计算复杂性理论

傅育熙 著

2023 年 9 月 11 日

《计算机专业教育丛书》基础理论系列编委会

编委的话

全国每年有不计其数的计算机专业教学研讨会，所讨论的内容大同小异。其中的一个名为“说书论教”的研讨会吸引了众多一线教师参与。会议组织者每年邀请四位教材作者进行为期两天的研讨。报告者均为某领域的学者，活跃于国际学术界，多年来一直讲授其研究领域内的一门核心课，并用心为该门课程撰写了教材。会议组织者希望这些作者和与会教师分享授课心得、介绍教材的组织、选题和撰写过程，给出如何围绕该教材开展课堂教学的详细建议。

今日的计算机专业教育被迫放弃一个想法，该想法说：在本科阶段让学生把该学的都学了。该学的太多了，对计算机科技工作者和从业人员而言，终身学习才是道理。在信息技术迅猛发展的今天，计算机专业负责人只负责为专业学生的终身学习计划的前四年提供引导，剩下的都是学生自己的事。基于这一认识，系主任应将计算机专业的基因培育和系统能力培养作为其首选考虑。那么，该如何为学生制定一个终身学习计划的前四年的课程体系？既然我们敢虚构一个研讨会，我们就不怕再虚构一个计算机系。校领导给这个系定名为“机算计系”，要求系主任为计算机专业的学生设计一个开放教育体系。图1是系主任的方案框架。主任认为，本科教育分为三阶段。第一阶段的任务是将专业基因灌注给学生，使学生具备基本的数理、计算思维、问题求解能力。第二阶段为学生提供各类可供选择的课程模块，如系统模块、人机交互模块、计算机应用技术 2.0 模块（即大数据-人工智能模块）、信息安全模块、物联网技术模块等。第三阶段要求每位学生设计一个系统或参与一个成品的研发，并允许学生在全校范围内选听任何和项目相关的课程。系主任认为，开放系统

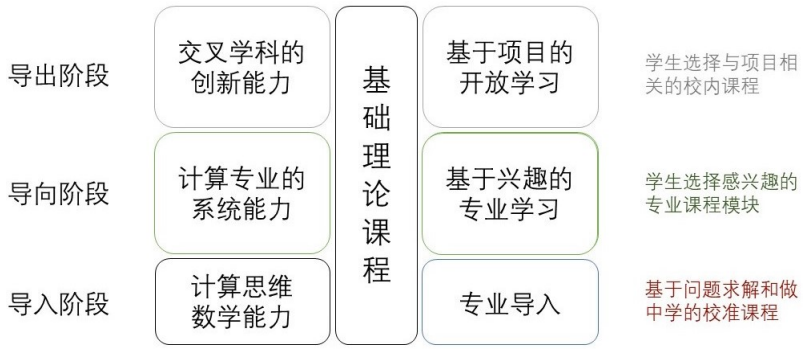

图 1　计算机专业教育的开放系统

可动态地建立和取消一个模块，也没有必要为每届学生安排同样的模块，所以能以最低的成本应对人才市场的需求变化。

学生首选的模块“计算机应用技术 2.0”越来越像社会科学那样大量地使用统计方法。当对事物的本质一无所知，当无法用数学时，我们只能借助于统计。但是，必须充分认识到，理论计算机科学在纵横两个方向有了极大的发展，一些计算机应用技术的重大突破源自深层次的理论结果。一个好的 211 大学的计算机专业应能在四年时间里为学生提供充足的理论课程。理论课程不一定要每年开设，可以两三年一循环，只要给每位在校生一次选听的机会即可。

无论是“说书论教”还是“机算计系”，都在呼唤好作者好教材。教育部高等学校计算机类教学指导委员会于 2019 年 5 月启动了“计算机专业教育丛书”计划，基础理论系列是该计划启动的首个项目。十余位志同道合的学者和出版社同仁走到一起，组成了编委会。他们中的好几位教授，已与出版社签订了出书合同。中国的计算机教育界期待着这套系列中的每一本早日与师生见面。正如一位编委所说的，《计算机专业教育丛书》不止是一套丛书，它是一项事业。

