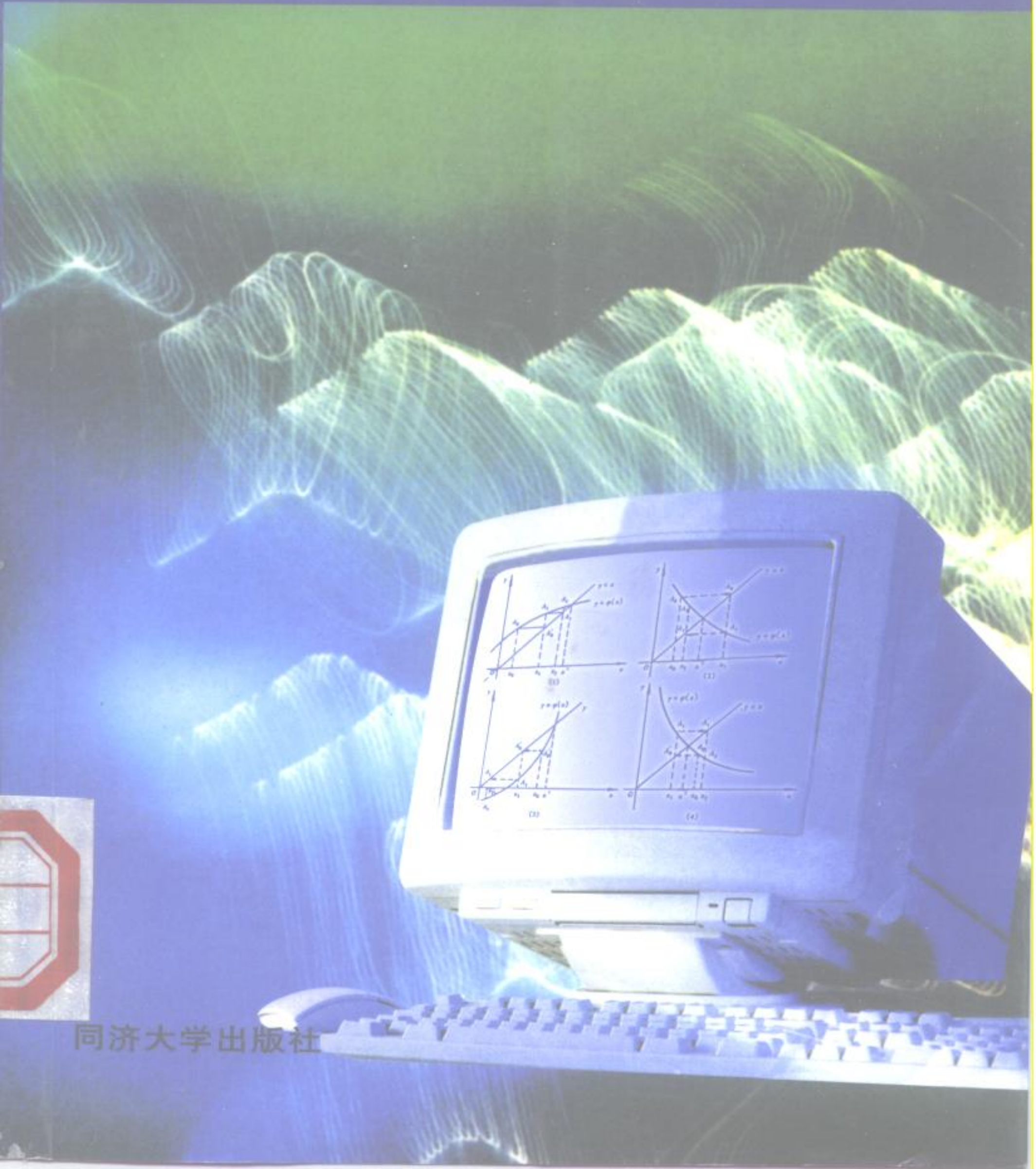
数值计算基础

沈剑华 编



同济大学出版社

图书在版编目(CIP)数据

数值计算基础/沈剑华编 .—上海:同济大学出版社,1999
ISBN 7-5608-2061-1
Ⅰ.数… Ⅱ.沈… Ⅲ.数值计算-理论 Ⅳ.0241
中国版本图书馆 CIP 数据核字(1999)第 21138 号

