Numerical Analysis and Experiments

数值分析与实验

黎健玲　简金宝　李群宏
钟献词　唐春明　编著

科学出版社

数值分析与实验

Numerical Analysis and Experiments

黎健玲　简金宝　李群宏
钟献词　唐春明　编著

科学出版社
北京

内 容 简 介

本书系统地介绍了数值分析中基本的数值计算方法和一些现代数值方法及有关理论分析，包括解线性方程组的直接法和迭代法，插值法，数值逼近，数值积分与数值微分，解非线性方程（组）的数值方法，矩阵特征值问题，常微分方程的数值解法，积分方程数值解法和最优化方法等. 对于每种常用的数值方法，不仅给出具体步骤，而且还给出了Matlab程序，便于读者调用. 同时每章配有丰富的例题、算例、上机实验题及习题，并在书末给出参考答案或证明提示. 本书阐述严谨，条理分明，深入浅出，可读性强. 本书不仅强调理论分析的严谨性，而且注重数值方法的实用性.

本书可作为高等院校信息与计算科学、数学与应用数学、计算机科学与技术等专业的本科生教材及许多理工科专业的研究生教材，也可供从事科学计算的工程技术人员参考使用.

图书在版编目(CIP)数据

数值分析与实验 / 黎健玲等编著.—北京：科学出版社，2012.8
ISBN 978-7-03-035410-5

Ⅰ.数… Ⅱ.①黎… Ⅲ.①数值计算-高等学校-教材

Ⅳ.①O241

中国版本图书馆 CIP 数据核字(2012)第 200702 号

责任编辑：杨 岭 郝玉龙 万 羽 / 封面设计：陈思思

科 学 出 版 社 出版
北京东黄城根北街16 号
邮政编码 100717
http://www.sciencep.com

成都创新包装印刷厂印刷
科学出版社发行 各地新华书店经销
*
2012 年 8 月第 一 版 开本：787*1092 1/16
2012 年 8 月第一次印刷 印张：16.75
字数：400 千字
定价：29.00 元

前　言

随着计算机科学与技术以及计算技术的飞速发展，科学计算突破了实验和理论的局限，在科学研究、经济计划与管理、工程设计等方面发挥着越来越重要的作用. 作为科学计算的核心课程，“数值分析”不仅是数学类专业本科生的必修课程，而且成为许多工科专业本科生的必修课程和许多理工科专业研究生的学位课程.

在传统的数值分析教学中，都是将数值分析作为一门理论课程来讲授，不重视算法如何在计算机上实现，更是忽视了解决实际问题的功能. 这极大地影响学生学习数值分析的兴趣和积极性，影响学生根据算法或具体问题独立编程解决问题能力的培养，从而不利于培养具有数学综合应用能力和创新能力的人才. 令人欣慰的是，越来越多的学者和教育者开始意识到这种现象，并努力采取措施使问题不断得到改善.

本书是作者在多年教学实践及科研成果的基础上，参考并吸收了当前数值分析和计算方法教材中的许多精华编撰而成的. 本书的宗旨亦即最大特点，是努力将数值分析的理论学习与利用数学软件 Matlab 编程上机实现算法紧密结合起来，提供一本真正将 Matlab 融合到教材中，完成具体算法实现的现代数值分析教材.

本书系统地介绍了数值分析中基本的数值计算方法和一些现代数值方法及其有关理论分析. 主要内容包括解线性方程组的直接法和迭代法，插值法，数值逼近，数值积分与数值微分，解非线性方程(组)的数值方法，矩阵特征值问题，常微分方程数值解法，积分方程数值解法和最优化方法.

本书除了介绍常用的算法外，还强调算法的基本原理和基本理论分析，阐述严谨、详略得当、条理分明. 各章内容相对独立，教师和读者可根据教学学时及专业需要进行取舍. 为了便于读者加深对书中内容的理解，书中精心挑选了一定量的例题，在每章之后都配有习题，在书末给出大部分习题的参考答案或证明提示.

本书介绍的数值方法都给出具体的计算步骤，并且大多数算法都给出详细的 Matlab 程序，并通过大量的算例向读者展示如何编写 m 文件调用这些现成的程序求解问题. 学习本课程应加强上机实验环节. 为方便读者练习，每章之后配有适量的上机实验题，并在书末给出参考答案.

