# 哥德尔不完全性定理

在关于完备性的讨论中，我们最终得到了只要有 $\Phi \vdash \varphi$ 成立，那么它当且仅当 $\Phi \models \varphi$ 。也就是说，在一阶逻辑上，所有机械地在语法上证明成立的命题恰好就是在语义上可以由 $\Phi$ 中的命题推出的。而我们意识到，有时 $\Phi \vdash \varphi$ （或等价地， $\Phi \models \varphi$ ）是否成立本身就是一个难以判定的问题。以自然数为例，对于自然数我们构建地非常完善的一个公理系统称为皮亚诺算术，它包括了六条基本的加法和乘法法则以及归纳法这一条准则。如今我们发现，由皮亚诺公理出发所能证明的所有命题并不就是所有自然数上的真命题，我们能够证明存在一个自然数上的真命题，既不能被皮亚诺算术证真也不能被皮亚诺算术证伪。这就是不完全性的一个例子。一般地，我们说当 $\Phi$ 满足了一些条件时，就会出现一个命题 $\varphi$ ，使得 $\Phi \vdash \varphi$ 和 $\Phi \nvdash \varphi$ 都不成立，这些条件将在之后被严格定义，但它大概刻画了系统的描述能力。一个系统一旦拥有描述自身的能力，也即它能推出具有*自指性质*的公式后，就会出现这样的情况。这一切的本质可以用"说谎者悖论"来解释明白：小明说"我在说谎"，那么如果小明在说谎他就没有说谎，如果小明没说谎那他就在说谎。这就是日常语言中一个最简单的既不能证真也不能证伪的命题的例子。我们将会看到，这一切与实数不可数，停机问题不可计算等等，都是由对角线argument产生的矛盾推出的。

所以当我们讨论完全性时，我们其实就是在给定的公理体系 $\Phi$ 中讨论是否能找到 $\varphi$ 的一个证明。对于拥有智能的人类，可以灵光一闪想到这个证明，而在很多时候这做不到。如果我们机械化地讨论这个问题，我们其实就是在讨论：是否存在一个能自动寻找证明的机器？我们可以设想这样一个程序，它会枚举世界上所有的证明，由于世界上所有的证明都必须是有限的，因此如果我们有一个确信它可证的命题，它的证明总会在有限时间内被找到。对于命题 $\varphi$ ，如果 $\varphi$ 和 $\neg\varphi$ 中总有一个是可证，那么我们可以同时开启两台机器分别寻找 $\varphi$ 和 $\neg\varphi$ 的证明，总有一天会成功。但这里有两个问题，首先不一定枚举到的每一个证明都是符合我们需要的，我们还需要检查证明用到的*前提是否都落在系统* $\Phi$ 内，而这件事必须是能够在*有限时间*内做到的；另外，我们并不存在一条规律保证 $\varphi$ 和 $\neg\varphi$ 中总有一个是可证，而恰恰相反哥德尔不完全性定理就描述了这样的公理系统，里面有一个命题 $\varphi$ 和 $\neg\varphi$ 都是不可证的。

# 可判定性

先来讨论我们遇到的第一个问题，我们称它为"可判定问题"，我们本质上要回答的是对于一个给定的集合，是否存在一台机器对于任何一个给定元素能在有限时间判定这个元素是否在集合内。在这里，我们所说的机器就是指图灵机。但为了讨论的方便，我们在这里使用一个图灵机的等价模型寄存器机。它有 $m$ 个用来存放符号串的内存，能够写入某个内存末尾加字符、减字符、跳转、打印和停机五种指令。一个寄存器机程序（简称程序）就是有限条寄存器机上的指令（且最后一条为停机）的集合。程序的输入是指程序开始时第一个内存上的符号串。程序的输出是指程序打印出来的符号串。一个元素可以理解为一个字符串，一个集合就是一个字符串的集合。于是我们可以严格定义可判定性：我们称一个字符串集合 $W$ 是可判定的，当且仅当存在一个程序在输入某个字符串时，如果这个字符串属于 $W$ 就输出空串，否则输出非空串。

