

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/315568710>

□□□□□□□\_□□□□□□□□□□□□□□□□

Article · March 2017

## CITATIONS

0

READS

188

1 author:



Peng Chen

Beijing Language And Culture University

18 PUBLICATIONS 3 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



17ZDA027 [View project](#)



“[XXXXXXXXXXXXXXXXXXXX](#)”[X XXXXX 17ZDA028](#) [View project](#)

# 逻辑的计算进路

## ——从莱布尼茨到图灵的逻辑发展

陈 鹏<sup>1,2</sup>

(1. 北京语言大学 信息科学学院, 北京 100083; 2. 中国社会科学院 哲学所, 北京 100732)

**摘要:** 现代逻辑肇始于莱布尼茨, 在布尔和弗雷格处发生了分流, 形成了所谓的逻辑的代数传统和逻辑的语言传统。然而, 无论是代数传统亦或者是语言传统都与莱布尼茨的逻辑学纲领(即“作为科学、数学和哲学”的基础)相去甚远。在图灵机理论中, 图灵核心阐述了“自动机”和“指令表语言”这两个概念, 这两者很好地契合了莱布尼茨关于“理性演算”和“普遍语言”的构想, 可以说, 图灵机理论最为接近莱布尼茨的逻辑学纲领, 也跳出代数传统和语言传统, 另辟蹊径, 形成了一种新的“作为计算的逻辑”传统(也可以称为“计算传统”)。“作为计算的逻辑”实质上是一种“主体转向”, “以往的逻辑”是当仁不让地以人类为主体, 研究的对象是人的思维、自然语言种种, “作为计算的逻辑”则是将计算机作为信息处理的主体, 研究的是计算机的处理方式以及人与计算机的互动关系。

**关键词:** 逻辑学; 莱布尼茨纲领; 图灵机; 作为计算的逻辑

**中图分类号:** B81 - 0 **文献标识码:** A

## 引 言

如果谈及逻辑与计算的关系, 大多数人都会认同逻辑与计算彼此紧密关联, 例如, 美国计算机科学家马纳(Manna)就曾经提出过“逻辑即计算机科学的演算”的观点<sup>[1]67-72</sup>, 此外, 甚至还有人认为“计算本质上就是逻辑”, 例如, 我国著名数理逻辑学家莫绍揆指出“事实上, 它们[程序设计]或者就是数理逻辑, 或者就是用计算机语言书写的数理逻辑, 或者是数理逻辑在计算机上的应用”<sup>[2]i</sup>。从某种意义上来说, 逻辑之于计算的重要性怎么强调都不过分, 这些主张可以通过如下论据来为之辩护:

1. 从计算机的发展历史来看, 计算科学起源于逻辑学

追溯现代计算机科学的起源, 应该说, 它与逻辑有着密不可分的关系。众所周知, 自从罗素与怀特海共同撰写《数学原理》之后, 兴起对数理逻辑的

研究, 人们甚至期望以逻辑为基础, 构建整个数学, 乃至科学大厦。在这种逻辑主义的驱使下, 不可避免地需要对“能行可计算(effectively calculable)”概念进行形式化。在“能行可计算”概念的探索中, 丘奇、哥德尔和图灵几乎在同一时间给出完全不同且又相互等价的定义。丘奇发明了Lambda演算, 用来刻画“能行可计算”。哥德尔提出“一般递归函数”作为对“能行可计算”的定义。图灵则通过对一种装置的描述, 定义“能行可计算”的概念, 这种装置被后人称作“图灵机”, 这正是现代计算机的理论模型, 标志现代计算机科学的诞生。

2. 逻辑成为计算机软硬件系统的理论基础

布尔逻辑成为集成电路设计的一个核心理论, 正如赫尔曼·戈德斯坦(Herman Goldstine)所说: “正是通过它[布尔逻辑], 使得电路设计从一门艺术变成一门科学”。同时, 也正是由于布尔逻辑的思想融汇在开关电路的设计中, 才会在集成电路领域形成著名的摩尔定律, 才使得集成电路和技术的创新发展得以实现。一阶逻辑、逻辑类型论和

收稿日期: 2016 - 11 - 03

基金项目: 国家社会科学基金特别委托项目“大数据时代的哲学理论与社会发展”(14@ZH023)。

作者简介: 陈鹏(1979—), 江西省南丰县人, 工学博士, 北京语言大学讲师, 中国社会科学院哲学所博士后, 主要研究方向: 计算机软件理论、逻辑、信息哲学。