基础理论系列编委会，二〇二二年十二月二十六日

鸣谢

国画大师汤胜天先生为本系列的每部著作赐画一幅，吾等感激备至。“科学与艺术是一个硬币的两面，谁也离不开谁”（李政道）。

基础理论系列编委会，二〇二二年十二月二十六日

前言

当学生时没学懂的地方，教这门课时学懂了。看书时忽略的细节，备课时注意到了。备课时自信能讲好一个定理的证明，上课时挂板了。上课时高调阐述一个得意的观点，回答学生提问时心虚了。十年后，当准备为这门课写自己的教材时，终于知道，关于这门课，知道的太少了。为写教材，花了一年多查阅文献，才发现，有个别地方，过去十年一直在误导学生。

解惑之道，是一条往返于细节与真理之间的路。这条路，教师多走几遍，学生多获益几许。要做到在讲堂上从心所欲，教师须经历读书、教书、写书全过程。写此教材，有几层考虑。其一，我非常想进一步提高计算复杂性理论这门课的课堂教学水平。2020 年秋季学期，当第十几次给高年级本科生和研究生讲授这门课时，我已没了以往的激情，不是个好兆头。为求改变，我边讲课，边把课上所讲内容写下来。2021 年春季学期，我讲授“计算复杂性理论高等议题”这门课。尽管我对这部分内容远不如我对秋季学期讲的那部分内容熟悉，我还是认真做了讲课笔记。之后的一年，我对这些笔记改头换面，补充了所有课上没讲的证明，增加了不少这门课以前没有涉及的内容，在结构上做了变动，直至最后，对这本书的定位也做了些许调整。在本书完稿之前，我已确信我能把这门课讲授得更好。考虑之二，中国的计算机专业教育急需一本系统介绍计算复杂性理论的中文教材。中国的计算机专业教育是全球最大的专业教育，如果它的学生还要为教材发愁，如果它的学生使用的大多数教材都是国外学者撰写的，我们有什么理由说我们的计算机专业教育是好的。出版社的同仁告诉我，百分之九十五以上的中国学生更愿意使用中文教材。我认为他们有绝对的权力要求好的中文教材，任何一本能解决中国计算机专业教育燃眉之急的教材都值得马上写。考虑之三，教育部高等学校计算机类专业教学指导委员会于 2019 年 5 月启动了“计算机专业教育丛书”系列，作为该丛书的编委会主任，我有幸与国内计算机专业领域的一些著名学者探讨过撰写中文专业教材的问题，他们中的一些接受了我的邀请，加入了撰写教材的队伍。受他们的鼓舞，我也决定写一本教材。我觉得，如果我不这样做的话，一定会被认为

有点虚伪。曾经犹豫，因为国内有一批教授能写一本比这本教材更好的计算复杂性理论教材；不再犹豫，因为有一本将被超越的教材总比没有好。当然，还有一些其它考虑。在高校教书近三十年，能有一本自己的教材，既多了一份对我们选择的终身职业的敬意，也多了一个和同行交流的话题。

本教材的主要读者群是高年级本科生、硕士生、博士生，以及希望了解（更多）计算复杂性理论的教师和科技工作者。我希望这本教材包含足够多的细节，学生在积极参与了课堂学习之后能通过阅读本书的相关章节完全理解有关的定义、定理、证明。我希望这本教材自圆其说，读者无需参考任何其它资料就可理解本书的全部内容。这些努力是否成功得由读者裁定。

本教材可作为以下课程的主参考书：

1. 面向高年级本科生、研究生的“计算复杂性理论导论”，内容涵盖前三章，可略去第1.6节（但要讲线性加速定理）、第1.24节、第2.8节、第3.6节。若对前三章内容做了较大删减，第4章的前几节内容也可被涵盖。