我们假定字符集的大小是有限的，那么我们总是可以枚举一个集合里的所有元素的。对于一个给定的元素，我们运行枚举程序，如果它是集合里的元素那么有限时间内我们一定能停机。而问题在于如果它不在集合内，那么我们可能永远不会停机了。因此我们显然不能说所有集合都是可判定的。（注意我们不能要求枚举程序能按照某种顺序枚举，比如字典序。如果一个集合能按字典序枚举，那么就很容易判定不在集合内的元素了）那么是否存在一个不可判定的集合？

现在我们要仿照说谎者悖论的例子来构造一个反例。这意味着，我们想让我们构造的集合能具有自指性质。这里的自指就存在于字符串与判定程序之间。一个程序本身就是由一些字符编码而成的，因此一个程序与一个字符串本身是没有界限的！在字典序意义下，字符串和它的字典序一一对应，因此我们本质上只是在讨论自然数上的情况。任何一个程序拼接成某个有限字符集下的一个有限字符串，取这个字符串在这个字符集下的字典序，称为程序的哥德尔编码。程序自身（编码后）可以作为输入——程序本身可以作为输入了——被输入进一个程序，这样就完成了自指。对于这个字符集上的所有程序我们都可以这么做。我们把所有可能的程序都输入它自身，对于所有这些程序，有一些会运行终止有一些不会。现在我们把这里面所有不停机的程序收集进集合 $\Pi_{\text{halt}}$ ，我们证明这个集合就是不可判定的！假如存在判

定这个集合的程序 $\mathbb{P}_0$ ，那么 $\mathbb{P}_0$ 会在输入的程序停机时输出空串，不停机时输出非空串。现在我们对 $\mathbb{P}_0$ 做一点修改，每当 $\mathbb{P}_0$ 要输出时就强制停机，而当 $\mathbb{P}_0$ 要正常停机时就让它死循环。这样我们得到了一个完全合法的这个字符集上的程序 $\mathbb{P}_1$ ，并且它也可以输入自身。那么现在让 $\mathbb{P}_1$ 输入自身，假如它要停机那么它就不停机，如果它不停机他就会停机。这样我们就得到了一个说谎的小明一样不可能存在的程序，矛盾。所以 $\Pi_{\text{halt}}$ 是一个不可判定的集合。

我们已经认识到，一个程序不仅是一个程序，它本质上只对应着一个字符串，甚至一个自然数。从程序出发建立双射，把问题规约为 $\Pi_{\text{halt}}$ ，我们就可以找到更多不可判定的集合。而从刚才的证明的过程中我们会发现，这些集合之所以不可判定也都是因为具有了自指性质。下面我们要说明一阶逻辑中valid(恒真)命题的集合是不可判定的，也即集合 $\{\varphi \in L_0^{S_\infty} \mid \models \varphi\}$ 是不可判定的。我们把这个问题规约为 $\Pi_{\text{halt}}$ 。我们想对于每个程序 $\mathbb{P}$ ，构造公式 $\varphi_{\mathbb{P}}$ 满足 $\mathbb{P} : \square \to \text{halt} \iff \models \varphi_{\mathbb{P}}$ 。假如所要证的集合是可判定的，那么对于一个给定程序 $\mathbb{P}$ ，我们可以根据它写出 $\varphi_{\mathbb{P}}$ ，那么只需判断 $\varphi_{\mathbb{P}}$ 是否valid（枚举没有前提的证明即可）就可以判定 $\mathbb{P}$ 是否停机了，矛盾。于是问题转化为了如何用一阶逻辑的valid命题来*刻画*程序的停机行为？这里，我们就可以充分发挥寄存器机的优势。寄存器机的程序状态只取决于寄存机上储存的值，也就是程序状态本质上就是一个 $n$ 元关系。程序停机当且仅当最终到达了最后一条指令，而每一行指令就是 $n$ 元关系的跳转关系。由此，只需要依据程序的每一条指令利用推出符号写出相应的跳转公式，最终要求到达最后一条指令，这就是一条等价刻画停机行为的公式了。在整个过程中，我们实际上证明了可以用一些一阶逻辑的valid命题精准刻画程序的停机行为，而停机行为是不可判定的，因此一阶逻辑的valid命题集合也是不可判定的。