为让读者了解和掌握约束优化的新算法，我们在书中的 11.2 节向读者介绍了不等式约束优化的三个序列二次规划(SQP)类算法，其中两个是作者近年的科研成果. 这些新算法在理论和数值实验方面都具有优良的特性.

我们在每章之后的注记中简要地总结了该章的主要内容，并向读者介绍一些相关的文献和知识，以便进一步深入学习. 为了方便教师们备课参考，我们提供了 PPT 课件(可在科学出版社主页 www. sciencep. com 免费下载).

本书可作为高等院校数学类专业、工科相关专业的本科生教材及一些理工科专业的研究生教材，并可供从事科学计算的工程技术人员参考使用. 本书的总体策划、资金筹措工作

由黎健玲和简金宝负责，全书总撰由黎健玲负责. 编写具体分工如下：第 1、2、3 章由黎健玲编写，第 4、5 章由李群宏编写，第 6、9、10 章由钟献词编写，第 7、8、11 章由简金宝和唐春明共同编写.

书末列出了参考书目和文献，作者在此谨向参考过的这些书目和文献的作者致以衷心的感谢. 编写和出版本书得到了广西大学数学与信息科学学院领导的大力支持，同时也得到了广西大学教材建设经费以及广西高校人才小高地创新团队“运筹学与最优控制”经费的资助，对此我们表示由衷的感谢. 由于作者学识水平和编撰时间所限，书中不足或错漏之处难免，恳请使用本书的师生和其他读者批评指正. 电子邮箱：numanalysis _ gxu @126. com.

作　者

2012 年 6 月 28 日

目　录

第1章 绪 论

1.1 数值分析的内容和特点

自然科学、工程技术以及社会经济等领域中遇到的许多问题，都可以应用相关的学科知识和数学理论，用数学语言描述为数学问题，即人们常说的数学模型. 然而，这些数学问题往往得不到它的准确解，或者解这种问题的计算工作量很大，只能借助计算机求其近似解(称为数值解或计算解).

随着计算机科学与技术以及计算技术的飞速发展，科学计算突破了实验和理论科学的局限，在科技发展中发挥着越来越重要的作用，科学计算和计算机模拟被称为继理论和实验之后的第3种科学研究方法.

1.1.1 数值分析的内容

数值分析是数学的一个分支，是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的一门学科. 数值分析的内容很丰富，包括函数的数值逼近(代数插值与函数的最佳逼近)、数值积分与数值微分、非线性方程数值解、数值代数、常微分方程数值解和最优化方法等.

应用计算机解决科学计算问题，需要经过下面几个主要过程：提出实际问题，建立数学模型，选用数值计算方法，设计程序以及上机计算求出数值结果.

1.1.2 数值分析的特点

数值分析既有纯数学类课程的理论抽象性与严谨性的特点，又有广泛的实用性和实验性的特点，但数值分析与纯数学类课程不同. 例如，在线性代数中，用Gramer法则求解一个n阶线性方程组需要计算$(n+1)$个n阶行列式，忽略加减运算，共需要做$(n+1)!(n-1)$次乘法. 当n很大时，这个计算量是相当惊人的. 例如，解一个20阶的线性方程组，大约要做10^{21}次乘除法. 如果用每秒百亿次的的计算机计算，则需3000多年才能完成. 由此可见，这样的方法毫无实用价值. 然而，如果采用某种数值方法，如Gauss消去法，乘除法次数不超过3000次，在微型计算机上仅需几秒钟时间就可以完成. 该例子说明研究实用的数值方法是非常必要的；在实现这些算法时还要依据计算机的容量、字长、速度等指标，研究具体求解步骤和程序设计技巧；有的数值方法在理论上虽不够严格，但通过实际计算、对比分析等手段，只要能证明它们是行之有效的，也应采用. 因此数值分析的特点可概括为以下4点：

(1)面向计算机　根据计算机特点，提供实际可行的有效算法，即算法只包括计算机能直接处理的加、减、乘、除运算和逻辑运算.

(2)有可靠的理论分析　能任意逼近并达到精度要求，对近似算法要保证收敛性和数

值稳定性，还要对误差进行分析.

