On the Halting Probability and Chaitin's Heuristic Principle

Saeed Salehi

January 2024

GREGORY JOHN CHAITIN

Born: 1947[77] (Jewish)

Argentine-American

Algorithmic Information Theory

A. KOLMOGOROV & R. SOLOMONOFF

1. Incompleteness (1971)[24]
2. Heuristic Principle (1974)[27]
3. Halting Probability (1975)[28]
   CHAITIN's Constant: **Ω**

$\leftarrow$ March 2001[54]

IBM's Thomas John Watson Research Center in New York

A Genius

Many Honors (& writings)

Lots of Criticism (& fans)

HP: **H**euristic **P**rinciple / **H**alting **P**robability

- On Chaitin's Heuristic Principle and Halting Probability.
  arXiv:2310.14807v3 [math.LO].
  `https://arxiv.org/abs/2310.14807`

1. **H**euristic **P**rinciple
2. **H**alting **P**robability

1. Chaitin's Heuristic Principle

- Greater Complexity Implies Unprovability

  If a sentence is more complex (heavier) than the theory, then that sentence is *unprovable* from that theory.

(Un-)Provability:

Example (Arithmetic & Geometry)

$\mathfrak{Arithmetic} \vdash \neg\exists x,y,z\,(xyz \neq 0 \;\wedge\; x^4+y^4=z^2).$ Pierre de Fermat

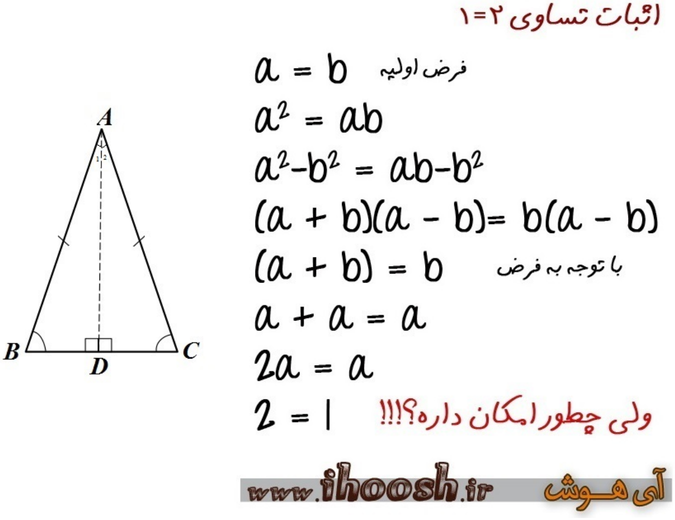
$\mathfrak{Arithmetic} \vdash \exists x,y,z>1\,(x^4+y^4=z^2+1).$ $x=5,\ y=7,\ z=55$

$\mathfrak{Arithmetic} \vdash \exists x,y,z\,(xyz \neq 0 \;\wedge\; x^4+y^4+1=z^2)$?

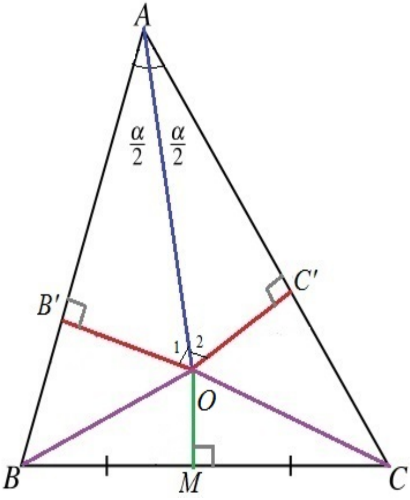
$\mathfrak{Geometry} \vdash \forall\triangle ABC\,(\overline{AB}=\overline{AC} \longleftrightarrow \angle B=\angle C)$

$\mathfrak{Arithmetic} \nvdash 1=2$ $\mathfrak{Geometry} \nvdash \forall\triangle ABC\,(\overline{AB}=\overline{AC})$