> “集合 $\{\varphi \in L_0^{S_\infty} \mid \models \varphi\}$ 是不可判定的”这一结论还意味着“集合 $\{\varphi \in L_0^{S_\infty} \mid \varphi \text{ is satisfiable}\}$ 是不可枚举的”。valid公式是可判定的因此是可枚举的，那么假设另有一个程序能枚举可满足的公式，那么对于任意一个公式 $\varphi$ ，我们同时运行这两个枚举程序。如果 $\varphi$ 是valid的，那么它就一定会被第一个程序枚举到，否则意味着存在 $\mathfrak{I}$ 使得 $\mathfrak{I} \not\models \varphi$ ，也即 $\mathfrak{I} \models \neg\varphi$ ，这说明 $\neg\varphi$ 是可满足的，它会被第二个程序枚举到。这样我们就有一个判定 $\varphi$ 是否valid的程序了，这就推出了矛盾。
>
> 我们知道 $L_0^{S_\infty}$ 是可数的，因此集合 $\{\varphi \in L_0^{S_\infty} \mid \varphi \text{ is satisfiable}\}$ 也是可数的。于是我们得到了一个例子：一个集合是可数的但却是不可枚举的。这表面上似乎是违反直观的，似乎一个集合只要是可数的我们就可以枚举它。但事实是一个集合的不可枚举性不仅取决于集合的势，还取决假若它作为一个可数集的子集，这个*子集关系*是难以刻画的。

# 完全性

我们将证明，自然数上成立的所有命题集合这一系统是不可公理化的。为此，我们先严格定义一阶逻辑中的理论和公理的概念。一个理论是一个特殊的公式集，它首先是一个可满足的sentence集合（不能有自由变元），重要的是，它要对推出关系是封闭的。对于任何一个sentence集合 $T$ ，我们可以取它推出关系上的闭包，记为 $T^{\models}$ ，这一定是一个理论。而对于任何一个模型 $\mathfrak{A}$ ，集合 $Th(\mathfrak{A}) = \{\varphi \mid \mathfrak{A} \models \varphi\}$ 一定是可满足的而且是对推出关系封闭的，因此是一个理论。如果一个理论 $T$ 能找到一个*可判定*集合 $\Phi$ 使得 $T = \Phi^{\models}$ ，就称这个理论是可公理化的，因为最小的那个 $\Phi$ 就可以理解为这个理论的公理。对于自然数模型 $\mathfrak{N} = (\mathbb{N}, +, \cdot, 0, 1)$ ， $Th(\mathfrak{N})$ 是一个理论，它描述所有在自然数上成立的事实。我们要证明 $Th(\mathfrak{N})$ 不可公理化。（我们根据皮亚诺公理 $\Phi_{PA}$ 得到的理论 $\Phi_{PA}^{\models}$ 一定满足 $\Phi_{PA}^{\models} \subseteq Th(\mathfrak{N})$ 。一旦我们证明了 $Th(\mathfrak{N})$ 不可公理化，就证明了 $\Phi_{PA}^{\models} \subsetneq Th(\mathfrak{N})$ ，也就是说一定存在一个数论上的真命题是无法由皮亚诺公理推出的。）

如果一个理论 $T$ 满足，对于任意的 $\varphi$ 都只有 $\varphi$ 和 $\neg\varphi$ 中的一个属于 $T$ ，就称 $T$ 是完全的。在自然数上成立的命题集合 $Th(\mathfrak{N})$ 就是一个完全的理论。

一个完全的可公理化理论是可判定的。根据完全性，对于给定的 $\varphi$ ，要么有 $\varphi$ 成立要么有 $\neg\varphi$ 成立。我们可以枚举所有的证明，检查每个证明用到的前提是否在公理集合内（可公理化要求公理集合是可判定的，因此可以这么做），只有这样的证明才是我们需要的。在有限时间内我们一定会枚举到 $\varphi$ 或 $\neg\varphi$ ，这样就能判定 $\varphi$ 的真假了。现在，由于 $Th(\mathfrak{N})$ 是完全的，只需证明 $Th(\mathfrak{N})$ 不可判定，就能推出 $Th(\mathfrak{N})$ 不可公理化。它的证明依然是停机问题！我们依然希望对于任何程序 $\mathbb{P}$ 构造一个公式 $\varphi_{\mathbb{P}}$ 使得