(3)有好的计算复杂性　一个算法的计算复杂性包括算法的空间复杂性和时间的复杂性. 空间复杂性指算法需占用的存储空间，时间复杂性指算法包含的运算次数. 空间复杂性和时间复杂性小的算法是计算复杂性好的算法，这也是建立算法要研究的问题，关系到算法能否在计算机上实现.

(4)有数值实验　任何一个算法，除了从理论上满足上述 3 点外，还必须通过数值实验证明它是行之有效的.

1.2 误　差

1.2.1 误差的来源与分类

对数学问题进行数值求解，求得的结果往往与准确值不相等，它们的差称为**误差**. 引起误差的原因是多方面的，根据产生误差的原因不同，通常把误差分成下面 4 类：

(1) 模型误差　根据实际问题建立数学模型时，对被描述的实际问题进行了抽象和简化，因此数学模型只是实际问题的一种近似. 数学模型与实际问题之间出现的误差称为**模型误差**.

(2)观测误差　在数学模型中往往涉及一些根据观测得到的物理量，如电流、电压、温度、长度等. 由于观测手段的限制、测量仪器精密程度的影响而产生的误差称为**观测误差**.

(3)截断误差　在求解一些数学问题时，常常需要通过无限过程才能得到最终结果，但实际进行数值计算时只能采用有限过程，如无穷级数求和，只能取前面有限项之和来近似代替，这种由有限过程代替无限过程而产生的误差称为**截断误差**. 这是计算方法本身所带来的误差，所以也称为**方法误差**.

(4)舍入误差　由于计算机的字长有限，原始数据或计算的中间结果数据要用“四舍五入”或其他规则取近似，由此产生的误差称为**舍入误差**.

研究计算结果的误差是否满足精度要求就是误差估计问题，由上述误差的来源得知误差是不可避免的，误差分析和估计是数值计算过程中的重要内容. 在数值分析中主要讨论截断误差和舍入误差.

1.2.2 绝对误差、相对误差和有效数字

1. 绝对误差

定义 1.2.1　设 x^* 为准确值，x 是 x^* 的一个近似值，称 $e=x^*-x$ 为近似值 x 的**绝对误差**，简称**误差**.

这样定义的误差 e 可正可负. 通常我们不能算出准确的 x^*，也不能算出误差的准确值，只能根据测量工具或计算情况估计出误差的绝对值的上界，即存在正数 ε，满足

$$|e|=|x^*-x|\leqslant\varepsilon, \tag{1.2.1}$$

称 ε 为近似值 x 的**绝对误差限**，简称**误差限**.

例如，用毫米刻度的直尺测量一长度 x^*，测得该长度的近似值 $x=96$ mm，由于直尺以毫米为刻度，所以 x 的误差不超过 0.5 mm，即

$$|x^* - 96| \leqslant 0.5.$$

从上述不等式我们仍不能得出准确值 x^*，但可以知道 x^* 的取值范围：

$$95.5 \leqslant x^* \leqslant 96.5.$$

2. 相对误差

绝对误差限的大小不能完全表示近似值的精确程度. 为了更好地反映近似值的精确程度，除考虑误差限的大小外，还应考虑准确值 x^* 本身的大小.

定义 1.2.2 设 x 是准确值 x^* 的一个近似值，称

$$e_r = \frac{x^* - x}{x^*} \tag{1.2.2}$$

为近似值 x 的**相对误差**.

在实际计算中，由于准确值 x^* 难以求得，人们通常取

$$\bar{e}_r = \frac{x^* - x}{x} \tag{1.2.3}$$

作为 x 的相对误差，前提是 $\bar{e}_r$ 较小. 事实上，经计算可知

$$\bar{e}_r - e_r = \frac{\bar{e}_r^{\,2}}{1 + \bar{e}_r}$$

是 $\bar{e}_r$ 的平方项级，故可忽略不计.

相对误差也可正可负，它的绝对值上界称为 x 的**相对误差限**，记为 ε_r，于是有

$$|e_r| \leqslant \varepsilon_r \text{或} \ |\bar{e}_r| \leqslant \varepsilon_r.$$

3. 有效数字

在表示一个近似数时，为了能反映它的精确程度，常常用到“有效数字”的概念.

