从 AIT 看不完全

摘要：Gödel1931 证明了一个包括初等数论的形式系统如果一致则不完全，即存在为真但形式系统内不可证的算数命题，Turing1936 对停机问题的分析以及 Martin Davis,Hilary Putnam,Julia Robinson 和 Yuri Matiyasevich1970 对 Hilbert 第十问题的否定解决加强了这一结果。但在现实中，不完全性所引起的兴趣仿佛仅限于逻辑学家中间,数学实践中主流数学家们对此故意视而不见无动于衷或敬而远之。不完全性的根源何在？“真”与“可证”之间的鸿沟有多宽？算法信息论给出了新的视角：因为那些真命题的复杂性大于形式系统本身的复杂性，所以不可证，即定理本身不能比系统更复杂。随着 n 趋向于无穷，形式系统内为真并可证的命题的概率趋向于 0。本文将综述与此相关的一些有趣结果并分析其哲学意蕴。

关键字：算法复杂性、算法随机性、丢番图方程、Ω、不完全

手扶拐杖的外星绅士造访地球，想把地球文明传播到自己星球。临别时，地球人慷慨赠送百科全书：“所有人类知识尽在其中！”。绅士谢绝：“不，谢谢。我只需在手杖上点上一点”。

1 小引

Kolmogorov1933 年对概率论的公理化不能处理样本空间中某个具体的个体。将一枚质地均匀的硬币连续抛掷 n 次，正面记为 1，反面记为 0，古典概率论只能告诉我们：出现 n 个 1 的概率与出现任何其他情况的概率相等，都是 2^{-n}。根据直觉，掷硬币是随机的，虽然某种具体的 01 分布出现的概率是 2^{-n},但散乱的情况更容易被接受,n 个 1 就太不够“随机”。这里的“随机”是什么意思呢？Kolmogorov 概率论并没有给出数学上的刻画。

一串序列是随机的，它至少应该具备如下几条性质：

1. 已知序列的前 n 项，并不能预测第 n+1 项。
2. 此序列不能被进一步压缩。
3. 它能通过统计学上的检测。

Kolmogorov1960s 根据性质 2 给出了一个序列的算法复杂性（或称信息量）的定义，即描述此序列的最小程序的长度。与此同时，Martin-Löf、Chaitin、Solomonoff 等人分别独立提出并发展了相关理论，算法信息论(AIT)从此起源。

2 算法复杂性

序列的集合记为 $\Sigma^\star$，ϵ 表示空串，$|s|$ 表示 s 的长度。这里 $\Sigma^\star$ 可以看作 $\{0,1\}^\star=\{\epsilon,0,1,00,01,10,11,000,001,010\cdots\cdots\}$ 即 Cantor 空间 2^ω。集合 $S\subset\Sigma^\star$ 是 prefix-free 的如果对于任意的 s、$t\in S$，不存在 s 是 t 的 prefix 的情况。

约定后面提到的所有机器默认的作用程序集皆为 prefix-free。这种机器俗称 Chaitin 机。即以后提到的机器都默认为 Chaitin 机。约定将自然数与其二进制表示等同视之。这里的二进制表示不是自然数的二进制展开式，而是采用如下形式的字典序表示方式：

$(\epsilon,0),(0,1),(1,2),(00,3),(01,4),(10,5),(11,6),(000,7),(001,8),(010,9)\cdots\cdots$。整个实数轴与单位区间一一对应，所以在此只考虑单位区间上的实数。这里我们约定将一个实数与其二进制展开式等同视之，即 $\alpha=0.a_1a_2a_3\cdots a_n\cdots$其中 $\alpha_i\in\{0,1\}$，为了方便，有时用 α_n 表示序列的前 n 项 $a_1a_2a_3\cdots a_n$或 $0.a_1a_2a_3\cdots a_n$。定义一个实数 α 是递归可枚举的(r.e.)如果它是一串递归不降有理数数列的极限。实数 α 是 computable 如果存在递归的有理数数列$\{a_n\}_{n\in\mathbb{N}}$使得对于任意的 $n\in\mathbb{N}$，$|\alpha-a_n|<2^{-n}$。一个实数序列$\{a_n\}_{n\in\mathbb{N}}$是 computable 如果存在全递归函数 f 使得对任意的 $m,n\in\mathbb{N}$，$|a_n-f(n,m)|<2^{-m}$。

