21世纪应用型机电规划教材

数值计算方法

主　编　颜兵兵
副主编　史立秋　张金波　冯　浩
主　审　马浏轩

北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

21 世纪应用型机电规划教材

数值计算方法

主　编　颜兵兵
副主编　史立秋　张金波　冯　浩
主　审　马浏轩

北京航空航天大学出版社

内 容 简 介

本书主要介绍数值计算方法的基本理论和 MATLAB 的应用。着重介绍数值方法的构造、使用范围以及应用时的计算效果、稳定性、收敛性等问题。既注重理论的严谨性，又注重方法的实用性。每章配备 MATLAB 应用实例，并附有习题，以帮助读者巩固和加深理解有关内容。

本书可作为理工科本科生、研究生“数值计算方法”课程的教材或参考书，也可作为科技人员使用数值计算方法和 MATLAB 的参考手册。

图书在版编目(CIP)数据

数值计算方法 / 颜兵兵主编. -- 北京 : 北京航空航天大学出版社，2012.9

ISBN 978-7-5124-0862-3

Ⅰ. ①数… Ⅱ. ①颜… Ⅲ. ①数值计算－计算方法 Ⅳ. ①O241

中国版本图书馆 CIP 数据核字(2012)第 152370 号

数值计算方法

主　编　颜兵兵

副主编　史立秋　张金波　冯　浩

主　审　马浏轩

责任编辑　刘　晨

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191)　http://www.buaapress.com.cn

发行部电话：(010)82317024　传真：(010)82328026

读者信箱：emsbook@gmail.com　邮购电话：(010)82316936

涿州市新华印刷有限公司印装　各地书店经销

*

开本：710×1 000　1/16　印张：13.75　字数：301 千字

2012 年 9 月第 1 版　2012 年 9 月第 1 次印刷　印数：3 000 册

ISBN 978-7-5124-0862-3　定价：29.00 元

本书编委会

主　编： 颜兵兵

副主编： 史立秋　张金波　冯　浩

主　审： 马浏轩

参　编： 李彩花　李亚芹　胡晓平　曹慧琦　陈　艳

丁　俊　方　强　何丽萍　李丹丹　陆毓明

裘建国　宋　琼　孙维君　唐清清　王莉莉

吴蕾颖　夏　娴　谢燕萍　徐晓峰　许颖寅

岳齐中　俞侃侃　朱　燕

前　言

数值计算方法是一种研究数学问题的近似解的方法，简称计算方法。它是一门不充分精确的学科，随着计算机科学的发展与普及，数值计算方法已成为许多工程及科学研究中普遍采用的工具。这门课程是大学本科及研究生普遍开设的一门必修课程。本教材根据目前普通高等院校专业教学的基本情况，结合应用型人才培养目标和专业教育教学需要，在总结近几年教学实践和同类教材编写经验的基础上编写而成。

本书可作为理工科本科生、研究生“数值计算方法”课程的教材或参考书，也可作为科技人员使用数值计算方法和 MATLAB 的参考手册。

本书共分 8 章，由颜兵兵任主编和统稿，史立秋、张金波、冯浩任副主编。参加本书编写工作的有：佳木斯大学颜兵兵（参编约 6 万字）、史立秋（参编约 10 万字）、张金波（参编约 5 万字）、佳木斯大学李彩花、李亚芹和胡晓平（分别参编约 2 万字）、佳木斯第十一中学冯浩（参编约 2 万字）。本书由佳木斯大学马浏轩教授主审。同时感谢李丹丹、宋琼、谢燕萍、裘建国、徐晓峰、何丽萍、丁俊、陆毓明、夏娴、吴蕾颖、曹慧琦、岳齐中、许颖寅、方强、王莉莉、陈艳、朱燕、孙维君、唐清清、俞侃侃等在教材编校过程中所做的工作。

