〔匈〕Ferenc Szidarovszky
〔美〕Sidney Yakowitz 著

数值分析的原理及过程

上海科学技术文献出版社

数值分析的原理及过程

〔匈〕Ferenc Szidarovszky
〔美〕Sidney Yakowitz 著
施明光、潘仲雄 译
黄育仁 校

上海科学技术文献出版社

Principles and Procedures
of Numerical Analysis

Ferenc Szidarovszky
and
Sidney Yakowitz

Plenum Press· New York and London, 1978

数值分析的原理及过程
施明光、潘仲雄 译
黄育仁 校
*
上海科学技术文献出版社出版
(上海高安路六弄一号)
新华书店上海发行所发行
上海商务印刷厂印刷
*
开本 787×1092 1/32 印张 11 字数 261,000
1982年6月第1版 1982年6月第1次印刷
印数: 1—9,600

书号: 13192·40 定价: 1.35 元

序　言

众所周知，数值分析方法实际上已比较多地应用在科学探索的每个定量领域(虽然不一定总是有效)．本书是针对工科和数学专业的高年级学生及研究生水平的人而写的，所选的内容，通常可在数值分析教材中找到．但是对那些根据经验和分析明显不合用的传统方法，则不予选用．其次在本书中，我们的基本愿望之一是使读者具有应用数值分析方法思考非传统课题的能力．因为使用计算机较多的学科诸如最优化、统计学、系统分析和系统识别，非常需要类似于这里分析经典数值分析问题的那些方法．

为了使读者能清楚地了解数值方法的结构，我们专门安排了一章度量空间的理论，供论述算子迭代时应用．该章中，我们收集了一些有关实变函数和泛函分析的定义和概念，这是数值分析原理的阐述达到一个近代中等水平所必不可少的，此外，我们对迭代过程导出了它的抽象理论(主要是压缩映象原理)．在以后各章中，我们就可以根据需要不再作进一步的推导而直接从该理论得出标准迭代法(例如：线性代数方程组的 Gauss-Seidel 法，非线性方程中的 Newton-Raphson 法和微分方程中的预估——校正法)是收敛的结论．另外，在特定的数值分析情况下，立刻就可以从一般的理论中推导出这些方法的标准误差界．通过这些方法的论述，读者将会看到，对于不同的数值分析问题，表面看来多种多样的数值方法都有相同的结构．

试图写一本包括传统数值分析题材的教材，必须权衡下面

的要求：一方面希望能够完整地提供各种有效的数值分析过程；而另一方面要给出数值分析主要原理的(独立的)详细数学论证，而目前有几部大部头著作，不加限制地在这两方面探讨，所以不宜作为教材．我们采用的办法是两者的折衷，仅将那些可靠性、有效性和收敛性均得到证明的最重要的方法包括进来．而对有关的专门内容，则列出了详尽的参考文献，使读者能找到其它的数值方法．同样，对于我们这里所讨论的方法的基本性质及误差属性，大部分均给出了完整的与严格的数学论证．但是除了那些与数学发展特别有关或有指导意义的以外，一般均略去了论证的推导．因为我们认为这些纵然对读者是有价值的但却不是必不可少的，然而，我们也列出了引文，说明在什么地方，可以找到这些方法的详细推导．

本书曾用复印形式，对工科和数学专业的四年级学生使用过．全书基本内容作为两个学期课程的教材，教学时间是充裕的．不然，略去特征值方法和偏微分方程方法两章，并适当删简某些内容(一般是每章的后面部分)．我们曾用本书作为一学期课程的教材，结果也是成功的．阅读本书时，形式上必须具备的条件是先熟悉微积分和矩阵理论．高等微积分或高级工程数学课程，对学习本书也会有所帮助，但不是决定性的．而真正的条件是必须对数学的抽象概念有兴趣，以及对数值分析基础结构的深入了解有一定的要求．

我们还用实例来补充数值分析理论的发展，使其内容更丰富，更系统化．在可能的情况下，我们通过例题，对同一问题，采用不同的数值分析方法求解，从而使读者对不同算法的精确度及计算工作量的比较有一个根据．通过匈-美国家科学基金会的合力资助，使作者得以相遇，并且在水文学数值模型的研究方面进行协作．本书的第一稿是 F. Szidarovszky 在担任 Arizona

大学系统和工业工程系的访问教授时写的，我们十分感谢该系的同行们，特别是秘书处工作人员给我们的支持、帮助和合作.