机器 U 为通用机如果对于任意机器 C 存在常数 c 对所有的序列 s、t，若 C(s)=t 则有 s',$|s'|\leqslant|s|+c$ 且 $U(s')=t$。$domU:=\{p\in\Sigma^*:U(p)\downarrow\}$。

定义 program-size 复杂性（或信息量、或 Kolmogorov 熵）：

$$H_C(x):=\min\{|p|:p\in\Sigma^*\&C(p)=x\}$$

C 是某台具体的机器，可以构造通用机器 U 使得：$U(1^{\#C}0p)=C(p)$

这里#C 指 C 的哥德尔数[1]。即，通过对 C 的每个程序 p 前面加上#C 个 1 跟着一个 0，U 模拟 C。所以，易得：$H_U\leqslant H_C+\#C+1$。我们选取某个特定的通用机 U，并简记 $H_U(x)$为 H(x)。

定义算法概率[2]：

$$P_C(x):=\sum_{C(p)=x}2^{-|p|}$$

停机概率即为 $\Omega_U:=\sum_{p\in domU}2^{-|p|}$

定理 2.1：$H(x)=-\log_2P(x)+O(1)$

定义 Joint complexity：

$$H(x,y):=\min\{|p|:U(p)=\langle x,y\rangle\}$$

定理 2.2

(i) $H(x,y)\leqslant H(x)+H(y)+O(1)$

(ii) $H(x,H(x))\leqslant H(x)+O(1)$

称 x,y 是算法独立的当且仅当 $H(x,y)\approx H(x)+H(y)$。

[1]一种常用的前缀码与这里的哥德尔编码类似，即在一个对象前加上它的长度。可以重复这一操作，得到更短的编码：
if $i=0$ $E_i(x)=1^x0$ else $E_i(x)=E_{i-1}(|x|)x$。这样，$E_1(x)=1^{|x|}0x$ 且有长度$|E_1(x)|=2|x|+1$。$E_1(x)$有时简记为$\bar{x}$。常用的是$E_2(x)$且$|E_2(x)|=|x|+2||x||+1$。

[2] 这不是严格的概率测度。而定义的概率会收敛是由 Kraft 不等式保证：对于任意的自然数序列$l_1,l_2,l_3\ldots\ldots$，存在一种前缀码以此序列作为二进制码字的长度，当且仅当 $\Sigma2^{-l_n}\leq1$。

定义 Mutual information content：

$I(x:y)=H(x)+H(y)-H(x,y)$

x 的 elegant program $x^\star$是计算 x 的最小程序。

显然，$U(x^\star)=x$，$H(x)=|x^\star|$，$H(U(x^\star))=|x^\star|$

定义 Relative information content：

$H(x|y)=\min\{|p|:U(p,y^\star)=x\}$

定理 2.3：$H(x,y)=H(x)+H(y|x)+O(1)$

条件概率 P(y|x)形式上与之非常类似：$P(x,y)=P(x)\times P(y|x)$，事实上，定理 2.1 的确建立了二者之间的联系。

将定理 2.3 用于 H(x:y)的定义，易得如下推论：

$I(x:y)=H(x)-H(x|y)+O(1)=H(y)-H(y|x)+O(1)$

下面考虑本文开始掷硬币的例子，因为掷出的 01 序列是随机的，所以它应该接近最高的复杂性：

$\max\{H(x):|x|=n\}=n+H(n)+O(1)\approx n+\log_2 n$

而 N 个 1 的复杂性为：

$H(1^n)=H(n)+O(1)\approx\log_2 n$

对于有穷序列，随机概念是一个“度”的问题，很难说某个长度为 N 的序列是完全随机的，如果一定要硬性区分随机和非随机的话，可以参考 elegant program 的复杂性：

