_i \in \mathcal{N}$$

5.4 Proof of the existence of $\mathcal{N}$

Non-trivial - equates to the proof of P $\neq$ NP

5.5 Union of Polynomial and Non-Polynomial Problems

Can be proven by the definition of $P, \mathcal{N}, D$

$$\mathcal{N} \cup P = D$$

5.6 Divergence of Polynomial and Non-Polynomial Decision Problems

Prove

$$limit_{n \to \infty} \frac{O_{\mathcal{N}}[n]}{O_P[n]} > 1$$

6 Fundamental Theorem of Computation - "Theorem of Divergent Programs"

6.1 Define $O_\perp$

Define Set $O_\perp$ as the set of all programs $\overline{p}_i$ whose complexity can be expressed as $(O[n])^n$

$$\{\overline{p}_1, \overline{p}_2, .., \overline{p}_n\} \equiv O_\perp$$
$$\forall \overline{p}_i \in O_\perp, O[n] = (O[n])^n$$

6.2 Fundamental Theorem of Computation - "Theorem of Divergent Programs"

$$O_\perp = \emptyset$$

6.3 Proof of Fundamental Theorem of Computation

Prove that $O_\perp = \emptyset$

Assumptions:
1. Let D be the set of all decision problems d_i

$$D \equiv \{d_1, d_2, ..\}$$

2. Let P be the set of Polynomial Decision Problems

$$P \equiv \{d_i, d_j, .., d_N\} \in D$$
$$\exists K, C \quad \ni \quad O_T[n] < n^K + C, \quad \forall d_i \in P$$

3. Let $\mathcal{N}$ be the set of Non-Polynomial Problems

$$\mathcal{N} \equiv P \perp$$

Assertions
4. P and $O_\perp$ are disjoint

$$\exists K, C \ \ni \ O_T[n] < n^K + C, \quad \forall d_i \in P$$

5. $\mathcal{N}$ and $O_\perp$ are disjoint
6. Statement 4. and Statement 5. $\implies O_\perp = \emptyset$

7 Proof of "P $\neq$ NP"

8 Conjecture of Prime Numbers

Define Ψ[n]as the system that maps an index n to the set of Prime Numbers $\mathcal{P}$

$$\Psi[n] \text{ is a divergent system}$$

9 "Universal Problem Solver"