$\mathfrak{Arithmetic} \nvdash 1 = 2$



$\mathfrak{Geometry} \nvdash \forall \triangle ABC\,(\overline{AB}=\overline{AC})$



- $\angle BAO = \angle CAO \implies$
  $\triangle OB'A \cong \triangle OC'A \implies$
  $\overline{AB'} = \overline{AC'}$ & $\overline{OB'} = \overline{OC'}$

- $\overline{BM} = \overline{MC} \implies$
  $\triangle OMB \cong \triangle OMC \implies$

$\overline{OB} = \overline{OC} \implies$
$\triangle OBB' \cong \triangle OCC' \implies$
$\overline{B'B} = \overline{C'C} \implies$

$\overline{AB'} + \overline{B'B} = \overline{AC'} + \overline{C'C}$
$\implies \overline{AB} = \overline{AC}$

Incompleteness (vs. Completeness)

Tarski$_{1930's}$

$$\xrightarrow{\texttt{input}:\ \psi} \boxed{\text{Algorithm}} \xrightarrow{\texttt{output}:} \begin{cases} 1 & \text{if } \mathfrak{Geometry} \vdash \psi \\ 0 & \text{if } \mathfrak{Geometry} \nvdash \psi \end{cases}$$

Gödel$_{1931}$

$$\xrightarrow{\texttt{input}:\ \psi} \boxed{\cancel{\text{Algorithm}}} \xrightarrow{\texttt{output}:} \begin{cases} 1 & \text{if } \mathfrak{Arithmetic} \vdash \psi \\ 0 & \text{if } \mathfrak{Arithmetic} \nvdash \psi \end{cases}$$

Solomonoff-Kolmogorov-Chaitin Complexity

Definition (Program Size Complexity)

$\mathcal{C}(x) =$ the length of
the shortest input-free program that outputs only x (and halts).

Example

$(10)^n = 1010\cdots10$	$\{10^n\}_{n=1}^{\infty} = 10100100010000\cdots10^n10^{n+1}\cdots$

`BEGIN`
`  input` n
`  for` $i=1$ `to` n
`      print` **1**
`      print` **0**
`END`
	`BEGIN`
	`  let` $n=1$
	`  while` $n>0$ `do`
	`    begin`
	`      print` **1**
	`      for` $i=1$ `to` n
	`        print` **0**
	`      let` $n=n+1$
	`    end`
	`END`

Descriptive Complexity & Randomness

- 1111111111111111111111111111111111111 $\cdots$ 1^*
- 100100100100100100100100100100100 $\cdots$ $(100)^*$
- 0101101110111101111101111110111 $\cdots$ $\{01^n\}_{n>0}$
- 0101111010111110111111111011 $\cdots$ $\{01^{(\pi-3)_n}\}_{n=1}^{\infty}$
- 110001100001111110000010010100001101010 $\cdots$

Definition (Random)

A random number or a string is one whose program-size complexity is almost its length.

COMPLEXITY OF SENTENCES AND THEORIES

Arithmetic:

- $\exists x,y,z\,(xyz\neq 0 \ \wedge\ x^2+y^2=z^2)$ $x{=}3,y{=}4,z{=}5$
- $\neg\exists x,y,z\,(xyz\neq 0 \ \wedge\ x^3+y^3=z^3)$
- $\neg\exists x,y,z\,(xyz\neq 0 \ \wedge\ x^4+y^4=z^4)$
- $\forall n>2\neg\exists x,y,z\,(xyz\neq 0 \ \wedge\ x^n+y^n=z^n)$

Geometry:

- $\forall\triangle ABC\,(M_a,M_b,M_c\text{midpoints}\longrightarrow\exists\mathbb{G}[AM_a\cap BM_b\cap CM_c=\{\mathbb{G}\}])$
- $\forall\triangle ABC\,(AA',BB',CC'\text{altitudes}\longrightarrow\exists\mathbb{H}[AA'\cap BB'\cap CC'=\{\mathbb{H}\}])$
- $\forall\triangle ABC\,\exists!\mathbb{O}\,(\overline{\mathbb{O}A}=\overline{\mathbb{O}B}=\overline{\mathbb{O}C})$
- $\forall\triangle ABC\,(\mathbb{G},\mathbb{H},\mathbb{O}\text{ are identical or on a line})$