责任编辑 林武军
封面设计 陈益平

数值计算基础
沈剑华 编
同济大学出版社出版
(上海市四平路 1239 号 邮编:200092)
新华书店上海发行所发行
大丰市印刷二厂印刷
开本:850×1168 1/32 印张:8.75 字数:250 千字
1999 年 8 月第 1 版 1999 年 8 月第 1 次印刷
印数:1—4000 定价:12.00 元
ISBN7-5608-2061-1/O·174

前　言

数值计算是计算机科学的重要内容，当前，由于科学技术的迅速发展和计算机的广泛应用，使继实验方法、理论方法之后，科学计算已成为科学研究的第三种方法，学习和掌握计算机上常用的数值计算方法及有关的基础理论，已成为现代科学教育的重要内容，这方面的知识对于当代的工科大学生来说，是非常需要的.

本书是为同济大学工科大学生学习“计算方法”课程而编写的基础教材，全书共分六章，内容主要包括：插值与逼近，数值积分与数值微分，非线性方程的数值解法，线性代数方程组的数值解法，常微分方程初值问题的数值解法等计算机上常用的数值计算方法及有关的理论.每章都给出典型例题并配有一定数量习题，便于练习.书后附有习题答案，并在附录中列举了几个数值方法应用程序，便于上机实习和提高数值计算工作的能力.

本书可作为工科大学生学习“计算方法”课程的基础教材，也可作为要想了解这方面知识的工程技术人员的参考用书.本书是编者多年来在同济大学为工科研究生和本科大学生讲授“数值分析”和“计算方法”课程的教材基础上经反复整理、修改和补充而写成的，在编写过程中参考了国内外的较多的优秀教材、文献资料，并总结了编者在教学实践中的体会，使本书在内容上尽力做到重概念，重方法，重应用，重能力的培养，并在阐明各种数值计算方法的同时，作必要的理论分析和论证.

本书内容安排由浅入深，通俗易懂，完全按照教学规律编写，

易于教学，便于自学．阅读本书只要求读者具备高等数学、线性代数和算法语言的基础知识．全书讲授时间约为 50～60 学时．

在本书编写过程中，得到计算数学教研室许多老师的帮助，特别是黄自萍教授和陈素琴老师均仔细地审阅了全书，并提出了宝贵意见．在本书的出版过程中，得到同济大学教务处、教材科、出版社的大力支持和帮助，在此，笔者对为本书的出版给予过帮助和付出辛勤劳动的同志们一并谨表诚挚的谢意．

因编者水平有限，书中难免有错误和不妥之处，诚恳希望读者批评指正．

编者

1999 年 7 月

目　录

第一章　引论 …………………………………………… (1)
1.1　数值计算方法的对象和特点 …………………… (1)
1.2　误差 …………………………………………… (6)
1.3　数值计算中应注意的一些问题 ………………… (10)
习题一 ……………………………………………… (15)
第二章　插值与逼近 ………………………………… (17)
2.1　插值的基本概念 ……………………………… (17)
2.2　拉格朗日(Lagrange)插值 …………………… (20)
2.3　牛顿(Newton)插值 ………………………… (25)
2.4　埃尔米特(Hermite)插值 …………………… (31)
2.5　三次样条插值 ………………………………… (37)
2.6　*B*-样条函数 ………………………………… (47)
2.7　正交多项式 …………………………………… (51)
2.8　最佳平方逼近 ………………………………… (57)
2.9　曲线拟合的最小二乘法 ……………………… (63)
习题二 ……………………………………………… (69)
第三章　数值积分与数值微分 ……………………… (73)
3.1　数值积分概述 ………………………………… (73)
3.2　牛顿-柯特斯(Newton-Cotes)求积公式 ……… (77)
3.3　自适应积分法 ………………………………… (88)
3.4　龙贝格(Romberg)求积算法 ………………… (92)