定义 1.2.3 如果近似值 x 的误差限是其某一位上的半个单位，该位到 x 的第一个非零数字共有 n 位，则称近似值 x 具有 n 位**有效数字**.

例 1.2.1 设圆周率 π 的近似值取为 $\bar{\pi}_1 = 3.14$，则 $|\pi - \bar{\pi}_1| < 0.002 < \frac{1}{2} \times 10^{-2}$，因此，$\bar{\pi}_1 = 3.14$ 有 3 位有效数字；若取 $\bar{\pi}_2 = 3.142$，则 $|\pi - \bar{\pi}_2| < \frac{1}{2} \times 10^{-3}$，因此，$\bar{\pi}_2 = 3.142$ 有 4 位有效数字；若取 $\bar{\pi}_3 = 3.141$，则 $|\pi - \bar{\pi}_3| < 0.0006 < \frac{1}{2} \times 10^{-2}$，因此，$\bar{\pi}_3$ 只有 3 位有效数字.

在科学记数法中，通常将具有 n 位有效数字的近似值 x 表示为

$$x = \pm 0.a_1 a_2 \cdots a_n \times 10^m, \tag{1.2.4}$$

即

$$x = \pm (a_1 \times 10^{-1} + a_2 \times 10^{-2} + \cdots + a_n \times 10^{-n}) \times 10^m, \tag{1.2.5}$$

其中，m 为整数，$a_1, a_2, \cdots, a_n$ 为 0～9 中的一个数字，且 $a_1 \neq 0$. 此时有

$$|x^* - x| \leqslant \frac{1}{2} \times 10^{m-n}, \tag{1.2.6}$$

即 x 的误差限 $\varepsilon = \frac{1}{2} \times 10^{m-n}$. 由此可知，在 m 相同的情况下，n 越大误差限越小，这说明一个近似值 x 的有效数字越多，其误差就越小.

至于有效数字与相对误差限的关系，有下面定理成立.

定理 1.2.1　设近似数 x 表示为

$$x = \pm(a_1 \times 10^{-1} + a_2 \times 10^{-2} + \cdots + a_l \times 10^{-l}) \times 10^m, \tag{1.2.7}$$

其中，m 为整数，$a_1, a_2, \cdots, a_l$ 是 0～9 中的一个数字，且 $a_1 \neq 0$. 如果 x 具有 n 位有效数字，则其相对误差限有估计式

$$\varepsilon_r \leqslant \frac{1}{2a_1} \times 10^{-(n-1)}; \tag{1.2.8}$$

反之，如果 x 的相对误差限 $\varepsilon_r \leqslant \frac{1}{2(a_1+1)} \times 10^{-(n-1)}$，则 x 至少具有 n 位有效数字.

证　由式(1.2.7)知 $a_1 \times 10^{-1} \times 10^m \leqslant |x| < (a_1+1) \times 10^{-1} \times 10^m$. 于是，当 x 有 n 位有效数字时，有

$$\varepsilon_r = \frac{|x^* - x|}{|x|} \leqslant \frac{\frac{1}{2} \times 10^{m-n}}{a_1 \times 10^{-1} \times 10^m} = \frac{1}{2a_1} \times 10^{-(n-1)}.$$

反之，由

$$|x^* - x| = \varepsilon_r |x| < \frac{1}{2(a_1+1)} \times 10^{-(n-1)} \times (a_1+1) \times 10^{-1} \times 10^m = \frac{1}{2} \times 10^{m-n}$$

知，x 有 n 位有效数字.

该定理表明：一个近似值的有效数字越多，其相对误差越小，从而精确度越高. 因此，在计算过程中，我们应保留尽量多的有效数字.

例 1.2.2　如果要求 $\sqrt{90}$ 的近似值的相对误差小于 0.1%，则应取多少位有效数字？

解　由定理 1.2.1 知

$$\varepsilon_r \leqslant \frac{1}{2a_1} \times 10^{-(n-1)},$$

而由 $\sqrt{90} = 10^1 \times 0.94\cdots$ 知 $a_1 = 9$，故令

$$\frac{1}{2 \times 9} \times 10^{-(n-1)} < 0.1\%,$$

解得 $n \geqslant 3$，即只要取 $n = 3$，则有 $\varepsilon_r \leqslant 0.1\%$，所以只要 $\sqrt{90}$ 的近似值取 3 位有效数字，则其相对误差限就小于 0.1%，此时 $\sqrt{90} \approx 9.49$.