$H(p^\star)=|p^\star|+O(1)$

即，如果 $H(x)\geqslant|x|$则 x 随机。界限划在这里也许是最好的选择。

对于无穷序列，随机性的定义就非常自然。事实上，类似“可计算”[3]的情况，随机性也有多种定义方式不同但相互等价的定义！

3 算法随机性

定义 3.1(weak Chaitin randomness)

对任意 $\alpha\in\mathbb{R}$，称 α 是 weakly Chaitin random，如果$\exists c\in\mathbb{N}\forall n\in\mathbb{N}^+(H(\alpha_n)\geqslant n-c)$。

定义 3.2(Martin-Löf randomness) 『可简记为 ML test』

[3] 如果哲学就是概念分析，那么图灵论题就是哲学的典范。

$C\subset\mathbb{N}^+\times\{0,1\}^\star$是一个 ML test，如果 C 是一个 r.e.集，且$\forall n\in\mathbb{N}^+(\Sigma_{s\in C_n}2^{-|s|}\leqslant 2^{-n}$，其中，$C_n:=\{s:(n,s)\in C\}$。

对任意$\alpha\in\mathbb{R}$，称α是 ML random 如果对任意的 ML test C，$\alpha\notin\cap_{n\in\mathbb{N}}C_n$。

定理 3.3 对任意$\alpha\in\mathbb{R}$，α是 weakly Chaitin random iff α是 ML random。

定义 3.4(universal probability) 函数$\mu:\{0,1\}^\star\to[0,1]$是 lower-computable semi-measure 如果$\Sigma_{s\in\{0,1\}^\star}\mu(s)\leqslant 1$且$\{(\alpha,s)\in\mathbb{Q}\times\{0,1\}^\star:\alpha<\mu(s)\}$为 r.e.

一个 lower-computable semi-measure μ 是一个 universal probability 如果对任意 lower-computable semi-measure ν，$\exists c\in\mathbb{N}^+\forall s\in\{0,1\}^\star(\nu(s)\leqslant c\mu(s))$。

定理 3.5 对任意的通用机器 U，$2^{-Hu(s)}$和$P_U(s)$都是 universal probability。

前面H(s)是据program-size定义的,这里根据定义3.4和定理3.5可给出另一种抽象定义：

$H(s):=-\log_2\mu(s)$。

定义 3.6(Ω-likeness) 对任意的 r.e. α,β，α dominates β 如果有递归的递增有理数序列$\{a_n\}$、$\{b_n\}$使得$\lim_{n\to\infty}a_n=\alpha$，$\lim_{n\to\infty}b_n=\beta$且$\exists c\in\mathbb{N}^+\forall n\in\mathbb{N}(c(\alpha-a_n)\geqslant\beta-b_n)$。

一个 r.e.实数α是Ω-like 如果它 dominates 所有的 r.e.实数。

定理 3.7 对任意的 r.e.实数α,β,如果α domintes β 那么$H(\beta_n)\leqslant H(\alpha_n)+O(1)$。

定义 3.8(universality)一个递归的递增收敛有理数数列$\{a_n\}$是 universal 如果对任意递归的递增收敛有理数数列$\{b_n\}$存在$c\in\mathbb{N}^+$使得对所有的$n\in\mathbb{N}$，有

$c(\alpha-a_n)\geqslant\beta-b_n$，且$\lim_{n\to\infty}a_n=\alpha$，$\lim_{n\to\infty}b_n=\beta$。

定理 3.9 $0<\alpha<1$是一个 r.e.实数，下面条件等价：

(i) α是 weakly Chaitin random。

(ii) α是 ML random。等价的：$\forall n[\mu(A_n)\leqslant 2^{-n}]\to\neg\forall n(\alpha\in A_n)$。