3.5 高斯(Gauss)求积方法 …………………………… (98)
3.6 数值微分 ……………………………………… (109)
习题三 ……………………………………………… (114)
第四章 非线性方程的数值解法 ………………… (117)
4.1 二分法 ………………………………………… (118)
4.2 迭代法 ………………………………………… (122)
4.3 迭代法的收敛阶和加速收敛方法 …………… (128)
4.4 牛顿迭代法 …………………………………… (132)
4.5 弦截法 ………………………………………… (138)
习题四 ……………………………………………… (140)
第五章 线性代数方程组的数值解法 …………… (142)
5.1 高斯(Gauss)消去法 ………………………… (144)
5.2 三角分解法 …………………………………… (152)
5.3 解带状方程组的三角分解法 ………………… (173)
5.4 范数与方程组的状态 ………………………… (178)
5.5 迭代法 ………………………………………… (188)
习题五 ……………………………………………… (206)
第六章 常微分方程初值问题的数值解法 ……… (210)
6.1 欧拉(Euler)方法 …………………………… (211)
6.2 龙格-库塔(Runge-Kutta)方法 …………… (220)
6.3 收敛性与稳定性 ……………………………… (230)
6.4 线性多步法简介 ……………………………… (236)
6.5 一阶常微分方程组和高阶方程 ……………… (241)
习题六 ……………………………………………… (246)
附录 部分数值方法的计算实例 ………………… (248)
习题答案 ………………………………………… (263)
实习题答案 ……………………………………… (270)
参考书目 ………………………………………… (272)

第一章　引　论

1.1　数值计算方法的对象和特点

一、研究的对象

数值计算方法是近代数学的一个重要分支,它是研究各种数学问题的**数值解法**(近似解法),包括方法的构造和求解过程的理论分析.

数值计算方法又称为**计算方法**或**数值分析**,当前是一门与计算机应用密切结合的实用性很强的数学课程.

利用计算机解决科学计算问题一般有以下几个大过程:

$\boxed{\text{实际问题}} \to \boxed{\text{构造数学模型}} \to \boxed{\text{选择数值计算方法}} \to \boxed{\text{程序设计}} \to \boxed{\text{上机计算求出结果}}$

可见,研究怎样通过计算机所能执行的基本运算,求得各类数学问题的**数值解**,这就是数值计算的根本任务.更确切地说,数值计算方法是对给定问题的输入数据和所需计算结果之间关系的一种明确的描述.

众所周知,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.例如,有人统计过,在 1955 ~ 1975 年的 20 年间,计算机的计算速度提高了约数千倍,而同一时间内解决一定规模的椭圆型偏微分方程的计算方法效率竟提高了 100 万倍!这说明研究和选择好的数值计算方法对提高计算速度,在某种意义上说比提高计算机速度更重要,因为计算方法研究所付出的代价比改进计算机的硬件设备毕竟要小得多.

随着计算机的发展与普及,继实验方法、理论方法之后,科学

计算已成为科学实践的第三种手段，求解各种数学问题的数值方法已被广泛应用于不仅是自然科学，还包括生命科学，经济科学和社会科学等各领域.因此，数值计算的内容是相当丰富的，本书仅向大家介绍在微积分、线性代数、常微分方程等基础数学中常用的、行之有效的数值计算方法，并通过典型例子阐明构造算法的基本思想和技巧，引出相应算法的步骤，便于大家更好地应用.

本书的内容可分为三大部分：

(1) 数值逼近与数值微积分；

(2) 数值代数；

(3) 常微分方程的数值解法.

