计算科学丛书

Mathematical Theory of Computer Algebra System

计算机代数系统的数学原理

李超 阮威 张龙 张翔 编

清华大学出版社

计 算 科 学 丛 书

Mathematical Theory of Computer Algebra System

计算机代数系统的数学原理

李超 阮威 张龙 张翔 编

清华大学出版社
北京

内 容 简 介

本书主要介绍了计算机代数系统的数学理论、经典结果和著名算法. 全书包含高精度运算、数论、数学常数、精确线性代数、多项式、方程求解、符号极限、符号求和、符号积分、微分方程符号解等 10 个部分, 涵盖了构建计算机代数系统的最基础也是最重要的内容. 书中的许多内容是第一次被系统地整理后出现在中文文献中, 并在一些领域体现了本方向的最新进展.

图书在版编目(CIP)数据

计算机代数系统的数学原理/ 李超等编. --北京：清华大学出版社, 2010.10

(计算科学丛书)

ISBN 978-7-302-23010-6

Ⅰ. ①计… Ⅱ. ①李… Ⅲ. ①电子计算机-数值计算 Ⅳ. ①TP301.6

中国版本图书馆 CIP 数据核字 (2010) 第 107278 号

责任编辑：刘 颖 陈 明

责任校对：赵丽敏

责任印制：李红英

出版发行：清华大学出版社　　**地　　址**：北京清华大学学研大厦 A 座

http://www.tup.com.cn　　**邮　　编**：100084

社　总　机：010-62770175　　**邮　购**：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质　量　反　馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者：北京鑫海金澳胶印有限公司

经　　销：全国新华书店

开　　本：185×230　**印　张**：24.75　**字　数**：473 千字

版　　次：2010 年 10 月第 1 版　　**印　　次**：2010 年 10 月第1 次印刷

印　　数：1～4000

定　　价：39.00 元

产品编号：034908-01

序

对于计算机代数的重要性, 作者已经在前言中做了极好地论述, 我在这里不再重复. 这里我想谈谈我个人对这本书的看法: 这是一本很有特点的书, 是一部真正的年轻人的作品.

清华大学基础科学班的几个学生, 李超, 阮威, 张翔, 张龙, 决定利用清华大学SRT计划的支持, 写一个自己的"Mathematica"系统, 他们称之为maTHμ系统. 我也应他们的邀请, 成了他们的指导老师.

我邀请到了数学系的特聘教授Jacques Peyrière先生, 和我一起来听听他们的计划. 几个年轻人叙述了一个雄心勃勃的计划. 我和Jacques都对这个计划表示怀疑: 他们不仅不可能完成这个计划, 而且还会因此影响到他们的正常学习. 幸运的是我们还是给予了力所能及的支持.

开始是艰苦的收集资料阶段. 不仅仅是收集资料, 还要把其中的内容吃透. maTHμ 小组成员表现出了强大的实力, 尽管有时候他们也表现出幼稚的一面. 这也许是贯穿于整个研究计划的一个特点: 既有深刻的地方, 也有幼稚的地方. 但是前者是主要特点. 许多非常困难的理论都被一一理解, 并且写进这本书里面.

紧接着是系统实现阶段, 来自计算机系、电子系、自动化系等院系的许多同学也参加进来. 在几个月的时间内, 初具规模、性能不凡的maTHμ系统就完成了第一个版本! Jacques教授感慨地对我说: 他们完成了一个"不可能的工作".

虽然是指导老师. 但鉴于能力有限, 我几乎没有给他们任何指导. 可以说, 这是一部纯粹年轻人的作品, 弥漫着年轻人独特的开拓性和创造性. 从这本书中, 我们看到了清华学生的风采.

我的博士导师曾对我说过: 年轻人是用心在做数学. 我在日本的导师曾批评我说: 你的思想要是再野一点就更好了. 看来科学和文学类似, 需要心灵之作, 需要狂野之作.

饶 辉

2009 年 9 月 16 日

前　言

计算机代数(Computer Algebra)在很多时候又被广义地理解为“符号计算”(Symbolic Computation), 成为与“数值计算”(Numerical Computation)相对的概念. “符号”的运算在这里代替了“数”的运算. 这是一种智能化的计算. 符号可以代表整数、有理数、实数和复数, 也可以代表多项式、函数, 还可以代表数学结构, 如集合、群、环、代数等. 我们在学习和研究中用笔和纸进行的数学运算多为符号运算.

利用计算机代数, 我们可以完成许多不可思议的事情, 例如可以对代数方程组进行精确的求解, 对多项式进行因子分解, 对复杂代数表达式进行化简归约, 对函数进行符号积分(求出原函数), 对微分方程求出精确解等.

