数值计算原理

李庆扬　关　治　白峰杉　编著

清华大学出版社

(京)新登字 158 号

内　容　简　介

本书的内容是现代科学计算中常用的数值计算方法及其原理，包括数值逼近，插值与拟合，数值积分，线性与非线性方程组数值解法，矩阵特征值与特征向量计算，常微分方程初值问题、刚性问题与边值问题数值方法，以及并行算法概述等。本书是为学过少量《计算方法》的理工科研究生学习《数值分析》而编写的教材。内容较新，起点较高，叙述严谨，系统性强，偏重数值计算一般原理。每章附有习题及数值试验题，附录介绍了 Matlab 软件以便于读者使用。本书可作为理工科研究生《数值分析》课程的教材或参考书，也可供从事科学与工程计算的科技人员学习参考。

图书在版编目(CIP)数据
数值计算原理/李庆扬等编著. —北京：清华大学出版社，2000
ISBN 7-302-03942-9
Ⅰ. 数…　Ⅱ. 李…　Ⅲ. 数值计算-研究生-教材　Ⅳ. 0241
中国版本图书馆 CIP 数据核字(2000)第 32999 号

出版者：清华大学出版社(北京清华大学学研大厦，邮编 100084)
http://www.tup.tsinghua.edu.cn
印刷者：北京国马印刷厂
发行者：新华书店总店北京发行所
开　本：850×1168 1/32　印张：14.875　字数：371 千字
版　次：2000 年 9 月第 1 版　2001 年 6 月第 2 次印刷
书　号：ISBN 7-302-03942-9/O・242
印　数：5001～10000
定　价：20.00 元

前　　言

本书是为清华大学理工科各专业硕士研究生学位课程《数值分析》编写的教材。我们开设研究生《数值分析》课已经 20 年,先后编写并使用过的教材有多种[1~3,5]。近六七年针对本科生已普遍学过 32～48 学时的《计算方法》的情况,为使学生学到更多的数值计算新内容,我们使用《现代数值分析》[1]作为教材(1995—1997),1998 年又使用了《数值分析基础》[3]作为教材。根据多年教学实践及使用上述教材中发现的问题,并考虑到新世纪对理工科研究生数学教育的要求,我们在原有教材的基础上编写新的教材。考虑到本科生已学过少量《计算方法》,新教材尽可能不重复这部分内容,但其基本内容仍包括数值逼近与插值,数值积分,线性方程组和非线性方程组数值解法,矩阵特征值计算及常微分方程刚性问题与边值问题数值方法等。考虑到理工科研究生对科学计算的需要以及近年计算机的硬件及软件环境,并兼顾到提高数学素质的目的,本教材具有以下特点:(1) 内容更新,起点更高。删除了以前教材中较基本和陈旧的内容,增加科学计算中的实用方法和一些新的方法。(2) 着重数值计算基本原理和各种方法的基本思想的阐述。注意数学概念的准确性和严密性,使学生的数学逻辑思维能力得到训练,但又根据非数学专业的要求减少了定理的证明,内容上还减少了具体算法步骤的描述。(3) 由于并行计算机硬件条件的改善,掌握并行算法并将它付之实践成为可能,本教材介绍了并行算法的基本概念并将并行算法思想贯穿于有关章节,但对具体并行算法不多作描述,目的只是使读者需要时能加以使用,并非基本要求,但有了并行算法思想将对数值计算算法理解更为全面。(4) 加

强数值试验，掌握并应用数值计算方法是本课程的重要部分，它必须通过数值试验达到，为此本书各章都给出了适当的数值试验题，供读者选做，并推荐 Matlab 软件作为基本计算工具。在附录中对 Matlab 做了介绍，附录中还简单介绍了其他数学软件工具，目的都是为了加强数值试验。

根据上述内容和特点，本书定名为《数值计算原理》，它适合于作为大学理工科各专业研究生“数值分析”学位课的教材，它在深度和广度上均超过现有《数值分析》教材的内容，较适合于在本科已学过少量《计算方法》，数学要求较高的高校选用。由于本书自成系统，即使本科没有学过《计算方法》，只要有微积分和线性代数基础也不会有太大困难。本书内容较多，对于只有 64 学时的课程，任课老师可根据学生情况和教学大纲要求适当删减某些内容。此外本书也可供从事科学与工程计算的科技人员学习参考。