二、主要特点

数值计算方法是以数学问题为研究对象的，它是数学的一个分支，只是它不象纯数学那样只研究数学本身的理论，而着重研究求解的数值计算方法及与此相关的理论，包括方法的收敛性、稳定性及误差分析，还要根据计算机特点研究计算时间最省的计算方法.有的方法尽管在理论上还不够严格，但通过实际计算，对比分析等手段，被证明是行之有效的方法也可采用.因此，数值计算既有纯数学的高度抽象性与严密科学性的特点，又有应用的广泛性与数值试验的高度技术性的特点，当前已发展成为一门与使用计算机密切结合的实用性很强的数学课程.它的任务就是提供在计算机上实际可行的，理论可靠的，计算复杂性好的各种数值方法.除此以外，它还有以下几个基本特点：

1. 采用“构造性”方法

数值方法中许多问题的存在性的证明都是以“构造性”方法为基础.所谓用构造性的方法证明一个问题的存在性，就是指具体地把这个问题的计算公式构造出来，这种方法不但证明了问题的存在性，而且有了具体的计算公式，就便于编制程序上机计算.

我们用一个简单的例子说明这个特点.

例 1.1 对命题"实系数二次方程

$$x^2+2bx+c=0$$

当 $b^2>c$ 时,有两个实根."

分别给出构造性与非构造性的证明.

证明 (1) 非构造性证明(如用反证法)

令 $$f(x)=x^2+2bx+c$$

若设方程无实根,即 $f(x)$没有零点,从而 $f(x)$恒不为零.由于 $f(x)$是 x 的连续函数,可知对所有 x,或者 $f(x)$是恒正的,或者 $f(x)$是恒负的.

根据对 b 和 c 所附加的条件

$$b^2>c$$

有 $$f(-b)=b^2-2b^2+c=c-b^2<0$$

因而 $f(x)$必须是恒负的.

另一方面,当$|x|$很大时,$x^2>|2bx+c|$,因此当$|x|$很大时,又有 $f(x)>0$,从而导出矛盾.这矛盾的由来是假设方程无实根,故假设错误,从而 $f(x)=0$ 至少有一个实根.

其次,若设方程仅有一个实根,则由 $f(x)$的连续性必有

① 当 x 很大时,$f(x)>0$; $-x$ 很大时,$f(x)<0$.

或者② 当 x 很大时,$f(x)<0$; $-x$ 很大时,$f(x)>0$.

这和无论 x 取什么符号,只要$|x|$很大,就有 $f(x)>0$ 的事实相矛盾.所以方程有两个实根.但上述证明过程并没有提供求根的方法.称之为"非构造性证明".

(2) 构造性证明

对任意 x,b 和c,由

$$\begin{aligned}x^2+2bx+c&=x^2+2bx+b^2-b^2+c\\&=(x+b)^2+c-b^2\end{aligned}$$

因此,当且仅当

$$(x+b)^2+c-b^2=0$$

时，x 是它的根，亦即

$$(x+b)^2=b^2-c$$

将两端开方并注意 $b^2-c>0$，可以推出

当且仅当
$$x+b=\pm\sqrt{b^2-c}$$
时，x 是根，从而

$$x=-b\pm\sqrt{b^2-c} \tag{1.1}$$

这样就证明了方程存在两个实根，并且可以根据(1.1)式具体算出这两个实根，即同时得到计算根的方法.

在本书的各章内容中，许多存在性证明都是运用了构造性的方法. 亦即这种证明同时还提供了具体的计算公式，使需要证明其存在的对象通过数值演算过程来完成.

2. 采用“离散化”方法

把求连续变量问题转化为求离散变量问题，称为**离散化**.

一个连续的数学问题要上机计算，必须进行离散化. **离散化**可以认为是数值计算中最基本的概念和方法之一. 例如把定积分离散成求和，把微分方程离散成差分方程等等.

例 1.2 计算定积分

$$I=\int_a^b f(x)\mathrm{d}x \tag{1.2}$$

解 这是一个连续性的数学问题，直接在计算机上计算有困难. 但大家知道，定积分可用复合梯形公式近似计算，即

$$\int_a^b f(x)\mathrm{d}x \approx \frac{b-a}{n}\left[\frac{1}{2}(y_0+y_n)+y_1+\cdots+y_{n-1}\right] \tag{1.3}$$

其中 $a=x_0<x_1<\cdots<x_n=b$ 是 n 等分区间 $[a,b]$ 的分点，$y_0,y_1,\cdots,y_n$ 是被积函数在各分点的函数值. 公式(1.3) 就是把定积分离散成为求和运算. 用(1.3) 的近似公式就可以求出积分(1.2) 的近似值.