$\mathfrak{N} \models \varphi_{\mathbb{P}} \iff \mathbb{P} : \square \to \mathrm{halt}$。与一阶逻辑中valid命题不同的是，我们现在不能使用 $n$ 元关系符号，只能使用自然数以及算术符号。但我们的目标依然是刻画程序的可达状态。为此，可以考虑把一个程序的所有可达状态串起来编码成一个自然数。但困难在于，由于没有了 $n$ 元关系，我们必须用存在量词来描述所有经历的程序状态。假设程序运行 $s$ 步，就需要 $n \cdot s$ 个存在量词。然而，对于所有的程序的 $s$，并不存在一个常数作为上界。因此想要把所有状态都放进一阶逻辑的公式里是做不到的。而如果我们能把一个给定的任意长度的自然数串编码成一个数字，这样它就可以被放入存在量词里了。这就要用到哥德尔的 $\beta$ 函数。对于一个有限长度的自然数串，我们可以找到一个比串里的每个自然数都大的一个质数 $p$ 作为进制，结合用来定位的自然数编码成一个 $p$ 进制数。根据进制表示的唯一性容易找到一阶逻辑的公式来刻画这个自然数串。这样就完成了证明。

# 哥德尔第一不完全性定理

一个程序的停机行为可以用自然数公式来描述，这直接意味着任何可判定的 $n$ 元关系可以用自然数公式来描述，因为可判定本质上就是程序的停机行为。对于系统 $\Phi$，称 $n$ 元关系 $R$ 是 $\Phi$ 中*可表示*的，如果 $R$ 能找到一个关于 $n$ 个变量的自然数命题 $\varphi$，使得 $R$ 成立 $\implies \Phi \vdash \varphi$，$R$ 不成立 $\implies \Phi \vdash \neg\varphi$。如果 $\Phi$ 上的所有关系（包括函数）都是可表示的，就说 $\Phi$ 是*允许表示*的。*哥德尔第一不完全性定理指出：一个自然数算术上的系统 $\Phi \subseteq L^{S_{ar}}$ 如果一致、可判定并且允许表示，那么就存在 $\varphi$ 使得 $\Phi \vdash \varphi$ 和 $\Phi \vdash \neg\varphi$ 都不成立*。皮亚诺公理 $\Phi_{PA}$ 就是自然数算术上的一个允许表示的系统，它也是一致的可判定的，所以如同我们已经看到的，存在一个数论上的真命题无法由皮亚诺公理推出。

反证法，假设一个自然数算术上的一致、可判定、允许表示的系统 $\Phi$ 是完全的。那么我们可以枚举所有证明，得到证明意义下的闭包 $\Phi^{\vdash}$ 是可判定的（因为 $\Phi$ 是可判定的）。现在我们想要通过构造自指来推出矛盾。显然算术公式的全集 $L^{S_{ar}}$ 是可枚举的，我们以自然数下标把它记为 $\varphi_i$，这本质上也看作对算术公式的一个自然数编码，记 $[\varphi_i] = i$。下面我们证明对于任何一个恰好允许一个自由变元的算术公式 $\psi$，我们始终能找到一个 $\varphi$ 使得 $\Phi \vdash \varphi \leftrightarrow \psi([\varphi])$ 成立。构造函数 $F(n, m)$，当且仅当编码为 $n$ 的算术公式 $\varphi_n$ 恰好允许一个自由变元时，函数返回 $\varphi_n(m)$ 的编码，否则返回0。由于 $\Phi$ 允许表示，$F(n, m) = l$ 可以被表示为 $\Phi \vdash \varphi_F(n, m, l)$。对于给定的 $\psi$，我们令 $\chi(v_0) = \forall x(\varphi_F(v_0, v_0, x) \to \psi(x))$，它表示第 $v_0$ 个算术公式代入 $v_0$ 时（自指！）编码为 $x$ 成立时能推出 $\psi(x)$。而 $\chi$ 本身也是一个算术公式，假设它有编码 $n$，现在我们构造 $\varphi = \chi(n)$。也即 $\varphi = \forall x(\varphi_F(n, n, x) \to \psi(x))$ 我们验证 $\Phi \vdash \varphi \leftrightarrow \psi([\varphi])$ 成立。左推右，$F(n, n) = F([\chi], n) = [\chi(n)] = [\varphi]$，根据可表示的定义有 $\Phi \vdash \varphi_F(n, n, [\varphi])$，于是根据 $\Phi \vdash \varphi$ 直接根据 $\varphi$ 的定义得到 $\Phi \vdash \psi([\varphi])$。右推左，已知有 $F(n, n) = [\varphi]$ 成立，那么根据函数映射的唯一性写出 $\Phi \vdash \forall z(\varphi_F(n, n, z) \to z \equiv [\varphi])$，如果已知 $\psi([\varphi])$ 成立，那么就可以由 $\varphi_F(n, n, x)$ 推出 $x = [\varphi]$，所以 $\psi(x)$ 成立。而 $\forall x(\varphi_F(n, n, x) \to \psi(x)$ 恰好是 $\varphi$ 的定义，所以我们证明了 $\Phi \vdash \varphi$。这个定理称为不动点定理，它是哥德尔定理——自指——的核心。对于 $\Phi$，我们取证明意义下的闭包 $\Phi^{\vdash}$，编码后这就是自然数的一个子集，因此是一个一元关系。如果这个一元关系是可表示的，就称 $\Phi^{\vdash}$ 是可表示的。现在我们发现对于一个一致的可表示的 $\Phi$，$\Phi^{\vdash}$ 一定是不可表示的。因为如果它是可表示的，那么对于任意一个 $\varphi$，就有某个函数 $\eta$ 使得 $\Phi \vdash \varphi$ 就能推出 $\Phi \vdash \eta([\varphi])$，$\Phi \nvdash \varphi$ 就能推出 $\Phi \vdash \neg\eta([\varphi])$。根据 $\Phi$ 的一致性，直接写出 $\Phi \nvdash \varphi \iff \Phi \vdash \neg\eta([\varphi])$。而根据不动点定理，$\Phi \vdash \neg\eta([\varphi])$ 等价于 $\Phi \vdash \varphi$，矛盾。综上，闭包 $\Phi^{\vdash}$ 是不可判定的，我们完成了哥德尔第一不完全性定理的证明。