(iii) α是Ω-like。

(iv) 对任意 r.e.实数β，$H(\beta_n)\leqslant H(\alpha_n)+O(1)$。

(v) 存在通用机 U，$\alpha=\Omega_U$。

(vi) 有一个 universal probability μ 使得 α =Σ $_{s\in\{0,1\}^*}$μ (s)。

(vii) 任意收敛到 α 的递归的递增有理数数列是 universal。

(viii) 有一个 universal 递归递增有理数数列收敛到 α 。

(ix) $\sum\mu(A_i)<\infty\rightarrow\exists N(\forall i>N)[\neg(\alpha\in A_i)]$

(x) $\forall k\exists N_k(\forall n\geqslant N_k)[H(\alpha_n)\geqslant n+k]$。

4 智慧的 Ω 与不完全

定理 4.1 给定 Ω_n，可以判定 U 在所有那些长度不超过 n 的程序上是否会停机。

证明：显然，$\Omega_n\leqslant\Omega<\Omega_n+2^{-n}$。

楔形计算：U 第一次执行第一个输入的第一步操作，第二次执行第一个输入的第二步操作和第二个输入的第一步操作……第 i 次执行第 k 个输入的第 j 步操作,i=j+k。一旦某个 p 停止，循环执行 $\Omega'=\Omega'+2^{-|p|}$可逼近 Ω 直到 $\Omega_n\leqslant\Omega'$ 。如果此时 p 没停机则永远不会停，否则若 p 停机了，则 $2^{-|p|}\geqslant 2^{-n}$,那么将有 $\Omega_n+2^{-n}\leqslant\Omega'+2^{-|p|}\leqslant\Omega$,而这与 $\Omega_n\leqslant\Omega<\Omega_n+2^{-n}$ 矛盾。

定理 4.2 任意的 universal U，停机概率 Ω 是 weakly Chaitin random。

证明：$domU=\{p\in\Sigma^*:U(p)\downarrow\}=\{p_1,p_2,p_3\cdots\cdots\}$是递归可枚举的。

定义 $\omega_n=\Sigma_{i\leqslant n}\ 2^{-|pi|}$

显然 $\omega_n<\omega_{n+1}\rightarrow\Omega$

顺序计算 ω_k，k=1,2,3……直到 $\omega_k\geqslant\Omega_n$

即有 $\Omega_n\leqslant\omega_k<\Omega\leqslant\Omega_n+2^{-n}$

由于 $\{x:H(x)\leqslant n\}\subset\{U(p_i):i\leqslant k\}$ 而 $\{U(p_i):i\leqslant k\}$ 是递归的，所以可以取任意一个不在 $\{U(p_i):i\leqslant k\}$ 里面的 x 使 H(x)>n。即存在部分递归函数 Ψ 使得 $\Psi(\Omega_n)=x$，其中 H(x)>n。

但 $n<H(\Psi(\Omega_n))\leqslant H(\Omega_n)+c_\Psi$

所以 $H(\Omega_n)>n-c_\Psi$。

定理 4.3 有一个指数丢番图方程 $A(n,x_1,x_2,\cdots\cdots x_m)=0$，如果 Ω 的第 n 个元素是 0，则方程只有有穷多的解；如果 Ω 的第 n 个元素是 1，则方程有无穷多解。

证明：由 4.2 中的证明过程，$\Omega=\lim_{n\to\infty}\omega_n$

集合 $R=\{(n,k):\omega_k$的第 n 个元素是 1$\}$是递归可枚举的。

由 Yuri Matiyasevich：每一个递归可枚举集有一个 singlefold 指数丢番图方程表示。即：$p\in R$ iff $\exists!y(A(p,y)=0)$。p 和 y 都可以是多元组。这里的 p 是$\langle n,x_1\rangle$，y 是$\langle x_2,\cdots\cdots x_m\rangle$。

因此,丢番图方程 $A(n,k,x_2,\cdots\cdots x_m)=0$ 有唯一解 $x_2,\cdots\cdots x_m$ 如果 ω_k 的第 n 个元素是 1;否则，无解。所以，如果 Ω 的第 n 个元素是 1，则 $A(n,x_1,x_2,\cdots\cdots x_m)=0$ 有无穷多解 $x_1,x_2,\cdots\cdots x_m$;否则，有穷。

希尔伯特第十问题的否定解决告诉我们不存在普遍算法判定任意丢番图方程是否有整

数解，此定理则显示情况更糟，对于指数丢番图方程 $A(n,x_1,x_2,\cdots\cdots x_m)=0$ 中一个参数 n 的改变，方程是否有无穷解的波动竟然是算法随机的！