Heuristic Principle, HP

Definition (HP-satisfying weighing)

A mapping $\mathscr{W}$ from theories and sentences to $\mathbb{R}$ satisfies HP when, for every theory $\mathcal{T}$ and every sentence ψ we have

$$\mathscr{W}(\psi) > \mathscr{W}(\mathcal{T}) \Longrightarrow \mathcal{T} \nvdash \psi.$$

Equivalently, $\mathcal{T} \vdash \psi \Longrightarrow \mathscr{W}(\mathcal{T}) \geqslant \mathscr{W}(\psi)$

- Chaitin's Idea: program-size complexity
- Lots of Criticisms ...
- Some built their own *partial* weighting
- Fans come to rescue ...

HP, a lost paradise

- **Criticisms:**

  For complex sentences $\mathfrak{S}, \mathfrak{S}'$, or complex numbers $\mathcal{N}, \mathcal{N}'$, the following *complicated* sentences are all provable:

  - $\mathfrak{S} \to \mathfrak{S}$, $\mathfrak{S} \wedge \mathfrak{S}' \to \mathfrak{S}' \wedge \mathfrak{S}$, $(\neg\mathfrak{S}' \to \neg\mathfrak{S}) \Rightarrow (\mathfrak{S} \to \mathfrak{S}')$.
  - $1 + \mathcal{N} = \mathcal{N} + 1$, $\mathcal{N} \times \mathcal{N}' = \mathcal{N}' \times \mathcal{N}$, $n(\mathcal{N} + \mathcal{N}') = n\mathcal{N} + n\mathcal{N}'$.

- **A Salvage?**

  △ δ-complexity: $\mathcal{C}(x) - |x|$.

  XXX $\mathcal{T} \vdash \psi \Longrightarrow \delta(\mathcal{T}) \geqslant \delta(\psi)$ XXX

- **No Hope:**

  - $\bot \to \mathfrak{S}$, $\mathfrak{S} \to \top$, $p \to (\mathfrak{S} \to p)$, $\neg p \to (p \to \mathfrak{S})$.
  - $\mathcal{N} > 0$, $\mathcal{N} \times 0 = 0$, $1 + \mathcal{N} \neq 1$, $2 \leqslant 2 \times \mathcal{N}$.

More on the WLD maybe

2. Chaitin's Halting Probability

- Halting Probability (of a randomly given input-free program)

$$\mathbf{\Omega} = \sum_{p \text{ halts}} 2^{-|p|}.$$

Halting or Looping forever:

A random $\{0, 1\}$-string may not be (the ASCII code of) a program.
Even if it is, then it may not be input-free.
If a binary string is (the code of) an input-free program, then
it may halt after running or may loop forever.

$$\mathbf{\Omega} = \sum_{p \in \{0,1\}^* \text{ halts}}^{p\text{: input-free}} 2^{-|p|}.$$

A Partial Agreement

The probability of getting a fixed binary string of length n by tossing a fair coin (whose one side is '0' and the other '1') is 2^{-n}, and the halting probability of programs with size n is

$$\frac{\text{the number of } \textit{halting programs} \text{ with size } n}{\text{the number of } \textit{all binary strings} \text{ with size } n} = \frac{\#\{p \in \mathbb{P} : p\!\downarrow \;\&\; |p| = n\}}{2^n}$$

since there are 2^n binary strings of size n. Thus, the halting probability of programs with size n can be written as $\sum_{p\downarrow}^{|p|=n} 2^{-|p|}$.

Denote this number by Ω_n; so, the number of halting programs with size n is $2^n\Omega_n$.