Lambda 演算与编程语言的深度交叉,形成了程序设计理论的核心。形式语法(formal syntax)、类型系统(type system)和形式语义(formal semantics)成为一门程序设计语言的基础。逻辑的证明论、模型论思想与计算机软硬件系统的互动,构成了计算机系统正确性验证理论。人类基于霍尔逻辑、分离逻辑、Isabelle、Coq 等理论与工具,可以验证大型软硬件系统的正确性。

本文无意反驳上述各种观点,笔者也承认计算与逻辑有着密不可分的联系,尽管“逻辑”与“计算”本身的概念都不算特别清晰。本文试图从计算和逻辑二元关系中的另一个视角出发,不是研究逻辑之于计算如何,而是阐述现代“计算”的发现,对于我们认识“逻辑”有什么启示?也就是说,本文希望站在“计算”发现的起点,重新考古逻辑学的发展历史,并希望为逻辑学的发展方向做一些启示与思考。

## 一、莱布尼茨的逻辑学纲领

到了17世纪,逻辑学发生了变化,莱布尼茨提出了逻辑学应该做些什么。莱布尼茨旨在为科学建立一种普遍语言,这种语言对科学是理想的、合适的,以使用语句形式反映实体的性质。莱布尼茨认为,所有科学的思想,能化归为较少的、简单的、不可分解的思想,利用它们能定义所有其他思想,通过分解和组合思想,新的发现将成为可能,如同数学中的演算过程<sup>(3)30-31</sup>。

莱布尼茨并没有专门而系统地论述过逻辑学,与逻辑学相关的阐述主要在他的《论普遍文字》(On the General Characteristic)和《理性演算的基础》(Fundamenta calculi ratiocinatoris)等论文中。

莱布尼茨首先发现符号的普遍意义<sup>(4)181</sup>:

“人类的推理总是通过符号(signs)或者文字(characters)的方式来进行的。实际上,事物自身,或者事物的想法总是由思想清晰地辨识是不可能的,也是不合理的。因此,出于经济性考虑,需要使用符号。因为,每次展示时,一个几何学家在提及一个二次曲线,他将被迫回忆它们的定义以及构成这些定义的项的定义,这并不利于新的发现。如果一个算术家在计算过程中,不断地需要思考他所写的所有的记号和密码的值,他将难以完成大型计

算,同样地,一个法官,在回顾法律的行为、异常和利益的时候,不能够总是彻底地对所有这些事情都做一个完全的回顾,这将是巨大,也并不是必要的。因此,我们给合同、几何形状赋予了符号,在算术中,对数字赋予了符号,在代数中,对量纲赋予了符号,从而使得所有的符号都被发现为事物,或者通过经验,或者通过推理,最终能够与这些事物的符号完全融合在一起,在这里提及的符号,包括单词、字母、化学符号、天文学符号、汉字和象形文字,也包括乐符、速记符、算术和代数符号以及所有我们在思考过程中会用到的其它符号。这里,所谓的“文字”即是书写的,可追踪的或者雕刻的文字(Written, traced, or sculpted signs are called ‘characters’.)。此外,一个符号越能表达它所指称的概念,就越有用,不但能够用于表征,也可用于推理。”

基于此,莱布尼茨洞察到:我们可以为一切对象指派其文字数字(everything can be assigned its characteristic number),这样便能够构造一种语言或者文字,它能够服务于发现和判定的艺术,犹如算术之于数,代数之于量的作用。我们必然会创造出一种人类思想的字母,通过对字母表中的字母的对比和由字母组成的词的分析,我们可以发现和判定万物<sup>(3)30-31</sup>。

在莱布尼茨的洞察中,蕴含着两个非常重要的概念,即“普遍文字”和“理性演算”。

所谓的“普遍文字”,不是化学或者天文学的符号,也不是汉字或者古埃及的象形文字,更不是我们的日常语言,我们的日常语言虽然能够用于推理,然而它过于模棱两可,不能用于演算,也就是说,日常语言不能通过语言中的词的形成和构造来探测推理中的错误。相比较而言,与“普遍文字”最为类似的是算术和代数符号,在算术和代数符号中,推理都存在于文字的应用中,思想的谬误等同于计算的错误。普遍文字是一种人类思想的字母,通过由它组成的联系和词的分析,可以发现和判断一切。