# 哥德尔第二不完全性定理

我们自然要追问，如果自然数算术上有一个真命题是不可证的，那么这个神奇的命题究竟是什么？哥德尔第二不完全性定理回答了这个问题。

我们构造一个二元关系 $(n, m)$，它只有在 $\Phi \vdash \varphi_n$ 时成立，$m$ 是 $\varphi_n$ 的证明的编码（证明是可枚举的）。这个二元关系是可表示的，它刻画一个命题是否可证。由此我们定义公式 $\mathrm{DER}_{\Phi}(x)$ 表示编码为 $x$ 的公式是否可证。对这个函数运用不动点定理，能得到一个 $\varphi$ 满足 $\Phi \vdash \varphi \leftrightarrow \neg\mathrm{DER}_{\Phi}([\varphi])$。那么如果 $\Phi$ 是一致的，$\Phi \vdash \varphi$ 就会立即和 $\varphi$ 可证矛盾，因此一定有 $\Phi \nvdash \varphi$。

对于自然数有一个方便之处，我们可以直接用 $\neg 0 \equiv 0$ 不可证来刻画一致性。因为 $\Phi \vdash 0 \equiv 0$ 是始终成立的。如果 $\Phi$ 一致，那么一定成立 $\Phi \nvdash \neg 0 \equiv 0$；而如果有一个命题证不出来，它就一定是一致的。所以我们可以用公式 $\neg \mathrm{DER}_\Phi([\neg 0 \equiv 0])$ 来等价表示 $\Phi$ 的一致性。那么我们可以把刚才得到的 $\Phi \nvdash \varphi$ 的结论用公式本身来刻画！而这一结论的可证性可以用皮亚诺公理来验证。于是我们有 $\Phi \vdash \neg \mathrm{DER}_\Phi([\neg 0 \equiv 0]) \to \neg \mathrm{DER}_\Phi([\varphi])$。而 $\Phi \vdash \neg \mathrm{DER}_\Phi([\neg 0 \equiv 0])$ 是不可能的，这样就会得到 $\Phi \vdash \neg \mathrm{DER}_\Phi([\varphi])$ 了，根据不动点定理就有 $\Phi \vdash \varphi$，与刚才得到的 $\Phi \nvdash \varphi$ 矛盾。

综上，我们得到了 $\Phi \nvdash \neg \mathrm{DER}_\Phi([\neg 0 \equiv 0])$，也即 $\Phi$ 证不出用来刻画它自身是一致的那个公式，而这个公式根据我们的假设就是成立的。简而言之，一个一致的系统不能证明它自身是一致的。