由于编者水平有限，书中难免存在疏漏和不妥之处，恳请广大读者批评指正。

编　者

2012 年 5 月

目 录

第 1 章 绪 论 …… 1
1.1 数值计算方法及其主要内容 …… 1
1.2 误差的来源 …… 4
1.3 绝对误差、相对误差及有效数字 …… 5
1.3.1 绝对误差 …… 5
1.3.2 相对误差 …… 6
1.3.3 有效数字 …… 6
1.4 数值计算中误差的传播 …… 8
1.4.1 基本运算中的误差估计 …… 8
1.4.2 算法的数值稳定性 …… 10
1.5 数值计算中应注意的问题 …… 12
1.6 习 题 …… 15
第 2 章 MATLAB 数学软件简介 …… 16
2.1 MATLAB 的运行环境、安装及运行 …… 16
2.1.1 MATLAB 的运行环境 …… 16
2.1.2 MATLAB 的安装 …… 17
2.1.3 MATLAB 的运行及退出 …… 17
2.1.4 MATLAB 的联机帮助 …… 18
2.2 MATLAB 的基本功能 …… 18
2.2.1 MATLAB 中的数字、变量及其运算 …… 19
2.2.2 MATLAB 中矩阵的输入、生成及标志 …… 21
2.2.3 MATLAB 中矩阵的运算 …… 22
2.2.4 MATLAB 中矩阵的关系运算 …… 23
2.3 绘图及图像处理 …… 24
2.3.1 Plot 函数绘图 …… 24
2.3.2 常用绘图命令 …… 25
2.3.3 MATLAB 中的三维绘图 …… 25
2.4 MATLAB 中的程序结构及 M 文件 …… 26
2.4.1 顺序结构 …… 26

2.4.2　分支结构…… 26
2.4.3　循环结构…… 27
2.5　习　题…… 29
第 3 章　非线性方程的解法 …… 30
3.1　二分法…… 31
3.2　简单迭代法…… 33
3.3　牛顿(Newton)迭代法 …… 38
3.4　牛顿迭代法的变形…… 41
3.4.1　简化的牛顿迭代法…… 41
3.4.2　弦截法…… 43
3.4.3　牛顿下山法…… 45
3.5　MATLAB 应用实例 …… 47
3.6　习　题…… 48
第 4 章　线性方程组的解法 …… 50
4.1　向量范数和矩阵范数…… 50
4.1.1　向量的范数…… 50
4.1.2　矩阵的范数…… 51
4.1.3　误差分析…… 54
4.2　迭代法…… 58
4.2.1　雅克比(Jacobi)迭代法 …… 59
4.2.2　高斯-赛德尔(Gauss－Seidel)迭代法 …… 61
4.2.3　迭代法的收敛性…… 63
4.3　高斯(Gauss)消去法 …… 67
4.4　高斯(Gauss)列主元消去法 …… 70
4.5　三角分解法…… 73
4.6　MATLAB 应用实例 …… 76
4.7　习　题…… 78
第 5 章　插值法与最小二乘法 …… 82
5.1　插值法概述…… 82
5.1.1　插值问题…… 82
5.1.2　插值多项式的存在唯一性…… 83
5.2　拉格朗日(Lagrange)插值法…… 85
5.2.1　Lagrange 插值多项式 …… 85
5.2.2　高次插值多项式的问题…… 91
5.3　分段插值法…… 92
5.3.1　分段线性 Lagrange 插值 …… 92