本书第 1，2，4，6 章由李庆扬编写，第 3，5 两章由关治编写，各章的数值试验题及附录由白峰杉编写。

清华大学数学科学系领导对本书编写非常重视，组织原计算数学教研组教师对课程大纲进行了认真讨论，并对本书取材提出了宝贵意见，陆金甫教授仔细审阅了全书并提出了修改意见，清华大学出版社在时间紧迫的情况下为本书出版给予大力支持，在此我们表示衷心感谢。我们希望使用本书的老师、同学及广大读者对本书提出批评指正。

编　者

2000 年 3 月于清华园

目　　录

第 1 章　数值计算原理与计算精确度

1　数值计算的一般原理

1-1　数学问题与数值计算

数学与科学技术一向有着密切的关系并互相影响, 科学技术各领域的问题通过建立数学模型与数学产生了紧密的联系, 数学又以各种形式应用于科学与工程. 近几十年由于计算机的飞速发展, 求解各种数学问题的数值计算方法也愈来愈多地应用于各领域, 除科学与工程计算外, 还包括医学, 经济管理和社会科学等.

实际应用中所导出的数学模型其完备形式往往不能方便地求出精确解, 于是人们就局限于讨论问题的简化模型, 例如将复杂的非线性模型忽略一些因素, 简化为线性模型, 但这样做往往不能满足精度要求, 因此目前更多的是用数值计算方法来直接求解较少简化的模型, 而计算量大小往往依赖所用的数值方法及所需精度. 高速大型和并行计算机的发展为利用数值计算方法进行科学与工程计算(简称科学计算)提供了条件, 为适应这种需要数值计算方法的研究发展也很快, 一批适合计算机求解并节省计算量的数值计算方法随之产生, 并被广泛使用, 成为科学计算的主要方法. 数值计算是指面向数学问题适合于计算机使用的数值方法, 是计算数学重要部分, 也是计算科学的公共基础和最通用的数值方法.

本书作为研究生教材, 考虑到学生已学过最基本的“计算方法”, 原则上不再详细介绍基本的和目前较少使用的算法, 而着重

介绍一些现代科学计算中较常用的算法和数值计算的一般原理. 本书仍然只涉及微积分,常微分方程和高等代数等基础数学的数值计算问题,但其内容深度和广度均超过目前理工科研究生“数值分析”课的要求.

1-2 数值问题与算法

数值问题是指输入数据(即问题中的自变量与原始数据)与输出数据(结果)之间函数关系的一个确定而无歧义的描述. 输入输出数据可用有限维向量表示. 根据这种定义,“数学问题”不一定是“数值问题”,但它往往可用“数值问题”逼近. 例如,解常微分方程 $\frac{dy}{dx}=x^2+y^2, y(0)=0$,它不是数值问题,因为输出不是数据而是连续函数 $y=y(x)$,但只要规定输出数据是 $y(x)$ 在 $x=h, 2h, \cdots, nh$ 处的近似值,这就是一个数值问题,它可用欧拉(Euler)折线法或其他数值方法求解,这些数值方法就是算法.

计算的基本单位称为算法元,它由算子、输入元和输出元组成,算子可以是简单操作,如算术运算+、-、×、/,逻辑运算,也可以是宏操作如向量运算、数组传输、函数求值等. 输入元和输出元分别可视为若干变量或向量. 由一个或多个算法元组成一个进程,它是算法元的有限序列. 一个数值问题的算法是指按规定顺序执行一个或多个完整的进程. 通过它们将输入元变换成一个输出元. 面向计算机的算法可分为串行算法与并行算法两类. 只有一个进程的算法适用于串行计算机,称为串行算法;两个以上进程的算法适合于并行计算机,称为并行算法. 对于一个给定的数值问题可以有许多不同的算法,它们都能给出近似答案,但所需计算量和得到精度可能相差很大. 一个面向计算机,计算复杂性好,又有可靠理论分析的算法就是一个好算法. 所谓计算复杂性包含时间复杂性

和空间复杂性两方面,在同一精度下,计算时间少的为时间复杂性好,而占用内存空间少的为空间复杂性好.