1.2.3　数据误差对函数值的影响

在数值运算中，当自变量有误差时，计算函数值也会产生误差，其误差限可利用函数的 Taylor 展开式进行估计.

设 $y = f(x)$ 为一元函数，x 为准确值 x^* 的近似值，以 $f(x)$ 近似准确值 $f(x^*)$，其误差限记为 $\varepsilon(f(x))$. 由 Taylor 展开式

$$f(x^*) = f(x) + f'(x)(x^* - x) + \frac{f''(\xi)}{2!}(x^* - x)^2,$$

其中，ξ 介于 x^* 与 x 之间. 于是有

$$|f(x^*) - f(x)| \leqslant |f'(x)|\varepsilon(x) + \frac{|f''(\xi)|}{2!}\varepsilon^2(x).$$

假设 $f'(x)$ 与 $f''(x)$ 的比值不太大，可忽略 $\varepsilon(x)$ 的高阶项，从而得到函数值 $f(x)$ 的误

差限

$$\varepsilon(f(x)) \approx |f'(x)|\varepsilon(x). \tag{1.2.9}$$

进一步地,可得函数值 $f(x)$ 的相对误差限

$$\varepsilon_r(f(x)) \approx \frac{|f'(x)|\varepsilon(x)}{|f(x)|} = \left|\frac{xf'(x)}{f(x)}\right|\varepsilon_r(x). \tag{1.2.10}$$

当 $f(x)$ 为多元函数即 $y=f(x_1,x_2,\cdots,x_n)$ 时,设 $x_1,x_2,\cdots,x_n$ 为准确值 $x_1^*,x_2^*,\cdots,x_n^*$ 的近似值,则函数值 $y=f(x_1,x_2,\cdots,x_n)$ 为准确值 $y^*=f(x_1^*,x_2^*,\cdots,x_n^*)$ 的近似,其误差限仍记为 $\varepsilon(f(x))$. 记 $x^*=(x_1^*,x_2^*,\cdots,x_n^*)$, $x=(x_1,x_2,\cdots,x_n)$,则由 Taylor 展开式

$$f(x^*) \approx f(x) + \sum_{i=1}^{n} \frac{\partial f(x)}{\partial x_i}(x_i^* - x_i),$$

于是误差限为

$$\varepsilon(f(x)) \approx \sum_{i=1}^{n} \left|\frac{\partial f}{\partial x_i}\right|\varepsilon(x_i). \tag{1.2.11}$$

进一步地,可得函数值 $f(x_1,x_2,\cdots,x_n)$ 的相对误差限为

$$\varepsilon_r(f(x)) = \frac{\varepsilon(f(x))}{|f(x)|} \approx \frac{1}{|f(x)|}\sum_{i=1}^{n}\left|x_i\frac{\partial f}{\partial x_i}\right|\varepsilon_r(x_i), \tag{1.2.12}$$

其中,$\varepsilon_r(x_i)(i=1,\cdots,n)$ 为近似值 x_i 的相对误差限.

利用函数值的误差估计式(1.2.11)可得到两数和、差、积、商的误差限:

$$\varepsilon(x_1 \pm x_2) = \varepsilon(x_1) + \varepsilon(x_2), \tag{1.2.13}$$

$$\varepsilon(x_1x_2) \approx |x_2|\varepsilon(x_1) + |x_1|\varepsilon(x_2), \tag{1.2.14}$$

$$\varepsilon\left(\frac{x_1}{x_2}\right) \approx \frac{|x_2|\varepsilon(x_1) + |x_1|\varepsilon(x_2)}{|x_2|^2} \quad (x_2 \neq 0). \tag{1.2.15}$$

例 1.2.3 计算 $\sqrt{2001}-\sqrt{1999}$,并分析计算结果具有多少位有效数字.

解 记 $x_1^*=\sqrt{2001}$, $x_2^*=\sqrt{1999}$.

分别取 x_1^*,x_2^* 的具有 6 位有效数字的近似值 $x_1=44.7325$, $x_2=44.7102$.