5.3.2 分段二次 Lagrange 插值 …… 93
5.4 牛顿(Newton)插值法 …… 95
5.4.1 均 差 …… 95
5.4.2 Newton 插值公式及其余项 …… 97
5.4.3 差 分 …… 99
5.4.4 等距节点的插值公式 …… 100
5.5 埃尔米特(Hermite)插值 …… 103
5.6 样条函数与样条插值 …… 108
5.6.1 基本概念 …… 109
5.6.2 三弯矩插值法 …… 111
5.6.3 三转角插值法 …… 114
5.7 数据拟合的最小二乘法 …… 117
5.7.1 法方程组 …… 118
5.7.2 利用正交多项式作最小二乘拟合 …… 124
5.8 MATLAB 应用实例 …… 129
5.9 习 题 …… 131
第 6 章 数值微分与积分 …… 134
6.1 数值微分 …… 134
6.1.1 差商公式 …… 134
6.1.2 中点方法的加速 …… 136
6.1.3 插值型的求导公式 …… 137
6.2 牛顿-柯特斯(Newton - Cotes)求积公式 …… 138
6.2.1 插值型求积公式及 Cotes 系数 …… 138
6.2.2 低阶 Newton - Cotes 公式的余项 …… 141
6.2.3 Newton - Cotes 公式的稳定性 …… 143
6.3 复合求积法 …… 144
6.3.1 复合求积公式 …… 144
6.3.2 复合求积公式的余项及收敛的阶 …… 145
6.3.3 步长的自动选择 …… 146
6.4 龙贝格(Romberg)求积法 …… 148
6.4.1 梯形法的递推化 …… 148
6.4.2 龙贝格求积法 …… 148
6.4.3 龙贝格算法的收敛性 …… 151
6.5 高斯(Gauss)求积公式 …… 152
6.5.1 几种高斯型求积公式 …… 154
6.5.2 高斯型求积公式的稳定性和收敛性 …… 157

6.6 MATLAB 应用实例 …… 158
6.7 习 题 …… 159
第 7 章 常微分方程的数值解法 …… 161
7.1 欧拉(Euler)法 …… 163
7.1.1 Euler 方法公式 …… 163
7.1.2 Euler 方法的误差估计 …… 164
7.2 改进的欧拉(Euler)法 …… 166
7.2.1 梯形公式 …… 166
7.2.2 改进 Euler 法 …… 167
7.3 龙格-库塔(Runge - Kutta)法 …… 168
7.3.1 Runge - Kutta 法的基本思想 …… 168
7.3.2 Runge - Kutta 法的构造 …… 169
7.3.3 变步长的 Runge - Kutta 法 …… 172
7.4 线性多步法 …… 173
7.4.1 线性多步公式的导出 …… 173
7.4.2 常用的线性多步公式 …… 175
7.4.3 预测-校正系统 …… 179
7.5 MATLAB 应用实例 …… 181
7.6 习 题 …… 183
第 8 章 矩阵特征值和特征向量的计算 …… 185
8.1 乘幂法与反乘幂法求特征值 …… 185
8.1.1 乘幂法 …… 185
8.1.2 加速技术 …… 188
8.1.3 反幂法 …… 191
8.2 对称矩阵的雅克比方法 …… 193
8.2.1 旋转变换 …… 194
8.2.2 雅克比方法 …… 195
8.3 QR 法 …… 198
8.3.1 豪斯荷尔德阵 …… 198
8.3.2 QR 分解 …… 199
8.3.3 QR 方法 …… 200
8.3.4 原点位移的 QR 方法 …… 201
8.4 MATLAB 应用实例 …… 202
8.5 习 题 …… 205
参考文献 …… 207

第1章 绪 论

主要内容：本章主要介绍数值计算方法的研究对象和特点、绝对误差、相对误差、误差的传播以及在数值计算过程中应注意的问题。要求学生了解数值计算方法的特点，理解绝对误差、相对误差的概念及其对数值计算的影响。

1.1 数值计算方法及其主要内容

数学是研究数与形的科学，而计算数学是数学的一个分支，通常也称为数值分析或数值计算方法，它是研究如何利用计算工具（如计算器、计算机等）求出数学问题的数值解答（如数据、表格图形等）的学问。由此可见，计算数学的前身可追溯到人类文明萌芽时期的算学和测绘学，是数学中最古老的一部分。但是，由于计算工具的笨拙和数值计算的复杂，长期制约了计算数学的发展。随着20世纪40年代电子计算机的出现以及现代科学与工程中大规模科学计算的迫切需求，计算数学获得了前所未有的发展，半个多世纪以来，计算数学已经发展成为现代意义下的计算科学，成为继牛顿-伽利略（Isaac Newton，英国，1642—1727；Galileo Galilei，意大利，1564—1642）理论研究和科学实验两大科学方法之后的第三大科学方法，并深入到各个学科领域的方方面面，扮演着越来越重要的角色。