例 1.1 计算多项式

$$p(x) = a_0x^n + \cdots + a_{n-1}x + a_n$$

的值.

这是一个数值问题,输入数据为 $a_0, \cdots, a_n$ 及 x,输出数据为 $p(x)$,若直接由 x 算出 $x^2, \cdots, x^n$ 再乘相应的系数 $a_{n-1}, a_{n-2}, \cdots, a_0$ 并相加,则要做 $2n-1$ 次乘法和 n 次加法,占用 $2n+1$ 个存储单元.若将 $p(x)$ 改写为

$$p(x) = (\cdots(a_0x + a_1)x + \cdots + a_{n-1})x + a_n$$

用递推公式表示为

$$b_0 = a_0, \quad b_i = a_i + b_{i-1}x, \quad i = 1, 2, \cdots, n, \quad b_n = p_n(x)$$

它只用 n 次乘法和 n 次加法,并占用 $n+2$ 个存储单元,故这是一个好的串行算法,它称为秦九韶方法,也称霍纳(Horner)算法(秦九韶于 1247 年提出此算法,比霍纳于 1819 年提出的算法早 500 多年).

对于大型计算问题,不同算法计算复杂性差别就更大.例如解线性方程组,当 $n=20$ 时,用克莱姆(Cramer)法则,其运算次数(乘除法)需 9.7×10^{20},用每秒运算 1 亿次的计算机也要算 30 多万年.而用高斯(Gauss)消去法只需乘除运算 3060 次,并且 n 愈大相差就愈大.这个例子既表明算法研究的重要性,又说明只提高计算机速度而不改进和选用好的算法也是不行的.人类的计算能力是计算工具的性能与计算方法效率的总和,因此,计算能力的提高有赖于双方的提高.例如,1955 至 1975 年的 20 年间,计算机速度提高数千倍,而同一时间解决一定规模的椭圆型偏微分方程计算方法效率提高约 100 万倍,这说明研究和选择好的算法对提高计算速度,在某种意义上说比提高计算机速度更重要,因为算法研究所需代价要小得多.当然,选择好算法的前提是保证计算结果的可

靠性，这就要求有可靠的理论分析，使计算结果满足精度要求.

1-3　数值计算的共同思想和方法

在基本的"计算方法"中已经知道了一些常用的数值算法，它们可概括为以下几种共同的思想与方法.

1-3-1　迭代法

迭代法即逐次逼近法，它指按同一公式重复计算的一个数值过程. 例如在方程求根中，求方程

$$x = G(x) \tag{1.1}$$

的根，假定 $G(x)$ 在有根区间$[a, b]$上连续、可微，可构造迭代法

$$x_{k+1} = G(x_k), \quad k = 0, 1, \cdots \tag{1.2}$$

从某个初始近似 $x_0 \in [a, b]$ 出发，由(1.2)可算得 $x_1, x_2, \cdots$，计算一次 $G(x_k)$ 称为一次迭代. 若序列$\{x_k\}$收敛于一个极限 x^*，则由(1.2)有

$$x^* = \lim_{k\to\infty} x_{k+1} = \lim_{k\to\infty} G(x_k) = G(x^*)$$

故 x^* 是方程(1.1)的一个根. 同一方程可构造出不同的迭代法，它们有的收敛快有的收敛慢，有的不收敛，这与 $G(x)$ 在根 x^* 附近导数 $G'(x)$ 的变化有关. 若在根 x^* 附近 $|G'(x)| < 1$，则序列$\{x_k\}$收敛，且 $|G'(x^*)|$ 越小收敛越快，在精度要求内迭代次数愈少则收敛越快.

例 1.2　用下列三种迭代法求方程 $x^2 - 3 = 0$ 的根 $x^* = \sqrt{3}$，各计算 3 步，考察它们是否收敛和收敛快慢.