现用两种方法计算:

方法一 $x_1^*-x_2^*=\sqrt{2001}-\sqrt{1999} \approx x_1-x_2=44.7325-44.7102=0.0223.$

方法二 $x_1^*-x_2^*=\sqrt{2001}-\sqrt{1999}=\dfrac{2}{\sqrt{2001}+\sqrt{1999}}=\dfrac{2}{x_1^*+x_2^*}\approx\dfrac{2}{x_1+x_2}$

$$=\frac{2}{44.7325+44.7102}=0.022360684\cdots\approx 0.0223607.$$

下面分析上述两种方法所得结果各具有有效数字的位数.

由式(1.2.13)知

$$\varepsilon(x_1-x_2)=\varepsilon(x_1)+\varepsilon(x_2)=\frac{1}{2}\times10^{-4}+\frac{1}{2}\times10^{-4}=10^{-4}<\frac{1}{2}\times10^{-3},$$

所以按第一种方法所得结果具有 2 位有效数字.

对于第二种方法,由式(1.2.11)知

$$\varepsilon\left(\frac{2}{x_1+x_2}\right)\approx\frac{2}{(x_1+x_2)^2}[\varepsilon(x_1)+\varepsilon(x_2)]$$
$$=\frac{2}{(44.7325+44.7102)^2}\left(\frac{1}{2}\times10^{-4}+\frac{1}{2}\times10^{-4}\right)$$
$$=0.25\times10^{-7}<\frac{1}{2}\times10^{-7},$$

所以按第二种方法所得结果具有 6 位有效数字. 因此第二种方法比第一种方法精确.

由上例可知,当两个相近的数直接相减时会造成有效数字位数减少. 因此,为防止计算精度降低,在实际计算中应该尽可能避免两相近数相减.

1.3 计算机中数的浮点表示

利用数值方法解决科学计算实际问题时,都需要设计程序并在计算机上运行才能求出数值结果,因此,了解数在计算机中的表示是有必要的.

1.3.1 以 β 为基的数系

以 β 为基的数(β 进制数)可表示为

$$x=\pm(a_{n-1}\beta^{n-1}+a_{n-2}\beta^{n-2}+\cdots+a_0\beta^0+a_{-1}\beta^{-1}+a_{-2}\beta^{-2}+\cdots+a_{-m}\beta^{-m})$$
$$=\pm(a_{n-1}a_{n-2}\cdots a_0.a_{-1}a_{-2}\cdots a_{-m})_\beta,$$

其中,$0\leqslant a_k<\beta$ 为正整数.

在计算机中广泛采用的有二进制数、八进制数、十进制数和十六进制数,它们的基数分别是 2、8、10 和 16.

例如,十进制数 $(256)_{10}=2\times10^2+5\times10^1+6\times10^0$,

$$-(3512.78)_{10}=-(3\times10^3+5\times10^2+1\times10^1+2\times10^0+7\times10^{-1}+8\times10^{-2}).$$

二进制数 $(11011)_2=1\times2^4+1\times2^3+0\times2^2+1\times2^1+1\times2^0$
$=(27)_{10}$,

$(10.101)_2=1\times2^1+0\times2^0+1\times2^{-1}+0\times2^{-2}+1\times2^{-3}$
$=(2.625)_{10}$.

1.3.2 数的浮点表示

一般地,一个 β 进制数 x 可以表示为

$$x=\pm\beta^J\sum_{k=1}^{t}d_k\beta^{-k},\tag{1.3.1}$$

其中,$d_k(k=1,2,\cdots,t)$是 $0,1,\cdots,\beta-1$ 中的一个数字. 上式也可写成

$$x=\pm a\times\beta^J,\tag{1.3.2}$$

其中,

$$a=\sum_{k=1}^{t}d_k\beta^{-k}=0.d_1d_2\cdots d_t.\tag{1.3.3}$$

我们称 a 为数 x 的**尾数**(其值小于 1);自然数 t 为计算机的**字长**,它表示数 x 的尾数的位数;J 是整数,称为数 x 的**阶**,它用来确定 x 的小数点的位置.

在各种计算机中，有各自规定的字长，以及阶 J 的范围：$-L \leqslant J \leqslant U$（$L$ 和 U 为正整数或零）. L、U 的大小表明计算机中表示的数的范围大小. 式(1.3.1)或(1.3.2)称为数的**浮点表示**. 再假设 $x \neq 0$ 时 $d_1 \neq 0$，则称 x 为**规格化浮点数**.