传统的代数计算冗长繁杂, 而现代的计算机技术为大型的符号计算提供了可能性. 关键的问题就在于如何把抽象的代数理论算法化, 使其高效地处理形形色色的代数问题. 强大的计算机代数系统不仅是各类工程技术的助手, 对纯粹科学研究也起着不可忽略的推动作用.

经过数十年的发展, 在国外已经形成了诸如 Wolfram Research[1], Maplesoft[2] 等巨型的商业软件公司, 其产品具有可观的经济效益; 其他一些研究者的专用系统开发也具有了相当的规模. 然而, 在我们国内, 科学软件领域则远远落后于发达国家, 能够与国外产品相抗衡的通用计算机代数系统还暂时没有出现. 而另一方面, 国内对科学软件的需求量却是巨大的, 昂贵的进口产品意味着大量的科研、工程经费的无奈外流. 从某种意义上来说, 对国外系统的依赖对国家信息安全也有着潜在的威胁.

在我们看来, 造成这种状况的原因一方面是由于科学软件的复杂性, 另一方面则反映了国内创新能力的缺乏. 就这一点来说, 国内的大环境是一个很重要的因素:

- 盗版冲击. 在国内盗版软件获取容易, 而大型的科学软件开发却需要长期、大量的投入, 其复杂程度及难以预料的经济前景使得企业少有问津;
- 知识结构缺欠. 在中国往往出现这种情况: 软件写得顺手的人, 未必有很强的科学计算背景, 而科学背景强的人又没有较高的计算机和软件水平, 这使得科学软件的开发难以进行;

[1] http://www.wolfram.com
[2] http://www.maplesoft.com

- 少有“不切实际的幻想”. 由于科学软件的复杂程度, 企业少有问津, 单独的科研人员也不得不认为完成这样一件工作是“不切实际的幻想”. 然而我们回顾 Wolfram Research, Maplesoft, Mathworks 等大公司的发展道路, 无不是从一个人或几个人的的微薄之力逐渐发展到现在令人赞叹的规模. 或许“不切实际的幻想”正是创新氛围的一个绝佳体现吧!

尽管我们力量渺小, 但对这种状况也不愿意置若罔闻. 清华大学作为一所综合性大学, 众多学子具有较强综合能力, 也理应在此领域有所作为. 我们利用自身的数学基础与应用背景, 整理出一份较为完整的计算机代数理论文档, 进行较为完整的系统设计, 并实现一个计算机代数系统核心 maTHμ[1]. 希望在此过程中也能开阔我们的眼界, 提高我们自身的理论和技术水平, 为更进一步的工作打下良好基础.

这本《计算机代数系统的数学原理》, 作为 maTHμ 系统的理论文档, 是 maTHμ 项目组成员通过近一年半时间集体整理撰写的成果. 计算机代数理论方面的中文文献稀缺. 即使在英文文献中, 能够足以支撑一个通用计算机代数系统的系统论述也较为少见, 更多的内容散见在专著、博士论文及专业期刊中. 花大力气整理文档的初衷十分简单: 理论不清, 则后续的设计与开发阶段根本无法进行, 更何况计算机代数理论本身就复杂而相互交织. 举例来说, 求微分方程的符号解需要符号积分的支持, 而有理函数符号积分需要借助精确线性代数、多项式因子分解、代数方程组求解等算法, 其中 $\mathbb{Z}[x]$ 上的多项式因子分解依赖有限域上的因子分解, 从而需要数论中模算法的支持, 而数论算法又建立在任意精度的快速运算上.

本书包含高精度运算, 数论, 数学常数, 精确线性代数, 多项式, 方程求解, 符号极限, 符号求和, 符号积分, 微分方程符号解等十个大部分, 涵盖了构建计算机代数系统的最基础也是最重要的内容. 整个书稿的内容组织也大体符合构建计算机代数系统的逻辑顺序. 许多内容是第一次被系统地整理出现在中文文献中, 在一些领域也追踪到了最新进展. 其中第 1, 5, 14, 15 章由张翔撰写, 第 2~4, 16, 17 章由李超撰写, 第 6 章由张龙撰写, 第 7~13 章由阮威撰写. 李超负责统筹全书并撰写了附录.

本书中大部分算法都有理论推导, 努力做到了自成体系并力求阐明各种方法背后的想法. 对于若干深刻的结果(如 Hilbert 零点定理, Liouville 定理, Lie-Kolchin 定理等), 鉴于篇幅和目的, 我们选择只给出相关参考文献而略去了严格证明.