3. 采用“递推化”方法

所谓**递推化**，其基本思想就是将一个复杂的计算过程归结为

简单过程的多次重复. 由于递推化算法便于编写计算机程序,所以数值计算中的许多数值方法常常采用“递推化”方法.

例 1.3 对给定的 x,计算多项式

$$P_n(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$$

的值.

解 可利用递推公式

$$\begin{cases} u_0 = a_n \\ u_k = u_{k-1}x + a_{n-k} \end{cases} \tag{1.4}$$

对 $k = 1,2,\cdots,n$ 反复执行(1.4)式,最终得到的 u_n 就是多项式 $P_n(x)$ 的值.

用公式(1.4)计算多项式 $P_n(x)$ 的值不但逻辑结构简单,而且计算量最小. 这个方法就是历史上著名的**秦九韶**方法.

4. 采用“近似替代”方法

计算机运算必须在有限次停止,所以数值方法常表现为一个无穷过程的截断,把一个无限过程的数学问题,转化为满足一定误差要求的有限步来近似替代.

例 1.4 计算无理数 e 的近似值.

解 根据泰勒公式得

$$\mathrm{e} = 1 + 1 + \frac{1}{2!} + \cdots + \frac{1}{n!} + \cdots \tag{1.5}$$

这是一个无限过程,计算机无法实现,只能在(1.5)式中取有限项计算,再估计误差. 若取

$$\mathrm{e} \approx 1 + 1 + \frac{1}{2!} + \cdots + \frac{1}{n!}$$

其误差为

$$|R_n| < \frac{\mathrm{e}}{(n+1)!} < \frac{3}{(n+1)!}$$

以上这些特点,将在以后各章中进一步体现,了解了这些特点,对学习数值计算方法很有好处.

1.2 误 差

许多数值方法给出的解答仅仅是所要求的真解的某种近似,因而研究数值方法,必须注重误差分析,分析误差的来源,误差的传播情况以及对计算结果给出合理的误差估计.

一、误差的来源

误差的来源是多方面的,但主要有以下几种:

1. 模型误差

用计算机解决科学计算问题首先要建立数学模型,它是对被描述的实际问题进行抽象、简化而得到的.因而总是近似的,这就不可避免地要产生误差,我们把这种数学模型的解与实际问题的解之间出现的误差称为**模型误差**.

2. 观测误差

在数学模型中通常总包含有一些观测数据,如温度,长度,电压等等,这些数据的值一般是由观测或实验得到的,由于观测不可能绝对准确,由此产生的误差称为**观测误差**.

3. 截断误差(也称为方法误差)

由实际问题建立起来的数学模型,在很多情况下要得到准确解是困难的.当数学模型不能得到准确解时,通常要用数值方法求它的近似解,例如常把无限的计算过程用有限的计算过程代替,这种模型的准确解和数值方法的准确解之间的误差称为**截断误差**.因为截断误差是方法固有的,又称为**方法误差**.

例 1.5 函数 $f(x)$ 的泰勒(Taylor) 展开式

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n$$

$$+\frac{f^{(n+1)}(\xi)}{(n+1)!}(x-x_0)^{n+1}$$

其中 ξ 在 x_0 与 x 之间.

若记
$$S_n(x)=\sum_{k=0}^{n}\frac{f^{(k)}(x_0)}{k!}(x-x_0)^k$$
$$R_n(x)=\frac{f^{(n+1)}(\xi)}{(n+1)!}(x-x_0)^{n+1}$$

则
$$f(x)=S_n(x)+R_n(x)$$

若取
$$f(x)\approx S_n(x)$$

则截断误差为 $R_n(x)$,也即用 $f(x)$的泰勒展开式的部分和 $S_n(x)$来近似函数 $f(x)$,其余项 $R_n(x)$就是真值 $f(x)$的截断误差.

