

Computability Notes I

Nibbles



1 可计算性理论（一）

大自然到底能否究诘呢？18世纪的歌德在其诗作中发出这样的诘问.人类是否能穷尽大自然的可能，算尽天机？早在公元前1400年的商代，人们已经开始用算筹表示数字.而在战国初年，人们开始使用数字进行四则运算、乘方以及开方等运算.可以说，从这时起，算法已经开始发展了.无论是乘方与开方，抑或是欧几里得在《几何原本》中提出的各类算法，都是由有限一组指令指定；计算经过一系列有限步骤，最终得出答案；它以确定性、完全规定的方式进行，没有任何人类智能或随机设备的参与，这也是我们对算法的通常定义.

自19世纪中叶开始，数理逻辑蓬勃发展，人类的数学能力迎来了一座高峰.许多千年未解之谜逐渐得到厘清和解答，数学家们如饥似渴地不断寻找下一个难题，再尝试云集众智力克之.在 1930 年之前，人们从未考虑可计算函数（有效可计算函数）的严格定义.相反，以希尔伯特为代表的数学家热衷于获得针对特定问题的算法解，例如希尔伯特的第十个问题：丢番图方程解，或者 Entscheidungsproblem（判定问题），即找到一种普适的算法来判定一阶谓词逻辑中哪些公式是有效的.

然而，19世纪末发现的几条悖论戳破了一切问题皆有算法解决的美好泡沫.哥德尔的第一不完备定理更是进一步表明，任何包括初等数论的可有效公理化的形式系统 T 要么不一致，要么不完备.人们从此开始重新审视算法的解决能力的边界，以不同形式提出了可计算性的概念.1936 年，图灵在其论文《论可计算数及其在判定问题上的应用》中，提出了图灵机的概念，并证明了图灵机的可计算函数与可计算数是一致的.这一结果被称为图灵-判定定理.图灵机的提出，使得可计算性的研究有了一个统一的框架，也为后来的计算机科学奠定了基础.

1.1 计算问题

为了研究可计算性，首先需要定义计算问题。

定义 1. 计算问题是指，对于任意给定良定义的输入，求满足条件的输出的问题。

所谓良定义的输入，是指其没有歧义、含义明确并且取值确定。良定义的输入确保同一个输入能获得相同的输出。例如，对于一个给定的图，求其从起点到终点的最短路径。则输入时给定的图及起点和终点，输出需要满足的条件是最短路径。因此这是一个计算问题。

1.2 可计算问题

定义 2. 对于任意计算问题，如果存在一个算法能够解决该问题，则称该问题是可计算的。

注意这里所说的算法，即指前文所提的通常定义的算法，由有限一组指令指定、计算步骤有限，并能得出答案。自然要问，是否所有的计算问题都是可计算的呢？如前所述，希尔伯特执迷于寻找判定问题的通用算法，即能够判定任意一阶谓词逻辑公式是否有效的算法。他曾在自己撰写的课本《数理逻辑原理》一书中这样写道：“一旦我们找到了一种能够对任意给定的逻辑表达式都能在有限步内判定可满足性或者有效性的程序，我们就能解决判定问题。……判定问题必须作为数理逻辑的主要问题来考虑。” (Böerger, Grädel, & Gurevich 1997). 既然大师如此重视，那么我们不妨就从判定问题开始研究。

2 判定问题

判定问题的输入是一个命题逻辑公式，输出是该公式的可满足性。对于通常讨论的算法范围，判定问题的输入也可以是任意形式。例如，输入是一个图，输出是该图是否存在从起点到终点的路径。

用朴素集合论可以很好地建模判定问题。由于判定问题的输出只有可满足/不可满足或者真/假，算法实际上将输入分为了两类，一类是可满足，另一类是不可满足。因此判定问题也可以表述如下：给定集合 S 和输入 I ，求是否有 $I \in S$ 。

2.1 形式语言理论

尽管算法的输入有很多种形式，如文字、图、图片、音频、视频等，但是这些输入都可以编码为字符串。例如图可以用邻接矩阵或者邻接表来表示，而邻接矩阵或者邻接表都可以编码为字符串；图片可以将各个通道的矩阵首尾相接；音频则是编码为音频信号的数字采样值。因此，所有判定问题都可以解释为：给定一个字符串，求该字符串是否属于某个字符串的集合。

同样一串字符，对英文使用者可能难以理解，但中文使用者看了之后可能会心一笑。字符串是否属于集合的问题，与语言问题有诸多相似之处。更进一步，我们可以将字符串看作是语言的句子，将字符串的集合看作是一门语言。这样，判定问题就可以转化为语言问题：给定一个语言 L ，对输入字符串 w 是否有 $w \in L$ 。这也就是所谓的形式语言理论。

定义 3. 给定字母表 Σ ，则 Σ 中的任意字符构成的有限序列称为字符串。所有字符串的集合记为 Σ^* 。 Σ^* 的子集称为语言。

到这里，判定问题的可计算性问题也就转换为了语言的可识别性问题.即，对于任意语言 L ，是否存在一个算法，能够判定任意字符串 w 是否属于 L . 如果计算机能够识别任意的语言，那么它就能解决判定问题，完成了希尔伯特的梦想.但是，这样的计算机是否存在呢？且听下回分解.