Aspects of Kolmogorov complexity

Brett Berger, Scott Blair, Taha Çakmak, Brenda Chen, Michael Chen*, Johanna Franklin*, Sripad Ganti, Dario Greco, James Hu, Wanting Huang, Hritika Kathuria, Zelong Li, Nethmin Liyanage, Santiago Morales, Martín Prado, Bengüsu Nar, Yusuf Nar, Shantanu Nene, Yunheng (Danny) Ni, Sanat Shajan, John Solak, Wenxuan Su, Raymond Tao, James Thompson, Berke Yavuz, Alan Zhou

January 6, 2023

Kolmogorov Complexity

Imagine this scenario, if you want to see the latest movie Bullet Train starring by Brad Pitt this Saturday, then you texted three of your friends, and they replied:

A: "Hi Taha, I am busy Saturday and Monday."

B: "Sorry, I gotta work on a school project all week."

C: "I'm all tied up this weekend!"

Kolmogorov Complexity

Kolmogorov complexity is a measure of the amount of information needed to describe the object.

Take the following two strings of length 16 as an example:

0101010101010101

1100100001100001

The FIRST string can be briefly described as 01 repeated 8 times, and the SECOND string has no obvious brief description.

Notation

- α, β: Infinite binary strings
- τ, σ: Finite binary strings
- 2^ω: Set of all the infinite binary strings (bit-strings)
- $2^{<\omega}$: Set of all the finite binary strings (bit-strings)
- $\tau \preceq \sigma$: τ is an initial segment of σ

Fundamental concepts

- Partial computable functions
- Computably enumerable sets
- Machines

Partial computable functions

A function is partial computable if it can be computed by a program.

Computably enumerable sets

A set is computably enumerable if it can be enumerated (listed) by a program. Equivalently, a set is computably enumerable if it is the domain of some partial computable function.

Hereinafter referred to as "C.E sets"

Machines

Regular Turing machines

Let $M : 2^{<\omega} \to 2^{<\omega}$ be a partial computable function. The Kolmogorov complexity of a string σ with respect to M is

$$C_M(\sigma) = \min\{|\tau| : M(\tau) = \sigma\}.$$

Machines

Universal machines

We need a universal description system to get rid of the dependence on M from the last slide. Such a system should only increase the descriptions' lengths by a constant amount. We call such a system a universal machine.

We fix a partial computable function $U : 2^{<\omega} \to 2^{<\omega}$ that is universal in the sense that, for each partial computable function $M : 2^{<\omega} \to 2^{<\omega}$, there is a string ρ_M such that

$$\forall \sigma \left[U\left(\rho_M \sigma \right) = M(\sigma) \right].$$

ρ_M is the coding string and $|\rho_M|$ is the coding constant of M in U .

A process machine M is a partial computable function that behaves in the following manner: Given $\tau, \tau' \in 2^w$ where $\tau' \preceq \tau$, then $M(\tau') \preceq M(\tau)$ (if $M(\tau') \downarrow$ and $M(\tau) \downarrow$).

The process complexity of σ is

$$K_{M_D}(\sigma) = \min\{|\tau| : \mathcal{U}(\tau) \downarrow = \sigma\}$$

for a universal process machine U.

Note: The $\downarrow$ means the machine returns an output for that input.

Partial Order

Basic Definitions

- A partial order on a set A is a binary relation $\preceq$ which shares many properties with the normal $\leq$ relation, except that there might be incomparable elements.
- Example: Finite binary strings with $\alpha \preceq \beta$ if β is an extension of α (so $0 \preceq 011$, but 0110 and 001 are incomparable).
- Example: Divisibility of natural numbers. In this partial order, 6 and 15 are incomparable.

Inspired by Calhoun (2006), we define an ordering on infinite binary strings (also called reals) using process complexity:

Definition

Given reals α and β, we say that $\alpha \leq^* \beta$ if there exists $c \in \mathbb{N}$ such that

$$K_{M_D}(\alpha \restriction n) \leq K_{M_D}(\beta \restriction n) + c$$

for all $n \in \mathbb{N}$.

If $\alpha \leq^* \beta$ and $\beta \leq^* \alpha$, we say $\alpha =^* \beta$.

Definition
The K_{M_D}-degree of α is the set $\{\beta \in 2^\omega \mid \beta =^* \alpha\}$, i.e., the set of reals that have the "same" complexity as α.

The relation $\leq^*$ defined above gives us a partial order on K_{M_D}-degrees.

Partial Order

Bounds of Process Complexity

Theorem

There is a minimum K_{M_D}-degree, and it consists precisely of the computable reals.

Theorem

There is a maximum K_{M_D}-degree, and it consists precisely of the 1-random reals.

Partial Order

Subadditivity

- A nice property we might wish to have is *subadditivity*: the existence of a constant c such that for all finite strings σ and τ, $K_{M_D}(\sigma\tau) \leq K_{M_D}(\sigma) + K_{M_D}(\tau) + c$.
- Subadditivity holds for prefix-free Kolgomorov complexity.
- Unfortunately, this was previously known to be false for process complexity.