在本书编著过程中，我们麻烦了许多朋友和熟人，并给研究生助教增添了不少麻烦，在这些研究生助教中，特别要感谢 Arik Kashper, Dan Murray 以及 Ed Zucker. 数值分析家 Warren Ferguson, Michael Golberg, Gene Golub, Dennis McCaughey, Olig Palusinski 以及 John Starner 付出了大量的时间，Golub 教授让我们翻阅了他收集的资料，Ferguson 教授对初稿及定稿提出了详细而有价值的建议.

F. Szidarovszky

S. Yakowitz

目　录

第一章 预备知识

1.1 数制和数的表示

在我们习惯的数制中，符号 649 是指

$$6\times10^2+4\times10+9$$

的和. 换句话说，这样的一个符号表示各系数取自集合 $\{0, 1, 2, \cdots, 9\}$ 中的一个以 10 为基的多项式的值. 以 10 为基数制的原由可能起因于人们用手指计数. 在另外一些学者中，也曾用过不同的数制. 例如，巴比伦天文学家曾用一种以 60 为基的数制，其影响之一，可以从分圆周为 360 度中得到考证.

下面整个论述中，N 均表示大于 1 的整数，基为 N 的数制的系数是 $0, 1, 2, \cdots, N-1$，这样，在该数制中一个正整数的表示必为

$$a_0+a_1N+a_2N^2+\cdots+a_kN^k$$

的形式，这里 k 是非负整数，$a_0, a_1, \cdots, a_k$ 是不超过 $N-1$ 的非负整数. 整数 N 通称为数制的基(或基数).

其次，我们证明如何将整数用任何希望的基表示出来. 设 M_0 是任何正整数. 表示式

$$M_0=a_0+a_1N+a_2N^2+\cdots+a_kN^k \tag{1.1}$$

意味着 a_0 是 M_0 被 N 除的余数，因为 M_0-a_0 的展开式的各项均为 N 的倍数. 令

$$M_1=(M_0-a_0)/N.$$

则由(1.1)得

$$M_1=a_1+a_2N+\cdots+a_kN^{k-1},$$

这样，a_1 等于 M_1 被 N 除的余数. 重复该过程，所有数 a_0, a_1, …, a_{k-1} 均可逐一求得. 如果 $M_k<N$ 则算法终止，且这种情况意味着 $a_k=M_k$.

令 $[x]$ 表示实数 x 的整数部分，我们可以将上述的算法概括为：

$$\begin{aligned}
&M_0=M,\\
&M_1=[M_0/N], \qquad a_0=M_0-M_1N,\\
&M_2=[M_1/N], \qquad a_1=M_1-M_2N,\\
&\quad\vdots \qquad\qquad\quad \vdots \qquad\qquad (1.2)\\
&M_k=[M_{k-1}/N], \quad a_{k-1}=M_{k-1}-M_kN,\\
&a_k=M_k,
\end{aligned}$$

由这些讨论以及除法中商和余数的唯一性可以推出下面的定理：

定理 1.1 设 N 和 M 是任意整数，且 $N>1$, $M\geqslant 0$. 则整数 M 在以 N 为基的数制中能唯一地表示.