(1) $x_{k+1} = \dfrac{3}{x_k}$,　$k = 0, 1, \cdots$

(2) $x_{k+1} = x_k - \dfrac{1}{4}(x_k^2 - 3)$,　$k = 0, 1, \cdots$

(3) $x_{k+1} = \dfrac{1}{2}\left(x_k + \dfrac{3}{x_k}\right)$,　$k = 0, 1, \cdots$

解　取 $x_0 = 2$，分别计算 3 步，结果见表 1-1.

表 1-1　分类迭代表

x_k	方法(1)	方法(2)	方法(3)
x_0	2	2	2
x_1	1.5	1.75	1.75
x_2	2	1.73475	1.73214
x_3	1.5	1.73236	1.732051

此题精确解 $x^* = \sqrt{3} = 1.7320508\cdots$, 可见迭代法(1)不收敛,(2)、(3)收敛,而(3)收敛最快.实际上迭代法(1)的 $G(x) = \frac{3}{x}$, $G'(x) = -\frac{3}{x^2}$, $G'(x^*) = -1$; 在迭代法(2)中, $G(x) = x - \frac{1}{4}(x^2 - 3)$, $G'(x) = 1 - \frac{1}{2}x$, $G'(x^*) = 1 - \frac{1}{2}\sqrt{3} \approx 0.134 < 1$; 在迭代法(3)中, $G(x) = \frac{1}{2}\left(x + \frac{3}{x}\right)$, $G'(x) = \frac{1}{2}\left(1 - \frac{3}{x^2}\right)$, $G'(x^*) = 0$.

对线性方程组

$$Ax = b \tag{1.3}$$

其中 $A \in R^{n \times n}$, $b \in R^n$ 已知. 用迭代法求解 $x = (x_1, \cdots, x_n)^T$, 若 $A = M - N$, M 非奇异,也可构造迭代法

$$x^{(k+1)} = Bx^{(k)} + f, \quad k = 0, 1, \cdots \tag{1.4}$$

其中 $B = M^{-1}N \in R^{n \times n}$ 称为迭代矩阵, $f = M^{-1}b$. 若 B 的范数 $\|B\| < 1$, 则对 $\forall x^{(0)} \in R^n$, 可由(1.4)逐次求得 $x^{(1)}, x^{(2)}, \cdots$, 且 $\lim\limits_{k \to \infty} x^{(k)} = x^*$, x^* 即为方程(1.3)的解.

无论在实用上或理论上,处理线性或非线性问题,迭代法都是最重要的手段之一,但无论哪种问题都必须找到合适的方法把方程转化成类似于方程(1.1)的形式,并选取某个合适的初始近似.为了减少迭代次数,通常必须在多种方案中选取收敛较快的方法,因而同一问题可产生各种不同的迭代法.

1-3-2　以直代曲

另一个经常出现的思路是将非线性问题线性化, 也就是在一个局部范围中用直线近似代替曲线, 更进一步是用多项式逼近复杂函数. 仍以方程 $f(x)=0$ 的求根为例, 在几何上 $y=f(x)$ 是一曲线, 它与 x 轴交点的横坐标即为方程的根, 假如已给出一个近似根 x_k, 我们用曲线在点$(x_k, f(x_k))$上的切线逼近该曲线, 令 x_{k+1} 是该切线与 x 轴交点的横坐标, 在正常情况下 x_{k+1}对根的近似比 x_k 好(如图 1-1). 上述以直代曲相当于用 $f(x)$在 $x=x_k$ 处泰勒(Taylor)级数中一次项近似 $f(x)$, 然后求线性方程的根, 即

$$f(x)\approx f(x_k)+f'(x_k)(x-x_k)=0$$

图 1-1　以直代曲

将它的解记作 x_{k+1}, 重复这一过程则产生序列

$$x_{k+1}=x_k-\frac{f(x_k)}{f'(x_k)},\quad k=0,1,\cdots \tag{1.5}$$

称为牛顿(Newton)法, 它是局部线性化与迭代法结合产生的, 是一个具有 2 阶收敛的迭代法.

另一个用直线逼近 $y=f(x)$的方法是在曲线上任取两点, 然