2. 面向研究生的“计算复杂性理论高等议题”，内容涵盖第4章（“导论”课讲过的除外）、第5章、第6章。

3. 面向已经学过算法课程的高年级本科生、研究生的“算法理论”课程，内容涵盖第4章、第6章中有关随机算法和去随机、近似算法和不可近似性的内容。

4. 面向高年级本科生、研究生的“计算理论”，以第1章的内容为核心，并根据学分多少和授课对象不同做适当增减。

作者在上海交通大学讲授“计算复杂性理论导论”和“计算复杂性理论高等议题”多年，也在其它学校讲过“计算复杂性理论导论”。感谢选读这些课的学生，他们的问题总会驱使我去探究计算复杂性理论的更多细节。特别感谢所有的助教，他们为提高这两门课的教学质量做出了贡献。两门课的所有电子演示文稿可供读者随意使用，无需征得作者同意。感谢清华大学出版社龙启铭编辑，无论是作为作者还是作为主编，与出版界的合作一直是愉快的。

我希望本书能让读者收获人类智慧，我也期待着收获读者的批评和建议。

傅育熙，二零二二年二月八日，上海交通大学徐汇校区

目录

第 1 章　计算理论

物理系统从一个状态（初始状态）到另一个状态（终止状态）的转变过程就是计算，只要初始状态是人类可预置的，且终止状态是人类可观测的。

计算理论要回答三个问题。第一个问题是：什么是计算？什么问题可以借助机器求解？著名的丢番图方程要求找出整系数多项式方程 $a_1x_1^{n_1} + a_2x_2^{n_2} + \ldots + a_kx_k^{n_k} = 0$ 的整数解。希尔伯特第十问题问：是否存在一个计算过程，判定任给的一个丢番图方程是否有整数解。数学家对什么是计算这个问题颇感兴趣。从机械化的角度看，证明过程是一个寻找满足一定数学和逻辑性质的符号串，读者在理解这个证明时，要进行一个形式化验证过程。数学家的兴趣是，证明过程在多大程度上可以机械化。二十世纪上半叶在数学基础和计算基础领域的研究最终达成了共识：计算是一个独立于任何模型的概念，所有计算模型定义的计算都是等价的。建立在这一共识基础上的可计算理论回答了第一个问题。希尔伯特第十问题最终被年青的苏联数学家马蒂雅谢维奇于 1970 年解决，他证明了该问题的答案是否定的 [165]。若一个问题可以借助于机器求解，我们总会设法让机器代替人类解决该问题，与人类相比，机器的优势不言而喻。所以第二个问题是：如何让机器求解一个可计算问题？一台专用设备可以解决某一类特定问题，一台冯诺依曼体系架构的计算机可以通过预置一段程序来解决指定问题。无论是用专用计算设备还是通用计算设备解题，核心是算法。算法理论研究的，正是如何让机器解决问题。尽管人类对算法的兴趣历史悠久，但作为一门理论，系统性的研究和理论突破发生在计算机出现之后。一个著名的例子是素数分解。这是数论中一个古老问题，直到 2004 年，人们才发现这个问题有高效算法 [7]。另一个著名的问题是图同构问题。种种迹象表明，这不是个难问题，但一直没有找到它的高效算法。巴柏在 2016 年发表了图同构问题的一个准多项式时间算法 [24]，被发现了一个错误后，巴柏在几天之后公布了一个更新。这些例子，把我们带到了第三个问题：给定一个计算问题，解决该问题需要多少资源？资源包括时间、空间，但最根本的资源限制

Diophantine
Hilbert

Matiyasevich

efficient

何谓难？见第2章

Babai
quasi-polynomial
文章未见正式发表

是能量。围棋机器人可以完败人类选手，但在下棋过程中，前者所消耗的总能量远大于后者。信息技术的发展迫使我们思考如何制定更公平（因而也更环保）的游戏规则。以围棋为例，游戏规则应要求博弈双方在博弈过程中的能耗差限制在一个合理的范围。能耗限制是实实在在的。实际应用中，我们关心一个问题是否有可行算法，即它是否有一个多项式时间算法。如果一个可计算问题没有可行解，它就是一个理论上可计算但实际中不可计算的问题，我们得想其它办法。再看一个著名的问题: 给定一个图，该图的一个结点覆盖是一个结点子集，图中的任何一条边都和该结点子集中的某结点关联。最小结点覆盖问题要求计算出一个图的极小的结点覆盖集。实际中，这就是探头安装问题，我们希望用最少的探头，监控到一个楼面的所有走廊。遗憾的是，探头安装问题没有可行解。计算复杂性理论研究如何根据解决问题所消耗的资源量对问题进行分类。它试图刻画一个问题的绝对复杂性，即界定解决该问题所需的最小资源，尽管在这方面计算复杂性理论不太成功。它还希望比较不同问题对资源的相对消耗量，在这方面计算复杂性理论非常成功。计算复杂性理论的核心关注就是可行计算，为了可行性，可以在一定范围内牺牲正确性、精度、完全自动化，甚至可以同时牺牲三者 [240]。

