Computational Complexity and Computability

Lecture 10 - Some More Decidable & Undecidable Problems

Koushik Pal

University of Toronto

February 10, 2021

Kolmogorov Complexity

Definition

Let $x \in \{0,1\}^*$.

The minimal description of x, denoted $d(x)$, is the lexicographically shortest string $\langle M, w \rangle$ such that M is a TM that on input w halts with only x on its tape.

The descriptive complexity (also known as Kolmogorov complexity) of x, denoted $K(x)$, is the length of the minimal description of x, i.e.,

$$K(x) = |d(x)|.$$

Incompressible Strings

Definition

A binary string x is incompressible if $K(x) \geq |x|$.

Theorem

For all $n \geq 1$, there is an $x \in \{0,1\}^n$ such that $K(x) \geq n$.
(Incompressible strings of every length exist.)

Proof.

Since each description is itself a binary string, the number of descriptions of length less than n is at most the sum of the number of strings of each length up to $n-1$, i.e.,

$$\sum_{0 \leq i \leq n-1} 2^i = 1 + 2 + 4 + 8 + \cdots + 2^{n-1} = 2^n - 1.$$

But the total number of binary strings of length n is 2^n.
Hence, at least one string of length n is incompressible. □

Incompressible Strings

Definition

Let $INCOMP = \{x \in \{0,1\}^* \mid K(x) \geq |x|\}$ be the set of incompressible strings.

Theorem

$INCOMP$ is undecidable.

Intuition: If $INCOMP$ were decidable, we could design an algorithm that prints the first incompressible string of length n. But then such a string could be succinctly described by giving the algorithm and n in binary.

Berry's Paradox: "The smallest integer that cannot be defined in less than thirteen words."

Incompressible Strings

Proof.

Assume, for a contradiction, that M is a decider for $INCOMP$.

Define a new TM M' as follows:

On input $\langle n \rangle$:

- *Generate strings s of length n lexicographically*
- *Simulate M on s; if M accepts s, halt with s on the tape.*

Let $s_n \in \{0,1\}^n$ be the output of M' on $\langle n \rangle$. Then, M accepts s_n. Hence, $K(s_n) \geq n$.

Conversely, $\langle M', \langle n \rangle \rangle$ is a description of s_n. Hence,

$$K(s_n) \leq |\langle M', \langle n \rangle \rangle| \leq 2|\langle M' \rangle| + \lceil \log n \rceil + 1 \leq \log n + c,$$

where $c = 2|\langle M' \rangle| + 1$ is a constant.

Choosing n large enough so that $\log n + c < n$ yields the required contradiction. □

Exercise: $INCOMP \in coSD$.

Incompressible Strings

Theorem

$INCOMP$ does not contain any infinite subset that is Turing recognizable.

Theorem

The function K is not computable.

So we have no way to obtain long incompressible strings, and no way to determine whether a given string is incompressible.

A super quick introduction to Mathematical Logic

A first order language $\mathcal{L}$ consists of

- Variables, e.g., $x_1, x_2, \ldots$
- Constants, e.g., $c_1, \ldots, c_n$
- Relation Symbols, e.g., $R_1, \ldots, R_k$
- Function Symbols, e.g., $f_1, \ldots, f_\ell$
- Boolean operators, e.g., $\wedge, \vee, \neg, \implies$
- Quantifiers, e.g., $\forall, \exists$

Using these, one can inductively define $\mathcal{L}$-formulas and $\mathcal{L}$-sentences ($\mathcal{L}$-formulas without free variables).

An $\mathcal{L}$-theory T is simply a set of $\mathcal{L}$-sentences.

By $Th(M, c_1, \ldots, c_n, R_1, \ldots, R_k, f_1, \ldots, f_\ell)$ we mean the set of $\mathcal{L}$-sentences that hold true in the universe M. For example,

$$\forall x \exists y (x + x \leq y) \in Th(\mathbb{N}, \leq, +).$$

List of decidable problems

- $Th(\mathbb{Q}, <)$
- $Th(\mathbb{N}, 0, 1, +)$ (Presburger arithmetic)
- $Th(\mathbb{Q}, 0, +)$
- $Th(\mathbb{R}, 0, 1, +, \times)$
- $Th(\mathbb{C}, 0, 1, +, \times)$

List of undecidable problems

- $Th(\mathbb{N}, 0, 1, +, \times)$ (True Arithmetic)
- Hilbert's Tenth Problem: Given a Diophantine equation (multivariable polynomial equation with integer coefficients), is there an algorithm to decide if it has a solution in integers?
- Post Correspondance Problem: Given a collection of dominos $P = \left\{ \left[\frac{a_1}{b_1}\right], \ldots, \left[\frac{a_k}{b_k}\right] \right\}$, is there an algorithm to decide if there is a match, i.e. a sequence $i_1 \ldots i_\ell$ such that $a_{i_1} \cdots a_{i_\ell} = b_{i_1} \cdots b_{i_\ell}$?
- Wang Tiling Problem: Given a set of square tiles with a color on each side, is there an algorithm to decide whether they can tile the whole plane, where tiling requires that the tiles cannot be rotated or reflected and two adjacent tiles must have matching colors?
- Matrix Mortality Problem: Given a finite set of $n \times n$ matrices with integer entries, is there an algorithm to decide if they can be multiplied in some order, possibly with repetition, to yield the zero matrix?

Theorem (Pal, 2011)

Let $\mathcal{K} = (K, \Gamma, k; v, \pi, s, \sigma)$ be a multiplicative valued difference field. The theory of $\mathcal{K}$ is decidable (in a 3-sorted language $\mathcal{L}_{3vdfs}$) if and only if the theories of Γ and k are decidable.