按式(1.3.1)或(1.3.2)规定的规格化浮点数的全体组成的集合记为 F. 我们称 F 为一个**规格化浮点数系**. 规格化浮点数系 F 是一个有限的离散的数集合. 在数值计算中通常取十进制数.

1.3.3 舍入误差

在使用计算机进行数值计算时，若提供给计算机的数 x 的绝对值大于 F 中最大正数，则在计算机上出现**上溢**；若 $x(\neq 0)$ 的绝对值小于 F 中最小正数，则在计算机上出现**下溢**，此时计算机将数 x 作为 0 处理，称其为机器 0. 上溢和下溢统称为**溢出**. 在以后的讨论中，我们将假定所提供的初始数据或中间结果数据不会发生溢出现象.

初始数据或中间计算结果数据 x 可能不在规格化浮点数系 F 中，因此要用 F 中最接近 x 的数 $\bar{x}$ 作为 x 的近似值. 现设十进制数 $x \notin F$，则可按四舍五入原则取 $\bar{x}$，即若

$$x = \pm a \times 10^J, \tag{1.3.4}$$

其中

$$a = 0.d_1 \cdots d_t d_{t+1} \cdots d_n \cdots, \quad 0 \leqslant d_i \leqslant 9 (i = 1, \cdots, n, \cdots), d_1 > 0, \tag{1.3.5}$$

则取

$$\bar{a} = \begin{cases} 0.d_1 d_2 \cdots d_t, & 0 \leqslant d_{t+1} \leqslant 4, \\ 0.d_1 d_2 \cdots d_t + 10^{-t}, & d_{t+1} \geqslant 5, \end{cases} \tag{1.3.6}$$

$$\bar{x} = \pm \bar{a} \times 10^J. \tag{1.3.7}$$

于是有

$$\left| \frac{\bar{x} - x}{x} \right| = \left| \frac{\pm \bar{a} \times 10^J - (\pm a) \times 10^J}{\pm a \times 10^J} \right| = \left| \frac{\bar{a} - a}{a} \right|$$

而 $a \geqslant 10^{-1}$，$|\bar{a} - a| \leqslant \frac{1}{2} \times 10^{-t}$，因此

$$\left| \frac{\bar{a} - a}{a} \right| \leqslant \frac{1}{2} \times 10^{-t+1} = 5 \times 10^{-t},$$

即有

$$\left| \frac{\bar{x} - x}{x} \right| \leqslant 5 \times 10^{-t}.$$

记 $\varepsilon = \frac{\bar{x} - x}{x}$，则得到下面关系式

$$\bar{x} = x(1 + \varepsilon), \quad |\varepsilon| \leqslant 5 \times 10^{-t}. \tag{1.3.8}$$

说明 有的计算机采用的是只“舍”不“入”的断位原则，即取

$$\bar{a} = 0.d_1 d_2 \cdots d_t,$$

此时

$$\bar{x} = x(1 + \varepsilon), |\varepsilon| \leqslant 10^{-t+1}. \tag{1.3.9}$$

1.4　数值计算中的若干原则

在数值计算中，几乎每一步计算都会产生误差，但是每步都作误差分析是不可能的，也是不科学的. 这是因为，误差积累有正有负，绝对值有大有小，如果都按最坏情况估计误差限，则得到的结果比实际误差大得多. 这种保守的误差估计不反映实际误差积累.

由于对误差积累问题进行定量分析较为困难，因此，人们通常是进行定性分析. 下面给出数值计算中应注意的一些原则，它们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.

1.4.1　避免两个相近的数相减

由例 1.2.3 可知在数值计算中，两个相近的数直接相减会造成有效数字严重损失，因此我们必须尽量避免此类运算.

例如，当 x_1, x_2 很接近时，

$$\lg x_1 - \lg x_2 = \lg \frac{x_1}{x_2},$$

用右端算式代替左端，有效数字就不会损失.

当 x 很大时，

$$\sqrt{x+1} - \sqrt{x} = \frac{1}{\sqrt{x+1} + \sqrt{x}},$$

用右端算式代替左端，从而避免有效数字损失.