许多科学计算来源于科学与工程，其目的在于理解一切自然现象、解决工程实际问题或进行最优设计等。在计算科学中，通过计算机重造或模拟物理系统或过程，即计算仿真，能够大大增强人们对物理系统或过程的认知力，尤其是那些通过理论、观测、实验等手段难以观察到的系统或过程。例如，在天文学中，由于过程的复杂性，很难用理论描述两个碰撞黑洞的具体形态，也不可能直接观察到，更不可能在实验室中进行重造，而通过计算机仿真可达到我们的目的。它所需要的只是一个适当的数学模型（如广义相对论中的Einstein方程组）、求解该数学模型的数值计算方法和用来实现相应数值计算方法的计算速度足够快且内存足够大的计算机。由此可见，计算

数学不同于数学学科的其他分支，它不能单独存在而必须依托现代的计算机手段并随着计算机工具的发展而发展。同时，它也区别于计算机科学的各个分支，这是因为它所涉及的对象不仅有离散变量，而且有更多的连续变量，如时间、位移、速度等。

一个计算过程主要包括如下几个环节：

(1) 建立一个数学模型，即数学建模。

(2) 设计求解数学模型的算法，即算法设计。

(3) 设计计算机上实现算法的软件。

(4) 上机运行，数值模拟物理过程。

(5) 计算结果再表示，如图像的可视化等。

(6) 分析计算结果的可靠性，必要时重复上述过程。

本书将针对来源于科学与工程中的数学模型问题，介绍计算机上常用的数值计算方法的算法设计思想并进行算法分析。具体包括非线性方程求根、线性代数方程组求解、数值微积分和常微分方程数值解等内容。

设计一个算法的第一步是将问题可算化。一般地，可通过如下手段将问题可算化。

(1) 用有限维空间代替无限维空间，如多项式逼近连续函数等。

(2) 用有限过程代替无限过程，如积分和无穷级数用有限项和代替，导数用差分代替等。

(3) 用代数方程代替微分方程，线性问题代替非线性问题，低阶系统代替高阶系统，简单函数代替复杂函数等。换言之，用简单问题代替复杂问题。

例如，在求解一个非线性微分方程组时，首先将问题化为非线性代数系统，即非线性方程组，然后再用一系列线性代数方程组逼近该非线性方程组。而在求解线性代数方程组时，通常可将问题简化为易于求解的具有特殊系数矩阵(如三角形矩阵)的线性代数方程组。

在上述每个环节中，我们不仅希望问题得到简化，而且需要验证转化后的替代问题的解是否和原问题的解保持一致或者在容许的误差范围之内，也即说，我们还需做到替代问题的解和原问题的解在某种意义上保持一致。

从理论的角度上看，人们自然希望利用等解变换将问题可算化。而在实际上，这种等解变换往往不可能，这意味着在原问题和变换后的替代问题的解之间存在着扰动，因而需要我们对解的扰动进行分析，即估计逼近解(变换后的替代问题的解)与精确解(原数学模型的解)之间的误差或精度。

【例 1.1】 计算 $\sin x$ 的值，其中已知角 x 的弧度为 $0\sim\pi$。

解： 电子计算机实质上只不过是一个减法器，即只会做加减乘除等基本运算。因此，需将问题可算化，根据微分学的泰勒(Brook Taylor，英国，1685—1731)公式，函数值 $\sin x$ 的计算问题可化为无穷级数和问题。