本书中的部分算法已经在 maTHμ 1.0 版的内核中实现. 计算机代数系统 maTHμ 项目在 2009 年清华大学第二十七届“挑战杯”课外学术科技竞赛中, 经过 4 轮评审, 最终在 344 件作品中脱颖而出, 获得了特等奖的第一名. 2009年11月, 项目代表清华大学参加了两年一度的全

[1] http://www.mathmu.cn

国“挑战杯”竞赛并获特等奖. 项目未来的长期发展规划也已经得到了学校的大力支持. 项目团队感受到了来自各方的鼓励与期望, 希望能踏实地继续努力工作. 在理论文档部分, 除了继续丰富与完善呈现在这里的内容, 其他一些我们认为同样重要的主题也正在整理中, 包括初等与特殊函数的任意精度计算、组合函数、代数函数积分、更一般的符号求和理论、表达式化简等.

两年过去, maTHμ 从无到有, 一路走来. 项目团队最为欣慰的事莫过于看着我们的“幻想”正一点点变为现实. 作者感谢 maTHμ 项目指导老师饶辉和法国教授 Jacques Peyrière 对项目的精心指导. 感谢清华大学数学科学系李建国, 白峰杉, 李津, 扈志明等老师, 校团委阳波老师, 校科创中心及校科协对项目提供的支持. 感谢中科院数学机械化实验室李子明老师的有益讨论. 清华大学数学科学系卢旭光老师和清华大学出版社刘颖老师在本书出版过程中付出了辛勤劳动, 在此一并表示感谢.

maTHμ Project Group
maTHmU@gmail. com
2009年11月17日于清华园

目　　录

1 高精度运算

我们所熟知的科学计算一般就是指数值计算. 数值计算是计算数学的一个主要部分, 它研究用计算机求解各种数学问题的数值计算方法及其理论与软件实现. 关于数值计算的研究在发明计算机之前就已经有了相当的基础, 它涉及的内容包括函数的数值逼近, 数值微分与数值积分, 非线性方程数值解, 数值线性代数, 常微分方程与偏微分方程数值解等(参见 [7]). 数值计算中处理的对象并不仅仅是数值, 还包括由数值构成的简单数据结构, 例如一般的多项式、无穷级数、矩阵等, 数值计算处理问题的一般方法是通过数学推导将问题化归到这些数学对象的运算上.

数值计算的主要目标是解决来自于实践中的物理、工程、经济等领域的问题. 与此同时, 数学工作者做数学研究本身也是一种实践, 数学研究过程中同样会产生许多问题, 与工程学问题不同, 这些问题多是用抽象符号表达的, 因而仅用数值计算的方法是不易解决的, 对于这类问题解决方案的研究, 为了与数值计算相区别, 常常称之为符号计算. 类似地, 我们可以给符号计算下一个简单的定义:符号计算是一门研究用计算机求解各种数学问题的符号计算方法及其理论与软件实现的科学. 符号计算中处理的数据和结果都是符号, 这种符号可以是字母, 公式, 数也可以作为一种符号出现在符号计算中, 但这里关于数的运算应该是绝对精确的, 我们接下来就要讨论数的高精度运算.

1.1　整数

计算机的机器指令能够直接处理的整数是有界的, 在目前典型的计算机中整数的溢出界都不超过 2^{64}, 而符号计算中常常需要处理更大的整数, 例如阶乘, Fibonacci 数列这样简单的数论函数计算. 另一个不平凡的例子是所谓的中间表示膨胀(intermediate expression swell)(参见 [14] 第 2 章), 例如采用 Euclid 算法计算两个整系数多项式的最大公因子时, 即使输入的两个多项式和输出的最大公因子都具有绝对值较小的系数, 计算过程中的中间结果仍然很可能出现绝对值非常大的系数. 设

$$F = 7x^7 + 2x^6 - 3x^5 - 3x^3 + x + 5,$$
$$G = 9x^5 - 3x^4 - 4x^2 + 7x + 7,$$

在求它们的最大公因子的计算过程中需要将有理数化为整数, 我们将得到如下的多项式序列

$$1890x^4 - 4572x^3 - 6930x^2 - 846x + 4527,$$
$$294168996x^3 + 257191200x^2 - 20614662x - 142937946,$$
$$-10368527836984130520 0x^2 - 325760542331156100 00x$$
$$+ 122463167842311670000,$$
$$29567908335038495467893420575652070982917635200 00x$$
$$+ 55532526180624799696603478407402529168762016000 0,$$
$$10920746857330312192010416027912598626591699661845938035 18$$
$$60241877714068288433476964706040354360773769842688000000 0000,$$

最后一个整数达到了 118 位. 除此之外, 高精度浮点数的表示和运算也是直接依赖于高精度整数的.

1.1.1　进制转换

为了提高运算效率, 高精度整数的内部表示常常采用 2 的正整数次幂进制, 如 2^{32} 进制或 2^{64} 进制, 而人们书写或阅读时更习惯于采用十进制, 因此高精度整数输入输出时常常需要做进制转换. 即给定正整数 n 的 B 进制表示

$$n = (a_s \cdots a_1 a_0)_B,$$