一般情况下，当 $f(x^*)$ 和 $f(x)$ 很接近，但又需要作 $f(x) - f(x^*)$ 运算时，为避免有效数字的损失，可用 Taylor 展开式

$$f(x^*) - f(x) = f'(x)(x^* - x) + \frac{1}{2} f''(x)(x^* - x)^2 + \cdots,$$

取右端的有限项近似左端.

如果计算公式不能改变，则可采用增加有效数字位数的方法.

1.4.2　防止大数“吃掉”小数

由于计算机的位数有限，因此，在计算机上进行加减法运算时要对阶和规格化. 对阶是以大数为基准，小数向大数对齐，即比较相减两个数的阶，将阶小的尾数向右移，每移一位阶码加 1，直到小数的阶与大数的阶一致时为止，并将移位后的尾数多于字长的部分进行四舍五入. 然后将对阶后的两数相加减，最后将结果化为规格化形式. 当参加运算的两个数的数量级相差很大时，若不注意运算次序，就有可能出现数量级大的数把数量级小的数“吃掉”的现象，从而影响计算结果的可靠性.

例如，在四位十进制计算机上计算

$$0.7315 \times 10^3 + 0.4506 \times 10^{-5},$$

对阶后得 $0.7315 \times 10^3 + 0.0000 \times 10^3$，运算结果为 0.7315×10^3，此时大数“吃掉”了小数. 又如

$$0.8153 + 0.6303 \times 10^3,$$

对阶后得 $0.0008\times10^3+0.6303\times10^3$，运算结果为 0.6311×10^3. 此时大数“吃掉”了部分小数.

再如，已知 $x=-0.5675\times10^2$，$y=0.4812\times10^{-3}$，$z=0.5679\times10^2$，若按 $(x+y)+z$ 进行计算，则结果为 0.4000×10^{-1}；若按 $(x+z)+y$ 进行计算，则结果为 0.4048×10^{-1}，可见 $(x+y)+z\neq(x+z)+y$，且 $(x+z)+y$ 的结果较为接近准确值 0.404812×10^{-1}，这是因为在 $(x+y)+z$ 运算中出现了绝对值大的数 x“吃掉”小数 y 的现象.

例 1.4.1 求一元二次方程

$$x^2-(10^6+1)x+10^6=0$$

的根. 利用因式分解可知方程的两个根为 $x_1=10^6$，$x_2=1$.

若用六位十进制计算机进行编程计算，求根公式为

$$x_{1,2}=\frac{-b\pm\sqrt{b^2-4ac}}{2a},$$

其中，$-b=10^6+1=0.1\times10^7+0.0000001\times10^7$. 由于该计算机只能保留小数点后 6 位，所以 0.0000001×10^7 在计算中将会当作 0.000000×10^7 处理(即不起作用)，于是 $-b=0.1\times10^7=10^6$. 类似地，有

$$b^2-4ac\approx b^2,\sqrt{b^2-4ac}\approx|b|,$$

故求得的两根为 $\bar{x}_1=10^6$，$\bar{x}_2=0$.

出现以上结果的原因是：计算机在计算时大数“吃掉”小数. 为避免上述现象的发生，可将计算公式做适当处理，如取

$$x_1=\frac{-b-\mathrm{sign}(b)\sqrt{b^2-4ac}}{2a},$$

在计算另一根 x_2 时利用关系式 $x_1x_2=\frac{c}{a}$，得

$$x_2=\frac{c}{ax_1},$$

这时可求得 $x_1=10^6$，$x_2=1$.

在实际计算时，为避免大数“吃掉”小数，一定要注意安排计算次序，使计算始终在数量级相差不大的数之间进行.

1.4.3 避免绝对值太小的数作分母

用绝对值太小的数作除数进行除法运算时会使舍入误差增大，如计算 $\frac{x}{y}$，若 $0<|y|\ll|x|$，则可能使误差很大，甚至出现计算机上溢现象，导致计算无法进行下去.

例 1.4.2 考虑二元线性方程组

$$\begin{cases}0.0001x_1+x_2=1,\\ x_1+x_2=2,\end{cases}$$

其准确解为 $x_1=\frac{10000}{9999}$，$x_2=\frac{9998}{9999}$.

若在三位十进制数字的浮点系统中用消去法求解，方程组改写成