另一个关键概念是“理性演算”,其中演算不同于推理,它是一个计算或者操作,即根据某种预先设定规则,通过公式变换产生的关系,而公式是由一个或者多个文字(characters)组成。与演算概念密切相关的是“文字艺术”(arscharacteristica):“构成与排列文字的艺术,通过这样一种方式,它们表征思想,也就是说,使得文字之间具有了思维之间才

具有的关系”。

为了实现这种“普遍文字”与“理性演算”,莱布尼茨试图建立对所有观念都适用的文字数字。然而,由于所有事物的奇妙的相互联系,使得明确地用公式阐述各个事物的文字数字极为困难。为此,莱布尼茨曾创制了一种精美的技巧,通过它,确凿的诸关系可以用数字加以表述和固定,然后又能在数字计算中进一步加以确定。

莱布尼茨进一步假定,即有些特殊的文字数字早已是已知的,其中有些特有的一般性质是可以观察到的。因而,抓住那些和特有的性质相关联的数字,然后借助它们用惊人的灵巧直接在数字上证明全部逻辑的规则,并且能查明一个已知的论证是否在形式上确定地提供一个标准。不管怎样,一旦我们占有了事物本身的真正的文字数字,一个证明是否实质上是确定的就可以第一次得到判断,而没有任何困难,也没有陷入谬误的危险。

所有的人类思想都可以消解为几个原始的概念,如果将文字分配给这些原始概念,那么可以通过文字的构成来形成派生的概念,通过它,我们可以抽取出它们的条件以及包含在其中的原始概念,更确切地说,它们的定义和值,甚至从定义中推导出来的属性。

应该说,莱布尼茨关于普遍文字和理性演算的想法是非常“理想化的、乌托邦式的”,莱布尼茨自己仅仅只是提出这么个设想,虽然他有过一些尝试,然而并没有完全实现,因此,不少的逻辑学家、哲学家将莱布尼茨的这种设想称之为“莱布尼茨之梦”,因为,在此之后,希尔伯特、哥德尔的工作表明,不存在如此完美的语言与演算,基于此,不少学者断言莱布尼茨之梦已经破碎<sup>(5)</sup>。

然而,在我们看来,实际上,被大多数学者所忽略的是,莱布尼茨其实提出的是逻辑学的研究纲领,是将逻辑学看成是“大科学的真正工具论”(True Organon of the General Science),逻辑学应该研究所有科学的思想,将科学的思想化归为较少的、简单的、不可分解的初始思想,利用普遍文字(characteristica universalis)定义所有其他思想(概念),通过文字艺术(ars characteristica)分解和组合思想,新的发现将成为可能,如同数学中的演算过程。一旦这种逻辑学的研究得以实现,那么就会如同莱布尼茨所言<sup>(3)</sup>:

“一旦人们对最大部分的概念建立起文字数

字,那时人类将会拥有一种新工具,它提高智慧的能力,远胜过光学工具之加强人眼,而理智之优越于视力,将和显微镜及望远镜之取代视力一样。它的用途之大,犹如给水手以指南针,它比星座带给所有在海上从事调查和实验的人的用途大得更多,从其中会进而产生出什么,这取决于命运之手,但它肯定会带来重要的和卓越的结果。”

## 二、作为代数的逻辑与 作为语言的逻辑

1967年,让·范·海耶诺特(Jean van Heijenoort)通过综述从弗雷格到哥德尔的逻辑文献<sup>(6)</sup>,提出了将逻辑学的发展划分为作为代数的逻辑和作为语言的逻辑两条路径<sup>(7)</sup>。其中,弗雷格引领的逻辑是作为语言的逻辑,而布尔发展的逻辑是作为代数的逻辑。

在范·海耶诺特看来,弗雷格的逻辑区别于布尔逻辑的一个关键性概念是逻辑的普遍性(the universality of logic)。这个普遍性首先是量词理论的全称性,这是命题演算所缺失的,正是这种全称性使得弗雷格逻辑能够实现对整个科学知识的符号重写。同时,在弗雷格的系统中,逻辑的普遍性是可以自我表达的,世界的本体可以分为对象和函项。弗雷格的世界是一个世界,就是这个世界,它是固定的,在这一点上,布尔与德摩根的世界是可以随意变化的。