4. 舍入误差

由于计算机的字长有限,原始数据在计算机上表示会产生误差,每一次运算又可能产生新的误差,这种误差称为**舍入误差或计算误差**.

上述几种误差都会影响计算结果的准确性,数值计算中除了研究求解数学问题的数值方法外,还要研究计算结果的误差是否满足精度要求,这就是误差估计问题.但前二种误差往往不是计算工作者所能独立完成的,数值计算中用不到描述自然现象,也用不到观测测量,而主要研究的是截断误差与舍入误差对计算结果的影响.

重视误差分析和控制误差扩散是十分重要的,没有误差分析的数值计算结果是不可信的.

二、绝对误差、相对误差和有效数字

如何来定义误差? 人们常用**绝对误差、相对误差或有效数字**来说明一个近似值的准确程度.

1. 绝对误差和相对误差

定义 1.1 设 x 为准确值，x^* 为 x 的一个近似值，称 $e_a = x^* - x$ 为近似值 x^* 的**绝对误差**，简称**误差**.

必须指出，通常无法得到准确值 x，从而不可能得到 x^* 的绝对误差 e_a 的真值，只能根据测量的情况，估计出误差的绝对值的一个上界 ε_a，即

$$|e_a| = |x^* - x| \leqslant \varepsilon_a$$

这个正数 ε_a 通常叫做近似值 x^* 的**绝对误差限**. 有了绝对误差限，就可知道真值 x 的范围

$$x^* - \varepsilon_a \leqslant x \leqslant x^* + \varepsilon_a$$

绝对误差的大小，在许多情况下还不能完全刻划一个近似值的准确程度，例如测量 1000 米和 1 米两个长度，若它们的绝对误差都是 1 厘米，显然前者的测量比较准确. 由此可见，决定一个量的近似值的精确度，除了考虑绝对误差的大小外，还需要考虑该量本身的大小，为此引入**相对误差**的概念.

定义 1.2 设 x 为准确值，x^* 是近似值，则称

$$e_r = \frac{e_a}{x} = \frac{x^* - x}{x} \qquad (x \neq 0)$$

为近似值 x^* 的**相对误差**.

在实际计算中，由于真值 x 一般是不知道的，但可以证明，当 e_r 较小时，e_r 中分母 x 可用 x^* 代替，其两者之差是 e_r 的高阶无穷小，故可忽略不计.

相对误差可正可负，它的绝对值上界叫做**相对误差限**，即如果有正数 ε_r 使

$$|e_r| \leqslant \varepsilon_r$$

则称 ε_r 为 x^* 的**相对误差限**.

在误差分析中，相对误差比绝对误差更重要.

2. 有效数字

为了可以从近似数的有限位小数表示本身就能知道近似数的

精度，我们引入有效数字概念.大家知道，当 x 有很多位数字时，常按照"四舍五入"原则，取 x 的前几位数字作为 x 的近似值 x^*.

例如：

$$x = \sqrt{2} = 1.414\ 213\ 562\ 37\cdots$$

若只取到小数后四位数字得

$$x^* = 1.4142$$

其误差为 $0.00001356\cdots$，误差限为 $0.00005 = \frac{1}{2} \times 10^{-4}$. 此时称 x^* 准确到小数后第四位，并称由此位算起的前五位数字 14142 为 x^* 的有效数字.

一般说来，若近似值 x^* 的误差不超过某位数字的半个单位，而从该位数字到 x^* 最左边的那个非零数字(即自左向右看，第一个出现的非零数字)共有 n 位，那么这 n 位数字都称为 x 的有效数字，即

$$x^* = \overbrace{\times\ \times \cdots \times}^{n\text{位}}$$

误差不超过该位数的半个单位.

自左向右看，第一个非零数.

并称近似值 x^* 具有 n 位有效数字.

例如：

$$|e - 2.718| < 0.0005 = \frac{1}{2} \times 0.001$$

故 e 的近似值 2.718 具有四位有效数字.