feasible algorithm

可行计算的重要性很早就在埃德蒙兹和科伯姆的文章里被强调 [56, 75]。在他们之前，一位伟大的数理逻辑学家也注意到了可行计算的重要实际意义。1988 年 5 月 27 日，哥德尔在 1956 年 3 月 20 日写给病中的冯诺依曼的一封信重见天日。信中，哥德尔本人试图绕过他的不完备定理给数学带来的桎梏。他写道：“…容易构造一台图灵机，对每个一阶谓词公式 F 和每个自然数 n，判定是否存在一个长度是 n 的 F 的证明。设 $\psi(F,n)$ 是机器计算这个问题的步数，设 $\phi(n)=\max_F \psi(F,n)$。问题是，对一个最优化的机器而言，$\phi(n)$ 的增长速度有多快?” 如果 $\phi(n)$ 的增长速度是低的，那么 “只要取一个足够大的 n，当机器找不到一个证明时，继续想那个命题就没有任何意义”。的确，倘若世界上最优秀的数学家穷其一生都无法理解一个定理的叙述或该定理的证明，又有谁会在乎那个很长的符号串呢！做过严肃数学的人都不愿意相信 $\phi(n)$ 会是一个增长速度缓慢的函数。在读完本书的第二章，读者会确信，哥德尔定义的这个问题是一个 NP-完全问题。所以，那台图灵机不会对数学家有什么帮助，更不会对数学家这一职业构成威胁。哥德尔不仅是第一位对计算进行形式化研究的那个人，也是第一位提出 “**NP** 是否等于 **P**?” 的那个人。

Edmonds
Cobham

Gödel
von Neumann
原文及英译见 [194]

长度是指公式的长度加证明的长度

实际可计算
= 可行计算
= 高效可计算
= 多项式时间可计算

可行计算就是实际可计算这一思想贯穿于计算复杂性理论的研究。比如，当我们判断一个串是否是随机串时，我们的标准是看是否存在一个多项式时间算法将这个串和一个真正意义上的同等长度的随机串区分开来。如果没有任何多项式时间算法能做出有意义的区分，在实际中那个串就可被当成一个随机串。基于这一标准的伪随机理论 [236] 在计算机科学中有广泛和深刻的应用。非对称密码学基于同样的标准。如果必须消耗巨大的资源（时间、能量）才能破译一段密文（比如用蛮力算法），破译者不会花五十年的时间进行破译，届时明文所说的可能早已是公开的秘密。在区块链领域得到很好应用的零知识证明理论中，我们也是假定验证者无法在多项式时间内从交互中获取任何有用信息。另一方面，如果我们必须为某个无可行解的问题设计一个程序解决方案，我们只能放弃一些原则。有时我们不能要求程序在所有的输入上都给出正确的结果，有时我们不得不满足于次优解，有时我们允许程序在计算过程中停下来，让人类导航下一步计算。在一些应用领域，比如机器学习，我们会综合使用这些方法。“随机 + 交互 + 小概率错误”是计算技术这个行业流行的口号。如果读者是一位高校教师或学生，一定知道或想知道学校每学期的课程表是如何设计的。为上万名学生安排上课需考虑很多约束，教室、教师、同一门课两次授课之间的间隔、学生换教室过程中需步行多少时间、公共课、专业课以及各类权重。排课程表问题是一类著名的 NP-完全问题（约束可满足问题）的计算版本，因此没有一款软件能确保在开学前算出一个最优的课程表。目前大多数学校教务处的做法是，用某个（运行时间为多项式的）商业软件算出一个排课预案，然后进行人工调整。这个例子只是信息社会中众多案例中的一个，实用算法的能力及其局限性定义了我们的工作模式。当数据量很大时，实用算法不仅必须是多项式时间的，它们还必须是低次多项式时间的。一个著名的例子是快速傅里叶变换 [62]，该算法基于傅里叶变换的周期性和对称性，用二分法对离散傅里叶变换进行加速，将 $O(n^2)$ 时间算法改进成了 $O(n \log n)$ 时间算法。这个指数加速对实时数字信号处理技术而言是革命性的。信息时代的年青人已无法想象没有快速傅里叶变换算法的生活。如果计算机科学技术中有什么东西将定义人类社会一切的话，那无疑就是（低次）多项式时间算法。