1997年,辛迪卡(Hintikka)的《皮尔士在逻辑理论史上的地位》(The Place of C. S. Peirce in the History of Logical Theory)一文呼应了范·海耶诺特曾经提出的“作为语言的逻辑与作为演算的逻辑”区分<sup>(8)</sup>,并进一步提出:弗雷格同早中期罗素、早期维特根斯坦、维也纳学派、奎因、海德格尔等同属于一个传统,在他们眼中,只有大而全的一个世界即现实世界,现实世界是我们语言的唯一解释,不存在多个可能的世界,因而从根本上否定模态逻辑的合法性,否认真理的可判定性;而皮尔士同历史上的布尔、施罗德、勒文海姆、塔斯基、哥德尔、后期卡尔纳普等则属于另一个传统,他们支持包括可能世界理论(模态逻辑)、模型论、逻辑语义学和元逻辑理论等在内的一系列理论。

无论是逻辑的代数传统,还是逻辑的语言传

统,他们都将自身的逻辑起源归于莱布尼茨,都各自认为他们才是继承了莱布尼茨的正统。非常有趣的是,这两大传统在对莱布尼茨的继承中,都更多地关注了莱布尼茨的“普遍文字”的概念,而相对地忽略或者说轻视了莱布尼茨的“理性演算”的概念,两种传统都相互指出对方只是实现“理性演算”,而自身才真正实现“普遍文字”。事实上,逻辑的代数传统和逻辑的语言传统都未能实现莱布尼茨当初所描绘的“大科学的真正工具论”的逻辑学纲领。或许,罗素和怀特海在他们浩瀚的《数学原理》巨著中做过非常接近的尝试,然而最终被哥德尔的不完全性定理无情地击碎。似乎,莱布尼茨的逻辑学纲领已经走入穷途末路,正如一些逻辑学家所言,莱布尼茨之梦已逝。

### 三、图灵的计算转向

在逻辑的研究中,似乎忽略了图灵的工作(尽管在计算机科学研究中,图灵被追认为计算科学之父)。图灵通过细致的观察基本的数学运算,给出一种计算自动机的一般性定义,大略如下:

“自动机是一个‘黑箱’,具体细节没有给出,但它具有以下属性。它拥有有限数量的状态,这些状态通过这种方式便可表述:表明状态的总数,比方说 $n$ ,然后依次枚举它们 $1, 2, \dots, n$ 。自动机的核心操作特性包括描述如何使其发生状态的改变,即从一个状态 $i$ 进入另一个状态 $j$ 。这种状态改变需要与外部世界发生某种交互。对于机器而言,令整个外部世界由一条长纸带构成。令这条纸带宽,比如说1英寸,并令它分成一个个长1英寸的方格。在纸带的每个方格上,可以做或不做一个记号,比方说一个点,并假定这样一个点能够被擦除或者记上。标记有点的方格被称为1,无标记的方格被称为0。(我们可以使用更多的记号,但图灵表明,这无关紧要,不会给讨论的普遍性带来什么实质性帮助。)在描述纸带相对于自动机的位置时,假定纸带总有一个方格可被自动机直接阅读,并且自动机能够向前或向后移动纸带,比方说一次移动一个方格。具体而言,令自动机处于状态 $i$  ( $i = 1, 2, \dots, n$ ),并在纸带上读到一个 $e$  ( $e = 0, 1$ )。然后,它会转为状态 $j$  ( $j = 1, 2, \dots, n$ ),移动纸带 $p$ 个方格( $p = 0, +1, -1$ ;  $+1$ 表示向前移动一格, $-1$ 表示向后移动一

格),并对所读到的新方格进行标记( $f = 0, 1$ ; 0表示擦除,1表示在此处标记一个点)。将 $j, p, f$ 表示为 $i, e$ 的函数,这就描述了这样一个自动机功能的完整定义。”

上述就是图灵对计算机器的描述,后人称之为“图灵机”。基于图灵机,图灵给出了一个更加直观的“能行可计算的函数”定义:“如果能够通过某台机器的执行来获得某个函数的值,那么称该函数是能行可计算的”。根据丘奇-图灵论题,图灵的可计算定义与哥德尔的原始递归定义和丘奇的Lambda可定义是等价的。

这样,图灵机便成为一种可计算的表征,成为所有计算机的理论模型。尽管已经备受关注,然很少人注意到图灵机除了是一种可计算的表征之外,在图灵本人看来,它其实是一种符号逻辑。图灵曾经在1947年举办的伦敦数学学会的一次演讲中,阐述了他对符号逻辑和数学哲学的一些观点<sup>[9]</sup>:

“我期望数字计算机将最终能够激发起我们对符号逻辑和数学哲学的相当大的兴趣。人类与这些机器之间的交流语言,即:指令表语言,形成了一种符号逻辑。机器以相当精确的方式来解释我们所告诉它们的一切,毫无保留,也毫无幽默感可言。人类必须准确无误地向这些机器传达他们的意思,否则就会出现麻烦。事实上,人类可以与这些机器以任何精确的语言进行交流,即:本质上,我们能够以任何符号逻辑与机器进行交流,只要机器装配上能够解释这种符号逻辑的指令表。这也就意味着逻辑系统比以往具有更广阔的使用范围。至于数学哲学,由于机器自身将做越来越多的数学,人类的兴趣重心将不断地向哲学问题转移。”

图灵机有两个主要的组成:“自动机”和“指令表语言”。其中指令表语言是指描述状态转换表的语言,即描述自动机状态转换、读写以及移动的语言。图灵认为指令表语言是人类与机器之间的交流语言,形成了一种符号逻辑。这里的指令表语言就是我们后来所发展的各种类型的编程语言。

应该说,图灵机的“自动机”和“指令表语言”是对莱布尼茨的“普遍文字”和“理性演算”的诠释。图灵是比弗雷格、布尔和罗素更为成功地实践莱布尼茨梦想。在图灵的方案中:

“编程语言”是“普遍文字”的一种实现(编程语言 $\approx$ 普遍文字);

“自动机”是“理性演算”的一种实现(自动机

≈理性演算)。

在编程语言中,凡是与图灵机等价,或者说,能够计算任意可计算函数的编程语言称之为图灵完全的,目前大多数编程语言都是图灵完全的。同义反复,对于任何(图灵完全的)编程语言而言,它能够解决任何可计算的问题。图灵完全的编程语言在表达能力方面比弗雷格的谓词逻辑更强,能够描述包括物理、算术、化学、天文和生物等各种学科的问题,实现整个科学的符号重写。

图灵的逻辑与弗雷格和布尔的逻辑的本质区别在于,除了用编程语言实现“普遍文字”之外,图灵逻辑中还有一个负责“理性演算”的自动化演算机制,即“自动机”。也就是说,无论多么复杂的程序,都能够在“自动机”的自动演算机制下实现。

图灵将莱布尼茨的“普遍文字”与“理性演算”有效地融合,为逻辑带来一种“计算转向”,可以说,在作为代数的逻辑和作为语言的逻辑之外,图灵为逻辑开辟了“作为计算的逻辑”的新路径。这种对逻辑的审视,实质上是一种“主体转向”,“以往的逻辑”是当仁不让地以人类为主体,研究的对象是人的思维以及表征人类思维的自然语言种种,“作为计算的逻辑”则是将计算机作为信息处理的主体,研究的是计算机的处理方式以及人与计算机的互动关系。<sup>①</sup>

应该说,每一次对逻辑的重新审视,都会促进新的哲学反思。在哲学历史中,由于弗雷格开启了逻辑的语言传统,在哲学中引发了所谓的“语言转向”,大家都将弗雷格视为语言哲学的开创人,正是由于“语言转向”使得语言哲学家们言必谈弗雷格。近些年,国内外的哲学界,兴起了一股哲学的“计算转向”或者“信息转向”的思考,这些反思都源自于对图灵机及相关计算技术的审视<sup>(10)</sup>。笔者认为,站在逻辑学的视角,应该严肃地审视图灵的“作为计算的逻辑”,将逻辑学的发展适时地推进到计算的路径中。

## 四、结 语

自莱布尼茨撰写《论普遍文字》(On the General

Characteristic)至今,已经300多年。在这300多年之中,现代数理逻辑沿着布尔和弗雷格的代数与语言两大传统发展,然而,不难发现,无论是代数传统,还是语言传统,与当初莱布尼茨所构想的逻辑学纲领都相去甚远。

我们对图灵机在计算、演算方面的研究已经非常多,然而,似乎大多数人都忽略了图灵机的符号逻辑特性,按照图灵本人在1947年的讲话中所提到的,实际上,图灵机中的指令表语言形成了一种符号逻辑,我们可以将其称为图灵逻辑。图灵逻辑绕开了弗雷格逻辑的框架,也绕开了布尔—皮尔士—施罗德的逻辑传统,以莱布尼茨为起点,又重新建立起了一种新的“作为计算的逻辑”传统。