具有 n 位有效数字的近似值 x^* 也可以用指数形式写成：

$$x^* = \pm a_1.\ a_2 a_3 \cdots a_n \times 10^m$$

其中，a_1 是 1 到 9 中的一个数字，$a_2, \cdots, a_n$ 是 0 到 9 中的一个数字，m 为整数，且 x^* 的绝对误差为

$$|e_a| = |x - x^*| \leq \frac{1}{2} \times 10^{m-n+1}$$

$a_1, a_2, \cdots, a_n$ 是 x^* 的 n 位有效数字.

同一个准确值的不同近似值,有效数字越多,它的绝对误差和相对误差就越小.有了有效数字概念后,下面两个 3.140011 的近似值的写法是有区别的:3.14 和 3.1400,前者是三位有效数字,后者则表示五位有效数字.

1.3 数值计算中应注意的一些问题

由上述讨论可知,误差分析在数值计算中是一个很重要又很复杂的问题.因为在数值计算中每一步运算都可能产生误差,而一个科学计算问题的解决,往往要经过成千上万次运算,如果每一步运算都分析误差,显然是不可能的,其实也是不必要的.人们经常这样做,通过对误差的某些传播规律的分析,指出在数值计算中应该注意的一些问题,有助于鉴别计算结果的可靠性并防止误差危害现象的产生,下面我们给出在数值计算中应该注意的一些问题.

1. 要使用数值稳定的算法

所谓**算法**,就是给定一些数据,按着某种规定的次序进行计算的一个运算序列.它是一个近似的计算过程,我们选择一个算法,主要要求它的计算结果能达到给定的精确度.一般而言,在计算过程中初始数据的误差和计算中产生的舍入误差总是存在的,而数值解是逐步求出的,前一步数值解的误差必然要影响到后一步数值解.我们把运算过程中舍入误差不增长的算法称为**数值稳定的**,否则是**数值不稳定的**.只有稳定的数值方法才可能给出可靠的计算结果,不稳定的数值方法毫无实用价值.下面举一个数值计算的例子:

例 1.6 求

$$I_n = \int_0^1 \frac{x^n}{x+5} \mathrm{d}x, \ (n = 0,1,2,\cdots,8) \tag{1.6}$$

的值.

解 由于

$$I_n + 5I_{n-1} = \int_0^1 \frac{x^n + 5x^{n-1}}{x+5} \mathrm{d}x = \int_0^1 x^{n-1} \mathrm{d}x = \frac{1}{n}$$

初值

$$I_0 = \int_0^1 \frac{1}{x+5} \mathrm{d}x = \ln 6 - \ln 5 = \ln 1.2$$

于是可建立递推公式:

$$\begin{cases} I_0 = \ln 1.2 \\ I_n = \dfrac{1}{n} - 5I_{n-1}, \ (n = 1,2,\cdots,8) \end{cases} \tag{1.7}$$

若取 $I_0 = \ln 1.2 \approx 0.182$

按(1.7)式就可以逐步算得

$$I_1 = 1 - 5I_0 \approx 0.090,$$

$$I_2 = \frac{1}{2} - 5I_1 \approx 0.050,$$

$$I_3 = \frac{1}{3} - 5I_2 \approx 0.083,$$

$$I_4 = \frac{1}{4} - 5I_3 \approx -0.165,$$

$$\cdots\cdots\cdots\cdots$$

因为在[0,1]上被积函数$\dfrac{x^n}{x+5} \geqslant 0$ (仅当 $x = 0$ 时为零),所以,I_n $(n = 0,1,2,\cdots,8)$ 是恒正的.

但在上述计算结果中,I_4的近似值却是负的,这个结果显然是错的.为什么会这样呢?这就是误差传播所引起的危害.由递推公式(1.7)可看出,I_{n-1} 的误差是扩大了 5 倍后传给 I_n,因而初值 I_0 的误差对以后各步计算结果的影响,随着 n 的增大愈来愈严重,这