  However, we showed that a weaker version does hold:

Theorem

There exists a constant d such that if σ and τ are finite strings, we have

$$K_{M_D}(\sigma\tau) \leq K(\sigma) + K_{M_D}(\tau) + d.$$

Partial Order

In-progress results

Our recent result on subadditivity makes it likely that we will be able to prove the following:

Conjecture

For any real α which is not computable or 1-random, there exists an incomparable real β (a real β such that $\alpha \not\leq^* \beta$ and $\beta \not\leq^* \alpha$).

If this holds, then we will probably be able to show that there is an uncountable antichain of K_{M_D}-degrees (an uncountable set of degrees, all of which are incomparable).

Conjecture

There exists a minimal pair of K_{M_D}-degrees (a pair of reals α and β, neither of which are computable, such that if a real γ satisfies $\gamma \leq^* \alpha$ and $\gamma \leq^* \beta$, γ is computable.)

2-c.e. sets

Definition

An n-c.e. set S_n is defined for $n \in \mathbb{N}$ as n steps of alternating set differences and set unions of c.e. sets.
That is, $S_n = (U_1 \setminus U_2) \cup (U_3 \setminus U_4) \cup ...$, where $\forall i \in \{1, ..., n\}$, U_i is c.e.

Definition

2-c.e. sets are the difference of two c.e. sets. Namely, $S_2 = U \setminus V$ where U, V are c.e.

Example

Let A be the c.e. set of even naturals $\{2, 4, 6, 8, ...\}$ and B be the c.e. set $\{4, 5, 10\}$. Then $A \setminus B = \{2, 6, 8, 12, 14, 16, ...\}$ is 2-c.e.

2-c.e. sets

Theorem (Existence of non-trivial 2-c.e. sets; Ershov)

For any countably infinite c.e. set U, there exists V c.e. such that $U \setminus V$ is 2-c.e. but not c.e.

Theorem

A function is partial computable if and only if its graph is c.e.

Broad 2-c.e. machine

Definition

M is a Broad 2-c.e. machine if $M = U \setminus V$ for some c.e. sets of ordered pairs U, V.

- Note: This is a relation, not necessarily a function.

D1 2-c.e. and D2 2-c.e. machines

Definition

M is a D1 2-c.e. machine if $M = U \setminus V$ for some c.e. sets of ordered pairs UV, and

$$((\sigma, \tau_1) \in U \wedge (\sigma, \tau_2) \in U) \Rightarrow (\tau_1 = \tau_2).$$

Definition

M is a D2 2-c.e. machine if $M = U \setminus V$ for some c.e. sets of ordered pairs U, V and

$$((\sigma, \tau_1) \in (U \setminus V) \wedge (\sigma, \tau_2) \in (U \setminus V)) \Rightarrow (\tau_1 = \tau_2).$$

Non-equivalence of machines

Theorem

None of the three types of machines are equivalent to each other. More specifically, D1 machines $\subsetneq$ D2 machines $\subsetneq$ Broad machines.

Universal machines

Universal Machine

Theorem

There are universal Broad 2-c.e. machines, universal D1 2-c.e. machines, and universal D2 2-c.e. machines.

Summary

	D1	D2	Broad
Function	Yes	Yes	No
Universal machine	Yes	Yes	Yes
1-c.e def. familiar	Yes (standard)	Yes (standard)	No

Some results on complexity

Definition

The complexity of a string σ relative to a D1 machine M is defined as

$$C_{\mathrm{D1M}}(\sigma) = \min\{|\tau| \mid (\tau, \sigma) \in M\}.$$

Similarly, the D1 complexity of a string is defined as

$$C_{\mathrm{D1}}(\sigma) = \min\{|\tau| \mid (\tau, \sigma) \in U\}$$

where U is the universal machine.

Complexity is defined similarly for D2 machines.

Some results on complexity

Theorem
1) $C_{D1}(\sigma) \leq C(\sigma) + O(1)$

Theorem
2) $C(\sigma) \leq C_{D1}(\sigma) + O(1)$

Ongoing Research

- Closure properties: Do 2-c.e. machines map 2-c.e. sets to 2-c.e. sets?
- More complexity properties of D1 and D2 machines
- Generalization to n-c.e. machines for $n > 2$

References

General:

- Rodney G. Downey and Denis R. Hirschfeldt, *Algorithmic Randomness and Complexity*, Springer, 2010.

Partial orders:

- William C. Calhoun, Degrees of monotone complexity. *J. Symbolic Logic*, 71(4):1327–1341, 2006.

n-c.e. machines:

- Ershov, Y. L. A hierarchy of sets, Part I. *Algebra i Logika*, 7:24–43, 1968. English translation.
- Ershov, Y. L.A hierarchy of sets, Part II. *Algebra i Logika*, 7:212–232, 1968. English translation.
- Ershov, Y. L.A hierarchy of sets, Part III. *Algebra i Logika*, 9:20–31, 1970. English translation.