定理 4.4[4] FAS 是一个每个在其中可证的算术命题都真的可递归公理化的形式系统，如果形如“H(s)≥n”的命题可证那么一定有 $n\leqslant H(axioms)+O(1)$。

证明：按证明长度枚举 FAS 的定理，对任意的正整数 k，定义 s*为形如“H(s)≥n”的且满足 n>H(axioms)+k 的枚举序列中的第一个定理。由 s*的定义方式借助 Church-Turing 论题，存在部分递归函数 ϕ 使得 $s^*=\phi(axioms,H(axioms),k)$

由定理 2.2 $H(s^*)\leqslant H(axioms,H(axioms),k)+c_\phi\leqslant H(axioms)+H(k)+O(1)$

$\therefore H(axioms)+k<H(s^*)\leqslant H(axioms)+H(k)+O(1)$

$\therefore k<H(k)+O(1)$

然而，当 $k\geqslant k_0$ 时，上面不等式为假，其中常数 k_0 只依赖于 FAS 的推理规则。所以，s* 不存在,即对某个特定的 s，FAS 推不出“$H(s)>H(axioms)+k_0$”。

定理 4.5 任何包括初等数论可递归公理化的协调 FAS 只能判定 Ω 的有穷个元素。

证明：假设 FAS 包含递归可枚举的形如“Ω 的第 n 个元素是 0”、“Ω 的第 n 个元素是 1”的真命题。如果 FAS 能确定 k 个 Ω 的元素，就可以得到包含 Ω 的测度为 2^{-k} 的覆盖 A_K。枚举 FAS，直到 Ω 的 k 个元素都被确定下来。如果 FAS 能确定 Ω 中无穷个元素，那么对任意的 k 都可以进行上面的操作，而这与 Ω ML random 矛盾。

定理 4.6 对于 i≥0,考虑 r.e.随机实数 $\alpha=0.a_0a_1a_2\cdots a_{i-1}a_ia_{i+1}\cdots$其中 $a_0=a_1=a_2\cdots=a_{i-1}=1,a_i=0$。可以能行的构造通用 Chaitin 机 U（依赖于 ZFC 和 α ）满足如下条件：

1.在 PA 内可证明 U 的通用性。

2.ZFC 只能判定 Ω_U 的前 i 项。

3.$\alpha=\Omega_U$

取 i=0 时，则 ZFC 不能判定 Ω_U 的任何项。

综合定理 4.3、4.5，任何形式理论只能对有穷多个 n 确定丢番图方程 $A(n,x_1,x_2,\cdots\cdots x_m)=0$ 是有有穷解还是无穷解。

由定理 4.1，假如给定了 Ω_{10000}，那么长度短于 10000 的程序的停机问题将得到解决，事实上，这些程序将包括为费马大定理、哥德巴赫猜想、黎曼假设等重要命题寻找反例的程序！对任意的 FAS，如果 $H(FAS)<10000$，Ω_{10000} 将可以判定任何命题相对于 FAS 可证明、可证伪还是独立。因此，Ω 可以看作是最聪明的数！一个终极 oracle!?

5 没用的 Ω 与逻辑深度

毋庸置疑,怀尔斯证明费马大定理的著作非常有深度,即使是数学家,能读懂的也不多。但它的算法复杂性却不高,因为再难的的结果也是从初始的那些简单定理推导出来的。那么,怀尔斯大作令人敬畏的“深度”从何而来呢？它来自研究者长年累月创造性的辛勤劳动。如果一个数列不是一览无遗而是很慢的泄露自己的秘密,即只有在它很长的初始段被分析后它

[4] 本定理的证明思想可对比“不存在最无趣的自然数”和“n 是不能被少于二十个字定义的最小自然数。”

的规律性才能被慢慢揭示出来那么这个数列就应该被称为有深度的。

因此，最直接的想法是定义深度为从 x^*计算出 x 的最短步骤数，但可能存在比 x^*略长的程序却能更快的生成 x。所以要协调程序长度与计算时间。放宽对程序长度的要求，考虑几乎最短的程序，定义 x 在误差 2^{-b} 界内具有深度 d，如果至多比 x^*长 b 个比特的程序 p 在 d 步内计算出 x。即 $2^{-|p|}/2^{-H(x)} \geqslant 2^{-b}$。但直接这么定义仍有不妥之处，更合理的定义如下：