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + R_{2n+1}(x) \quad (1-1)$$

这是一个等解变换。变换后的级数求和问题包含有无穷多次的加减乘除运算。由于计算机只能进行有限运算，故用有限和代替无穷级数，即

$$\sin x \approx x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} = P_{2n+1}(x) \quad (1-2)$$

多项式 $P_{2n+1}(x)$ 仅包含有限次的加减乘除运算，可以上机计算。当 n 充分大时，余项（通常称为截断误差或公式误差）$R_{2n+1}(x) = \sin x - P_{2n+1}(x)$ 的绝对值会很小。故当 n 充分大时，可用多项式 $P_{2n+1}(x)$ 的值近似函数 $\sin x$ 的值，即三角函数 $\sin x$ 的计算问题转化为计算机可计算的多项式计算问题。式(1-1)通常称为泰勒多项式的逼近。

【例 1.2】 计算正实数 c 的开平方值 $\sqrt{c}$。

解： 下面介绍求解此问题的一种迭代算法，即从给定初值出发，通过某种方式，产生一个逐步逼近 $\sqrt{c}$ 的序列的方法。这种迭代方法早在 4000 多年以前就已被古巴比伦人所知。设给定初值 $x_0 > 0$。显然，根值 $\sqrt{c}$ 在正数 x_0 和 $\frac{c}{x_0}$ 更接近根值 $\sqrt{c}$。如此可得到点列 $x_0, x_1, \cdots, x_k, \cdots$（称为迭代序列），其计算公式（称为迭代格式）为

$$x_{k+1} = \frac{1}{2}\left(x_k + \frac{c}{x_k}\right) \quad (1-3)$$

如果 $\{x_k\}$ 收敛，即 $\lim\limits_{k\to\infty} x_k = x^*$ 存在，则在式(1-3)中令 k 趋于无穷，得

$$x^* = \frac{1}{2}\left(x^* + \frac{c}{x^*}\right)$$

即 $x^* = \sqrt{c}$，故当 k 充分大时，可用 x_k 的值近似根值 $\sqrt{c}$，即 $\sqrt{c}$ 的计算问题转化为有限次迭代计算 $x_0, x_1, \cdots, x_k$。

定义 1.1 通过已知点 x_0，按照某种规则产生一系列点 $x_0, x_1, \cdots, x_k, \cdots$ 的方法称为迭代法，而点列 $x_0, x_1, \cdots, x_k, \cdots$ 称为迭代序列。

对于迭代法，通常需要考虑迭代序列的收敛性和收敛效率。式(1-3)又称为求解线性方程 $x^2 - c = 0$ 的牛顿迭代。牛顿迭代是非线性方程求根的一种非常有效的数值迭代算法。

一个好的数值计算方法应具备适用范围广、运算量少、储存单元省、逻辑结构简单且计算结果可靠等特点。

【例 1.3】 计算多项式 $p(x) = 5 + 6x + 7x^2 + 8x^3 + 9x^4$ 并分析计算量。

解： 如果直接计算各项并累加需要 10 次乘除法和 4 次加减运算。但若改用下式计算

$$p(x) = 5 + x(6 + x(7 + x(8 + 9x)))$$

则仅需 4 次乘除法和 4 次加减法运算。

对于一般多项式计算，通常采用第二种方法。这种方法称为秦九韶（中国，1202—1261）算法或 Horner 算法，设多项式

$$P(x)=a_0+a_1x+a_2x^2+\cdots+a_nx^n$$

如果直接计算，需要 $\frac{n(n+1)}{2}$ 次乘除法和 n 次加减法运算。若采用秦九韶算法，即

$$\begin{cases} b_0=a_n, \\ b_k=a_{n-k}+b_{k-1}x\ ,k=1,2\cdots,n \\ P(x)=b_n, \end{cases}$$

则仅需要 n 次乘除法和 n 次加减法运算。因此，当多项式的次数很高时，秦九韶算法有明显的优越性。

对于小型问题，计算量的减少或存储单元的节省似乎不太重要，但对于大规模问题，有时却有着决定性的意义。

【例 1.4】 计算行列式 $D_n=\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}$ 并分析计算量。