考虑以 N 为基的数制中一个数的表示形式，这里 N 异于 10. 我们写出符号

$$(a_ka_{k-1}\cdots a_2a_1a_0)_N,$$

其意思是指由和式

$$M=a_kN^k+\cdots+a_2N^2+a_1N+a_0, \tag{1.3}$$

所决定的数. 可以用循环关系式计算(1.3)的值. 构造序列

$$\begin{aligned}
&b_k=a_k,\\
&b_{k-1}=b_kN+a_{k-1},\\
&\quad\vdots \qquad\qquad (1.4)\\
&b_1=b_2N+a_1,\\
&b_0=b_1N+a_0,
\end{aligned}$$

易证 $M=b_0$. 序列(1.4)称为 Horner 法则，由于允许 N 是任

一实数，故可用它来求多项式的值.

例 1.1 (a) 对 $M=649$ 和 $N=3$ 的情况，得到

$$a_0=1,\quad M_1=\frac{649-1}{3}=\frac{648}{3}=216,$$

$$a_1=0,\quad M_2=\frac{216-0}{3}=\frac{216}{3}=72,$$

$$a_2=0,\quad M_3=\frac{72-0}{3}=\frac{72}{3}=24,$$

$$a_3=0,\quad M_4=\frac{24-0}{3}=\frac{24}{3}=8,$$

$$a_4=2,\quad M_5=\frac{8-2}{3}=\frac{6}{3}=2<3,$$

最后一式推得

$$a_5=2,$$

这样，得

$$649=(220001)_3.$$

(b) 现计算数 $(220001)_3$ 的值. 利用(1.4)得到

$$b_5=2,$$
$$b_4=2\times3+2=8,$$
$$b_3=8\times3+0=24,$$
$$b_2=24\times3+0=72,$$
$$b_1=72\times3+0=216,$$
$$b_0=216\times3+1=649.$$

现在考虑实数 x 的表示形式，$0\leqslant x<1$. 希望采用表示式

$$x=a_{-1}N^{-1}+a_{-2}N^{-2}+\cdots, \tag{1.5}$$

这里系数 a_{-1}, a_{-2}, … 是不超过 $N-1$ 的非负整数.

设 $x_0=x$，以 N 乘(1.5)的两边，得到

$$Nx_0=a_{-1}+a_{-2}N^{-1}+\cdots, \tag{1.6}$$

这里 $0\leqslant Nx_0<N$，于是推得 a_{-1} 等于 Nx_0 的整数部分，即

$a_{-1}=[Nx_0]$，令

$$x_1=Nx_0-a_{-1}.$$

则利用(1.5)得到

$$x_1=a_{-2}N^{-1}+a_{-3}N^{-2}+\cdots,$$

重复应用上面的过程，可得系数 a_{-2}，a_{-3}，…上述算法可写成下列形式

$$\begin{aligned} &x_0=x, && a_{-1}=[Nx_0],\\ &x_1=Nx_0-a_{-1}, && a_{-2}=[Nx_1],\\ &x_2=Nx_1-a_{-2}, && a_{-3}=[Nx_2]. \end{aligned} \tag{1.7}$$

过程(1.7)可以无限地连续进行下去，或者当数 x_0，x_1，x_2，…中出现零时就终止. 易证，如果 a_{-1}，a_{-2}，… 是由上面过程得到的系数，则级数(1.5)的值为 x. 因而，记号 $(0.a_{-1}a_{-2}a_{-3}\cdots)_N$ 是 x 的一个明显表示形式. 通过这些讨论我们证明了下面的定理.

定理 1.2 设 N 是一个大于 1 的整数，任一实数 x，$0\leqslant x<1$ 可用下式

$$x=a_{-1}N^{-1}+a_{-2}N^{-2}+a_{-3}N^{-3}+\cdots,$$

表示，其中系数 a_{-1}，a_{-2}，a_{-3}，… 是不超过 $N-1$ 的非负整数.