应该说,莱布尼茨从事逻辑学研究的一个重大动机是研究人类思想的符号系统,从而导致他致力于思考命题的本质、真理的概念等诸如此类的逻辑问题,也导致他试图以一种百科全书的形式构思所有知识<sup>(11)20</sup>。时至今日,纵然计算技术得到快速发展,即使突破了数据、存储与网络技术的限制,构建了类似于Cyc<sup>②</sup>、DBpedia<sup>③</sup>等这样的超大型知识库,然而这些知识库中描述实体、关系、属性等所用的逻辑仍然沿着布尔或者弗雷格的老路走,导致尚未能够真正,甚至部分实现莱布尼茨所构想的体系。回归莱布尼茨,沿着图灵的“计算传统”,重新审视实体、本体、意义等形而上学问题,或许我们能够为重温“莱布尼茨之梦”铺垫一条新的通路。

## 参考文献

- (1) Manna Z, Waldinger R. The Logical Basis for Computer Programming[J]. 1984.
- (2) 王元元. 计算机科学中的现代逻辑学[M]. 北京: 科学出版社, 1989.
- (3) Leibniz, Zur allgemeinen Charakteristik. Hauptschriften zur Grundlegung der Philosophie. Philosophische Werke Band 1. page 30–31. Translated by Artur Buchenau. Reviewed and with introduction and notes published by Ernst Cassirer. Hamburg: Felix Meiner. 1966.
- (4) Leibniz. Fundamenta calculi ratiocinatoris, 1688; A VI 4 920; English transl. in Dascal 1987.
- (5) [美]马丁·戴维斯. 逻辑的引擎[M]. 张卜天,译. 长沙: 湖南

① 这种转向在现代逻辑的发展中有过先例,罗宾逊提出的消解原理实质上就是一个“主体转换”思考的案例。

② 在2012年7月发布的Cyc的开源版本OpenCyc中,包含了23.9万个概念以及209.3万个事实。

③ 在2014年发布的DBpedia知识库中,包含超过458万的物件,144.5万人、73.5万个地点、12.3万张唱片、8.7万部电影、1.9万种电玩游戏、24.1万个组织、25.1万种物种和6000个疾病。

- 科学技术出版社, 2005.
- (6) Van Heijenoort J. From Frege to Gödel: a source book in mathematical logic, 1879 – 1931 [M]. Harvard University Press, 1967.
- (7) Heijenoort J V. Logic as calculus and logic as language [J]. Synthese, 1967, 17(1): 324 – 330.
- (8) Hintikka J. The Place of C. S. Peirce in the History of Logical Theory [M]. Lingua Universalis vs. Calculus Ratiocinator. Springer Netherlands, 1997: 140 – 161.
- (9) Carpenter B E, Doran R W. A. M. Turing's ACE report of 1946 and other papers [M]. MIT Press, Tomash Publishers, 1986.
- (10) [美] L. 弗洛里迪, 刘钢. 什么是信息哲学? [J]. 世界哲学, 2002(4): 72 – 80.
- (11) [美] 汤姆森. 莱布尼茨——最伟大的思想家(精) [M]. 李素霞 杨富斌 译. 北京: 中华书局, 2014.

## Computational Perspective of Logic: From Leibniz to Turing's Logical Development

CHEN Peng<sup>1 2</sup>

- (1. School of Information Science, Beijing Language and Culture University, Beijing 100083;  
2. Institute of Philosophy, Chinese Academy of Social Sciences, Beijing 100732, China)

**Abstract:** Modern logic originated in Leibniz, Boole and Frege respectively developed the so-called logical algebraic traditions of logic and language tradition of logic, however, both algebraic traditions and language traditions are a far cry from Leibniz's Logic Program, that is "logic as the basis of science, mathematics and philosophy". In the theory of Turing machine, the core elaborated the two concepts of "automata" and "instruction table language", which fit well with Leibniz's idea of "rational calculus" and "universal language". We argued that the theory of Turing machine was the closest to Leibniz's Logic Program, which was out of the algebraic tradition and language tradition, and formed a new tradition, that is "computational perspective of logic" (also known as "computation tradition"). "computational perspective of logic" is essentially a "subject – shift", other views were human – beings – oriented, the object of study is human thinking, natural language, "computational perspective of logic" located computer as subject of information processing, object of the research is the computer's processing methods and the interaction between people and computers.

**Key words:** logic; Leibniz's Program; Turing machine; computational perspective of logic

(本文责任编辑: 董春雨)