解： 如果按照行列式的定义

$$D_n=\sum_{i_1i_2\cdots i_n}(-1)^{\tau(i_1i_2\cdots i_n)}a_{i_11}a_{i_22}\cdots a_{i_nn}$$

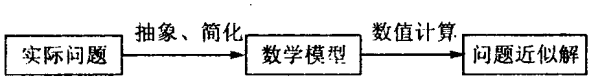
计算，需计算 $n!$ 项 n 个因子的乘积，故需 $M_n=n!(n-1)$ 次乘除法运算。当 $n=20$ 时，$M_{20}\approx 4.6\times 10^{19}$。做这么多的乘除法运算，即使在每秒做万亿次乘除法的计算机上也需要运行一年之久。因此，在计算行列式时，人们通常采用通过行或列变换将行列式化为上三角行行列式或下三角形行列式的途径计算。这样的计算方法仅需做

$$\overline{M_n}=(n-1)^2+(n-2)^2+\cdots+1^2+(n-1)=\frac{n(n-1)(2n-1)}{6}+(n-1)$$

次乘除法运算。当 $n=20$ 时，$\overline{M_{20}}\approx 2489$。此计算量较前者要少得多。

1.2 误差的来源

数值计算方法是研究数学问题近似解的方法和过程。因此，在计算过程中，误差是不可避免的。用数学方法解决实际问题，常按以下过程进行：



在此过程中，引起误差的因素很多，主要有以下几种：

(1) 模型误差。实际问题的解与数学模型的解之差称为模型误差。

(2) 观测误差。数学问题中所出现的一些参量,其值往往由观测得到,而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。一般数学问题常常难以求出精确解,需要简化为较易求解的问题,以简化问题的解作为原问题解的近似。如求一个收敛的无穷级数之和,总是用它的部分和作为近似值,也就是截去该级数后面的无穷多项。这样由于简化问题所引起的误差称为方法误差或截断误差。例如:

$$\cos x = 1 - \frac{x^2}{2} + \frac{x^4}{4} - \frac{x^6}{6} + \cdots + \frac{(-1)^n x^{2n}}{(2n)!} + \cdots$$

当 $|x|$ 很小时,可以用 $1 - \frac{x^2}{2}$ 作为 $\cos x$ 的近似值。由交错级数判断的莱布尼兹(Leibniz)准则,它的截断误差的绝对值不超过 $\frac{x^4}{24}$。

(4) 舍入误差。在计算过程中往往要对数字进行舍入。如受机器字长的限制,无穷小数和位数很多的数必须舍入成一定的位数。这样产生的误差称为舍入误差。

本书只讨论截断误差和舍入误差对计算结果的影响。

1.3 绝对误差、相对误差及有效数字

1.3.1 绝对误差

定义 1.2 设 x^* 为准确值 x 的一个近似值,则

$$e(x^*) = x - x^* \tag{1-4}$$

称为近似值 x^* 的绝对误差,简称误差。

一般情况下准确值 x 难以求出,从而也不能算出绝对误差 $e(x^*)$ 的准确值,但可以依据测量工具或计算的情况估计出它的取值范围,即估计出误差绝对值的一个上界 ε。