首先定义：$P_t(x)=\sum_{Ut(p)=x} 2^{-|p|}$ 其中 Ut(p)=x 意味着 U 在 t 步内计算出 x 并停机。

定义：在程度 $\epsilon=2^{-b}$ 下，x 的深度为：

$depth_\epsilon(x)=\min\{t:P_t(x)/P(x) \geqslant \epsilon\}$

x 是（d,b）-深的，如果 d= $depth_\epsilon(x)$，$\epsilon=2^{-b}$。

定义 x 是 b-可压缩的，如果$|x^*| \leqslant |x|-b$。否则称 x 是 b-不可压缩的。

如果 x 是(d,b)深的，则 x 的算法概率中大约 $1/2^{b\pm\delta}$ 的部分（对小的 δ ）来自在 d 步内运行的程序。

定理 5.1 串 x（大约）是（d,b）-深的，当且仅当 d 是打印出 x 的任何 b-不可压缩的程序所需的最短时间。精确形式是：$1/2^{b+H(d)+O(1)} \leqslant P_d(x)/P(x) \leqslant 1/2^{b-O(1)}$。证明略。

如果 x 是(d,b)深的，则 U 至少需要 d 步从 x^*中计算出 x 来。

定义：在任意程度上，x 是 d-浅的，如果它的深度不会超过 d。任何串 x 必须至少要用|x|步才能打印出。如果 x 是（$n\pm O(1)$）-浅的（在任意程度上），则简称 x 是浅的。

1^n是浅的，常数长的不可压缩程序将在 n 步内打印出它。

长度 n 的随机串也是浅的，因为它能够被长度 $n\pm O(1)$的最短程序 x^*在 n 步打印出。对于 Ω,由定理 4.2, $H(\Omega_n)>n-O(1)$，它与随机串是递归不可区分的，它也是浅的。结合上一节，我们知道，Ω 是不可计算的，虽然逼近 Ω 的前几步操作是容易的，因为前面的短程序要么是语法错误的要么很快停机要么很容易看出它永不停机，但想要计算出 Ω_{10000} 显然也并不现实。另一方面，即使知道了 Ω_n，判定那些长度短于 n 的程序耗费的时间 t(n)的增长速度将快于任何递归函数[5]。因此，Ω 无比诱人，却毫无用处！

6 Digital philosophy 与不完全

程序 → 计算机 → 输出

公理 → FAS → 定理

科学理论 → 推演 → 经验材料

DNA → 演化 → 有机体

终极真理 → 上帝 → 宇宙

[5] 可看作另一种形式的 BusyBeaver 函数。

上图作为一个不严格的类比，可以大致表达某种还原主义的想法，从左到右的方向是一个从简单到复杂的衍生过程，从右到左的方向是一个从复杂到简单的压缩的过程。

科学是压缩的艺术。压缩应该尽量满足 Occam 剃刀的要求：用尽量少的理由解释尽量多的现象[6]。俗语说“读书是一个把厚书读薄薄书读厚的过程”，所谓厚书读薄即是压缩，“压缩”就是“理解”，科学是对自然的理解，是从大量的经验材料抽象出规则，并保证原先的材料能被这些规则蕴涵，而薄书读厚即是从规则中推导出更多的内容，甚至达到一个推演封闭的集合，即一个理论。在实际生活中由于人的创造性，往往可以衍生出新的问题，从而无限继续。

从前面第 4 节的分析，我们知道，有无穷多的真命题具有太高的复杂性，它们不可能被“压缩”进一个 FAS，从而再从其中中推导出来，例如，类似判定 Ω 的二进制展开式的某一位是 0 还是 1 这种形式的命题，不但超出了任何可递归公理化的 FAS 的界限，而且因其随机性，要得到它们的最好方式恐怕是：直接把它们作为公理添加到系统中！