And a Disagreement

According to Chaitin (and almost everybody else), the halting probability of programs with size $\leqslant N$ is $\sum_{n=1}^{N}\Omega_n = \sum_{p\downarrow}^{|p|\leqslant N} 2^{-|p|}$; and so, the halting probability is $\sum_{n=1}^{\infty}\Omega_n = \sum_{p\downarrow} 2^{-|p|} (= \mathbf{\Omega})$!

Let us see why we believe this to be an error. The halting probability of programs with size $\leqslant N$ is in fact

$$\frac{\text{the number of halting programs with size} \leqslant N}{\text{the number of all binary strings with size} \leqslant N} = \frac{\sum_{n=1}^{N} 2^n\Omega_n}{\sum_{n=1}^{N} 2^n}.$$

Now, it is a calculus exercise to notice that, for sufficiently large Ns,

$$\frac{\sum_{n=1}^{N} 2^n\Omega_n}{\sum_{n=1}^{N} 2^n} \neq \sum_{n=1}^{N}\Omega_n, \text{ and } \lim_{N\to\infty}\frac{\sum_{n=1}^{N} 2^n\Omega_n}{\sum_{n=1}^{N} 2^n} \neq \lim_{N\to\infty}\sum_{n=1}^{N}\Omega_n.$$

Possible Errors / Mistakes

The number $\mathbf{\Omega}$ was meant to be "the probability that a computer program whose bits are generated one by one by independent tosses of a fair coin will eventually halt".

As pointed out by Chaitin, the series $\sum_{p\downarrow} 2^{-|p|}$ could be > 1, or may even diverge, if the set of programs is not taken to be *prefix-free* (that "no extension of a valid program is a valid program"—what "took ten years until [he] got it right").

So, the fact that, for *delimiting* programs, the real number $\sum_{p\downarrow} 2^{-|p|}$ lies between 0 and 1 (by Kraft's inequality, that $\sum_{s\in S} 2^{-|s|} \leqslant 1$ for every prefix-free set S) does not make it the probability of anything!

ANY SOLUTIONS?

1. CONDITIONAL PROBABILITY

   Let $\Omega_S = \sum_{s\in S} 2^{-|s|}$ and $\mho_S = \Omega_S/\Omega_{\mathbb{P}}$ for a set $S \subseteq \mathbb{P}$ of programs.
   This is a probability measure: $\mho_\emptyset = 0$, $\mho_{\mathbb{P}} = 1$, and for any family $\{S_i \subseteq \mathbb{P}\}_i$ of pairwise disjoint sets of programs, $\mho_{\bigcup_i S_i} = \sum_i \mho_{S_i}$.
   If $\mathcal{H}$ is the set of all the binary codes of the halting programs, then the (conditional) halting probability is $\mho_{\mathcal{H}}$, or $\boldsymbol{\Omega}/\Omega_{\mathbb{P}}$.
   We then have $\mho_{\mathcal{H}} > \boldsymbol{\Omega}$ since it can be shown that $\Omega_{\mathbb{P}} < 1$.

2. ASYMPTOTIC PROBABILITY

   Count $\hbar_n$ the number of halting programs (the strings that code some input-free programs that eventually halt after running) that have integer codes‡ less than or equal to n. Then define the halting probability to be $\lim_{n\to\infty} \hbar_n/n$, of course, if it exists.
   Or take $\lim_{N\to\infty} \left(\sum_{n=1}^{N} 2^n\Omega_n\right)/\left(\sum_{n=1}^{N} 2^n\right)$ if the limit exists.
   Note that this number can be shown to be $\leqslant \dfrac{\boldsymbol{\Omega}}{2}$.

‡ integer code: $0_1, 1_2, 00_3, 01_4, 10_5, 11_6, 000_7, 001_8, 010_9, \cdots$

Thank You!

Thanks to

The Participants .... For Listening $\cdots$

and

The Organizers, For Taking Care of Everything $\cdots$