$$|e(x^*)| = |x - x^*| \leqslant \varepsilon \tag{1-5}$$

通常称 ε 为近似值 x^* 的绝对误差限,简称误差限。显然,误差限不是唯一的。有了误差限及近似值,就可以得到准确值的范围。

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

即准确值 x 必定在区间 $[x^* - \varepsilon, x^* + \varepsilon]$ 内,也常记作

$$x = x^* \pm \varepsilon$$

容易看出,经过四舍五入得到的数,其误差必定不超过被保留的最后数位上的半个单位,即最后数位上的半个单位为其误差限。例如若取 π 的近似值为 3.14,则

$$|\pi - 3.14| \leqslant 0.0016 \leqslant \frac{1}{2} \times 10^{-2}$$

若取 $\pi \approx 3.142$，则

$$|\pi - 3.142| \leqslant 0.00041 \leqslant \frac{1}{2} \times 10^{-3}$$

误差限的大小不能完全反映近似值的精确程序。要刻画近似值的精确程序，不仅要看绝对误差的大小，还必须考虑所测量本身的大小，由此引出了相对误差的概念。

1.3.2 相对误差

定义 1.3 设 x^* 为精确值 x 的近似值，绝对误差与准确值之比称为近似值 x^* 的相对误差，记为 $e_r(x^*)$，即

$$e_r(x^*) = \frac{e(x^*)}{x} = \frac{x - x^*}{x} \tag{1-6}$$

由于在计算过程中准确值 x 总是未知的，故一般取相对误差为

$$e_r(x^*) = \frac{e(x^*)}{x^*}$$

可以证明，当 $|e_r(x^*)|$ 很小时，$\frac{e(x^*)}{x} - \frac{e(x^*)}{x^*}$ 是 $e_r(x^*)$ 的高阶无穷小，可以忽略不计。所以，取绝对误差与近似值之比为相对误差是合理的。

同样，相对误差也只能估计其上限。如果存在正数 ε_r，使得

$$|e_r(x^*)| = \left|\frac{e(x^*)}{x^*}\right| \leqslant \varepsilon_r \tag{1-7}$$

则称 ε_r 为 x^* 的相对误差限。显然，误差限与近似值绝对值之比 $\frac{\varepsilon}{|x^*|}$ 为 x^* 的一个相对误差限。

例如，由实验测得光速近似值为 $c^* = 2.997925 \times 10^5$ km/s，其误差限为 0.1 km/s，于是

$$\frac{\varepsilon}{|c^*|} = \frac{0.1}{2.997925 \times 10^5} < 4 \times 10^{-7}$$

所以 4×10^{-7} 是 c^* 的一个相对误差限。

1.3.3 有效数字

有效数字是近似值的一种表示法。它既能表示近似值的大小，又能表示其精确程度。在计算过程中，常常按四舍五入的原则取数 x 的前几位数 x^* 为其近似值。例如 $x = \sqrt{2} = 1.414213562\cdots$ 取前四位数得近似 $x^* = 1.414$，取前八位数得近似

值 $x^*=1.4142136$，前面已经提到。通过四舍五入得到的数，其绝对误差均不超过末位数字的半个单位，即

$$|\sqrt{2}-1.414|\leqslant\frac{1}{2}\times10^{-3}$$

$$|\sqrt{2}-1.4142136|\leqslant\frac{1}{2}\times10^{-7}$$

如果近似值 x^* 的误差限是 $\frac{1}{2}\times10^{-n}$，则称 x^* 准确到小数点后第 n 位，并从第一个非零数字到这一位的所有数字均称为有效数字。例如 $\sqrt{2}$ 的近似值 1.414 准确到小数点后第 3 位，它具有 4 位有效数字。1.414 2136 作为 $\sqrt{2}$ 的近似值精确到小数点第 7 位，有 8 位有效数字。一般地，如果近似值 x^* 的规格化形式为

$$x^*=\pm0.a_1a_2\cdots a_n\cdots\times10^m \tag{1-8}$$

其中 m 为整数，$a_1\neq0$，$a_i(i=1,2,\cdots)$ 为 0～9 的整数。如果

$$|x-x^*|\leqslant\frac{1}{2}\times10^{m-n} \tag{1-9}$$

则称近似值 x^* 有 n 位有效数字。

例如 $x=0.003\ 400\pm\frac{1}{2}\times10^{-5}$ 表示近似值 0.003 400 准确到小数点后第 5 位，有 3 位又有效数字。