总之，“不劳无获！”——想得到更多的输出，就需要投入更多的输入！证明越多，假设越多！

比如，P≠NP，在经历了这么多努力都看不到解决它的希望的时候，最好的出路也许是：直接把它作为我们的公理。

可以证明，对于有限序列，非随机数的集合是一个 simple set，随机数集就是一个不包含递归可枚举子集的无穷集。实数轴上随机数的测度为 1。随机是如此普遍以至于 Ω 的发现者 Chaitin 感慨：“上帝不仅在量子力学里掷骰子，而且在算术里掷骰子”。所以，有理由大胆猜测整个宇宙都是随机的。但是，从这种观点看，上图中最后一行的终极真理绝不会简洁（如果存在的话），它甚至跟整个宇宙一样臃肿。假如真像 digital philosophy[7]假设的那般——宇宙就是 0,1 序列——那么如果宇宙是随机的，为什么现代科学的发展一再向我们保证，自然是可以认识的，人类的存在绝不是一场悲剧？

因为可以严格证明，即使对于有限随机序列，在某种程度的高复杂性下必然会出现某些长度的有规律的块（如连续 n 个 1 的块），最随机的序列必然包含对数阶长度的非常规则的子序列。而无限随机序列有很多很好的性质，如重对数律、Borel normal[8]等，任意有限长的有规律的块都可以在其中找到，甚至会出现无限次。也许我们所处的世界可能是极大随机的宇宙中一片规则的绿洲[9]。正如维纳满怀激情的宣言——“面对着大自然朝着无序发展的不可挡的趋向，我们宣告自己的本性，建立起一个有组织的飞地，这是对众神及他们所强加给我们的铁一般的必然性的鄙视，这是不幸之所在，但这也是荣耀之所在”[10]。

---

[6]这可以表述为一种最短描述长度原则：
解释一组数据 D 的最好理论 T 应该使如下两项之和最小：
1 描述理论 T 所需要的比特长度。
2 在理论 T 的协助下，对数据 D 编码所需的比特长度。
即，使 H(D|T)+H(T)最小。

[7] “it from bit!”——Wheeler. “All is computation!”——Wolfram.

[8]无限序列是 b 进制 Borel normal 的，如果对任意的 k，在长为 n 的前段中，随着 n 的无限增加，任意长为 k 的块的相对频度趋向极限 b^{-k}。Borel normal 不足以保证随机性，比如 0.1234567890101112……是十进制 Borel normal 的，但显然不随机。

[9] 对于无限随机序列的情形，这是在预设了连续统存在的前提下才能得出的结论，而 Wheeler 是不承认这种假设的。Wolfram 也不相信真随机存在，他只相信确定性随机，类似物理中的混沌，或某种极度简单规则生成极大复杂现象的伪随机，如 $X:=X^2+C$ 确定的 Mondelbrot 集等等。

[10] 引自：维纳.《人有人的用处》.商务印书馆.1978.

但估计哥德尔本人不会认为数学里充斥着随机，虽然他也认为随机概念是协调的[11]。所以从算法信息论的观点看不完全性的时候，算法复杂性、算法随机性等概念把我们引得太远，现在让我们回到最初的不完全，看哥德尔自己会得出什么结论，对于新公理的地位问题他在《罗素的数理逻辑》里提出的观点可与此互相参照：

“他（罗素）把逻辑和数学公理同自然律相比，把逻辑证据和感官知觉相比，以致公理无需必然是自明的，它们的合理性毋宁依赖于这样一个事实（恰和物理学一样）：它们使得这些‘感官知觉’的推演成为可能……对于抽象集合论的某些问题的判定，甚至对于某些有关的实数论问题，建立在某一迄今未知的思想之上的新公理将是必要的。或许，某些其他数学问题多年来所呈现的看来不可克服的困难，也是由于必要的公理还没有找到这一事实。当然，在这些情况下，数学可能会丧失许多‘绝对确定性’；但在数学基础的现代批评的影响下，这在很大程度上已经发生了。”

在《康托尔连续统问题是什么》一文中，他再次阐述了类似观点：

“即使不考虑某一新公理的内在必然性，甚至在它根本就没有什么内在必然性的情况下，关于它的真理性的概然判定从另一条道路来说也是可能的，即归纳的研究它的‘成功’。这里成功的意思是在推论上的多成果性……可能存在这样一些公理……以致不管它们是否内在必然，这些公理至少应在和任何公认的物理理论一样的意义上被接受。”