文章 [110] 讨论了快速傅里叶变换的历史。

“软件定义一切”是一种没有文化的说法。

尽管时间是衡量一个问题是否具有可行解的主要指标，从能耗的角度看可行计算更具启发意义。当试图解决一个需要消耗高能量的问题时，我们允许算法通过交互输入一些正能量（如随机性、数据分布、对部分输入参数的限制、

启发式规则、人类的判断），也允许算法输出小量的负能量（如没有结果、错误结果）。如果想降低负能量的输出，只能倍增正能量的输入。能量的观点可帮助我们认清计算复杂性理论中的一些基本概念，如什么是问题的固有复杂性？从能量的角度看，问题的固有复杂性是对解决该问题所必须消耗的能量的度量。能量的观点可以简单地排除一些听上去有点恶作剧的“高效算法”[2]。其中一个利用相对论设计的算法是这样的：教授写下了一个 NP-完全问题的大的输入实例，让他的学生将该实例录入电脑并启动解决该问题的程序。之后教授乘上宇宙飞船，以接近光速遨游太空。等教授再次踏入实验室，他的学生早已作古，而教授则看到了电脑上显示的程序运行结果。问题是，如果教授真能以接近光速遨游太空，他的宇宙飞船必须携带起码是指数量级的燃料，他是不太可能在有生之年给宇宙飞船加入那么多燃料的。

技术的进步终将止步于物理极限，硬件速度的提高不可能让我们逾越多项式时间算法的制约。那么，关于可行计算，还有什么可多说的？计算力的提升让我们在很多领域进行范式转变成为可能。利用强大的计算设备，我们的低次多项式时间算法能够访问超大规模的数据库，处理超大规模的输入，这似乎又提供了无限多的可能性。通过与环境交互，多项式时间算法可以像数据科学家一样实时分析环境数据，可以不间断地进行有监督和无监督学习，可以通过模拟对手实现自我优化。未来会有越来越多被训练出来的不断演化的系统，我们甚至都不知道驱动这些系统运行的多项式时间算法是如何工作的。这一切才刚开始，它们对计算理论和计算复杂性理论提出的挑战也才开始 [249]。

paradigm shift

本书将只涉及经典复杂性理论那部分内容，重点讨论在各类模型中、各类场景下，“多项式时间可计算性”所对应的复杂性类，包括 $\mathbf{P}$、$\mathbf{NP}$、$\mathbf{PH}$、$\mathbf{P}_{/\mathbf{poly}}$、$\mathbf{RP}$、$\mathbf{BPP}$、$\sharp\mathbf{P}$、$\mathbf{IP}$、$\mathbf{PCP}(\log, 1)$。本书所追求的，不是介绍众多的计算复杂性类，而是试图就计算复杂性理论中的重要主题做较全面系统深入的讨论。

有空不妨去这里看看：ComplexityZoo

复杂性理论孕育了计算机科学中许多伟大的定理。要想理解这些定理，读者必须对它们的证明有透彻的理解。伟大的思想都在伟大的证明里。计算理论中的证明会用到递归论的、算术的、组合的、代数的、概率的、统计的、图论的、数论的、信息论的、博弈论的、证明论的、纠错码理论的方法，对这些方法的熟悉过程也是计算复杂性理论学习过程的一部分。

在进入主题之前，有一个决定，我们现在就得做。我们将基于哪个计算模型来展开计算复杂性理论的讨论？图灵机模型 [232, 233]。

文章 [232] 被认为是计算机科学中最重要的一篇文章。