如下例(在以 10 为基的数制中)所示，表示式(1.5)不是唯一的，

$$0.1=0.09999\cdots,$$

因其右端值等于

$$\begin{aligned} 9[10^{-2}+10^{-3}+\cdots]&=9\frac{10^{-2}}{1-10^{-1}}=9\frac{10^{-1}}{9}\\ &=10^{-1}=0.1. \end{aligned}$$

对于满足定理 1.2 条件的系数的每一集合，级数(1.5)总是

收敛的，因为它能被收敛和

$$(N-1)\left[N^{-1}+N^{-2}+N^{-3}+\cdots\right]$$
$$=(N-1)\frac{1}{N}\frac{1}{1-1/N}=1,$$

所控制. 当(1.5)是有限项和的情况，在(1.4)中以 N^{-1} 代替 N，以 a_{-j} 代替 a_j 并且令 $a_0=0$，于是我们可以用 Horner 法则由数 x 的系数 a_{-1}，…，a_{-k}，求出其值.

定理 1.1 和 1.2 意味着任一正实数 A 在以 N 为基($N>1$，整数)的数制中，可用式

$$A=\sum_{l=-\infty}^{k}a_lN^l, \tag{1.8}$$

来表示. 其中系数 a_l 是不超过 $N-1$ 的非负整数. 注意表示式(1.8)不必唯一，为得到表示式(1.8)的系数，当 $M=[A]$ 和 $x=A-M$ 我们可分别使用算法(1.2)和(1.7).

例 1.2 (a) 设 $x=0.5$ 及 $N=3$；则由(1.7)得

$$x_0=0.5,\quad a_{-1}=[1.5]=1,$$
$$x_1=0.5,\quad a_{-2}=[1.5]=1,$$
$$\vdots\qquad\qquad\vdots$$

于是 $0.5=(0.1111\cdots)_3$.

(b) 表示式 $(0.1111\cdots)_3$ 的值可由无穷级数

$$3^{-1}+3^{-2}+3^{-3}+3^{-4}+\cdots=\frac{1}{3}\frac{1}{1-1/3}=\frac{1}{2}=0.5,$$

的和求得.

(c) 将例 1.1(a)和本例结合起来，我们就知道

$$649.5=(220001.1111\cdots)_3.$$

近代计算机使用以 2 为基的数制，因为在这种情况下，仅用到两个符号，即 0 和 1，它们在物理上代表“触发”电路或磁性体中磁力线方向的两个稳定状态. 这样，在二进制(以 2 为基的数

制)中,实数可表示为

$$A=\sum_{l=-\infty}^{k} a_l 2^l,$$

其中每个系数 a_l 的值或是 0 或是 1.

负实数可在其绝对值前加一负号来表示. 在每种数制中零均用 0 表示.

在电子计算机的存储器中,仅能存放有限个符号,因而,通常在计算机中的数仅能表示成近似到所指望的位数. 现在讨论在计算机中,对实数逼近和运算的三种类型的算法.

这里讨论的定点算法形式,要求数 x 的大小必须小于 1,因此其表示形式为

$$x_{fi}=\pm\sum_{l=1}^{t} a_{-l}2^{-l}, \tag{1.9}$$

式中 t 是依赖于计算机结构的整数,且称 x_{fi} 为定点机器数(字长为 t). 显然,在区间

$$R=[-1+2^{-t-1},\ 1-2^{-t-1}],$$

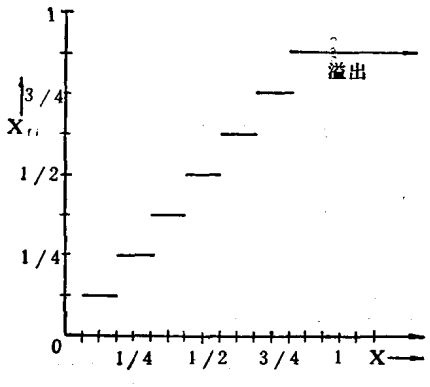

图 1.1 x 的定点近似

内的数可用(1.9)表示,且区间 R 中的任一实数 x,可用形如(1.9)的最接近的数来逼近,它的绝对误差 $|x-x_{fi}|$ 不超过 2^{-t-1}. 当表示式中包含异于 2 的基或当指数 l 的范围不同时,也容易仿照分析. 当 $t=3$ 时我们将 x_{fi} 对照 x 描在图 1.1 中.

如果运算结果不属于区间[−1, 1]则出现溢出,溢出虽相