上面的讨论表明，可以用有效数字位数来刻画误差限。形如式(1－8)的数，当 m 一定时，其有效数字位数 n 越大，则误差限越小。例如若 $x^*=1452.046$ 是具有 7 位有效数字的近似值，则它的误差限为

$$|x-x^*|\leqslant\frac{1}{2}\times10^{-3}$$

又若 $x^*=1452.0$ 是具有 5 位有效数字的近似值，则其误差限为 $\frac{1}{2}\times10^{-1}$。

下面的定理给出了相对误差限与有效数字的关系。

定理 1.1 若 x 的近似值 $x^*=\pm0.a_1a_2\cdots a_n\times10^m(a_1\neq0)$ 有 n 位有效数字，则 $\frac{1}{2a_1}\times10^{-n+1}$ 为其相对误差限。反之，若 x^* 的相对误差限 ε_r 满足

$$\varepsilon_r\leqslant\frac{1}{2(a_1+1)}\times10^{-n+1}$$

则 x^* 至少具有 n 位有效数字。

【证明】 由式(1－9)

$$|e(x^*)|=|x-x^*|\leqslant\frac{1}{2}\times10^{m-n}$$

从而有

$$|e_r(x^*)| = \left|\frac{e(x^*)}{x^*}\right| \leqslant \frac{\frac{1}{2}\times 10^{m-n}}{0.a_1a_2\cdots a_n \times 10^m} \leqslant \frac{1}{2a_1}\times 10^{-n+1}$$

所以 $\frac{1}{2a_1}\times 10^{-n+1}$ 是 x^* 的相对误差限。

若 $\varepsilon_r \leqslant \frac{1}{2(a_1+1)}\times 10^{-n+1}$，由式(1-7)

$$|e(x^*)| = |x^* e_r(x^*)| \leqslant 0.a_1\cdots a_n\cdots \times 10^m \varepsilon_r \leqslant$$

$$(a_1+1)\times 10^{m-1}\times \frac{1}{2(a_1+1)}\times 10^{-n+1} = \frac{1}{2}\times 10^{m-n}$$

由式(1-9)，x^* 至少有 n 位有效数字。

定理 1.1 表明，由有效数字位数可以求出相对误差限。如 $x^* = 2.72$ 是 $x = e$ 的具有 3 位有效数字的近似值，故其相对误差限为

$$\varepsilon_r = \frac{1}{2\times 2}\times 10^{-3+1} = 0.25\times 10^{-2}$$

1.4 数值计算中误差的传播

1.4.1 基本运算中的误差估计

本节中所讨论的基本运算是指四则运算与一些常用函数的计算。

由微分学，当自变量改变量(误差)很小时，函数的微分作为函数改变量的主要线性部分可以近似函数的改变量，故利用微分运算公式可导出误差运算公式。

设数值计算中求得的解与参量(原始数据) $x_1, x_2, \cdots, x_n$ 有关，记为

$$y = f(x_1, x_2, \cdots, x_n) \tag{1-10}$$

参量的误差必定引起解的误差。设 $x_1, x_2, \cdots, x_n$ 的近似值分别为 $x_1^*, x_2^*, \cdots, x_n^*$，相应的解为

$$y^* = f(x_1^*, x_2^*, \cdots, x_n^*) \tag{1-11}$$

假定 f 在点 $(x_1^*, x_2^*, \cdots, x_n^*)$ 可微，则当数据误差较小时，解的绝对误差为

$$e(y^*) = y - y^* = f(x_1, \cdots, x_n) - f(x_1^*, \cdots, x_n^*) \approx df(x_1^*, \cdots, x_n^*) =$$

$$\sum_{i=1}^{n} \frac{\partial f(x_1^*, \cdots, x_n^*)}{\partial x_i}(x_i - x_i^*) =$$

$$\sum_{i=1}^{n} \frac{\partial f(x_1^*, \cdots, x_n^*)}{\partial x_i} e(x_i^*) \tag{1-12}$$

其相对误差为

$$e_r(y^*) = \frac{e(y^*)}{y^*} \approx d(\ln f) =$$