作为最强硬的柏拉图实在论者的哥德尔，把逻辑、数学与物理学类比在本体论上虽无不妥，但在认识论上，数学上寻找新公理的过程跟物理学一样求助于归纳的研究它的“成功”仿佛是向自己的不完全性结果的妥协。

也许，数学和物理的确没有我们想象的那么不同。

If mathematics describes an objective world just like physics, there is no reason why inductive methods should not be applied in mathematics just the same as in physics.——Gödel

附录：

哥德尔不完全性定理

下面给出哥德尔不完全性定理的证明梗概，采用从图灵停机定理推不完全性定理的简略证法。首先给出图灵停机定理 AIT 风格的证明。

一、图灵停机定理：停机问题不可解，即对于任何声称要判定所有图灵机程序是否停机的图灵机程序 H，都存在一个程序 P 和输入数据 I，使程序 H 不能判定处理数据 I 时，P 是否会停机。

证明：假设存在停机程序 H，那么可以构造如下程序 V：

1 输入 N。

2 生成大小不超过 N 的所有程序 $P_1, P_2, P_3 \cdots\cdots P_n$。

3 利用 H，检查 2 中所有程序是否停机，排除所有不停机的程序。

[11] Solovay 曾猜测哥德尔心目中的随机应指非序数可定义性。若这种意义上的随机集存在，由于 $L \subset HOD \subset OD \subset V$，则应有 $V \neq L$。

4 模拟 3 得到的所有停机程序的运行。

5 取 4 中最大的输出，乘以 2 作为 V 的输出。

上面构造的程序 V 显然对于任意的 N 都停机，它的长度为 logN+O(1)。取足够大的 N，有 logN+O(1)<N，所以，V 必然也是某个 P_i，它的输出是自己输出的两倍，矛盾！所以停机问题不可解。

由此易见 $K=\{a:\phi_a(a)\downarrow\}$是递归可枚举集但不是递归集。

二、哥德尔不完全性定理：Th$\mathfrak{N}$不能递归公理化。

证明：因为 K 递归可枚举，在$\mathfrak{N}$中可定义，所以有公式 σ(x)在$\mathfrak{N}$中定义 K。

$a\in K^c$ iff $(\neg\sigma(S^a0))\in Th\mathfrak{N}$

任给 a，可能行的找到$\neg\sigma(S^a0)$，所以借助 Church-Turing 论题，存在 f 使得

$a\in K^c$ iff $f(a)=\#(\neg\sigma(S^a0))\in\#Th\mathfrak{N}$

∴ $K^c\leqslant_m\#Th\mathfrak{N}$

假设 Th$\mathfrak{N}$可递归公理化，则# Th$\mathfrak{N}$递归可枚举，所以 K^c 也递归可枚举，所以 K 就是递归的。矛盾！

参考文献：

1 Calude. Information and Randomness. Springer. 2002.

2 Chaitin. Algorithmic Information Theory.Cambridge Universtity Press.1987.

3 Downey. Algorithmic Randomness and Complexity.Springer.2010.（to appear）

4 M.Li,P.M.B.Vitanyi. An Introduction to Kolmogorov Complexity and Its Applications.Springer-Verlag.(1^{st}.ed.1993 \ 2^{nd}.ed.1997 \ 3^{rd}.ed.2008)

5 Cristian S. Calude, Nicholas J. Hay. Every Computably Enumerable Random Real is Provably Computably Enumerable Random.(to appear)

6 Kohtaro Tadaki. Equivalent characterizations of partial randomness for a recursively enumerable real. (to appear)

7 Chaitin. A theory of program size formally identical to information theory.Journal of the ACM 22 pp.329-340.1975.

8 Chaitin. Information theoretic limtations of formal systems.Journal of the ACM 21.pp.403-424.1974.

9 Nies.Computability and Randomness.Oxford University Press.2009.

10 贝纳赛拉夫、普特南编. 数学哲学. 商务印书馆. 2003. 其中哥德尔《罗素的数理逻辑》、《康托尔连续统问题是什么》在本文集中收录。