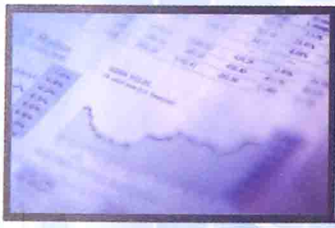
工程硕士系列教材

数值分析

（第二版）



主　　编　王开荣
副主编　杨大地

科学出版社

工程硕士系列教材

数值分析

（第二版）

主　编　王开荣

副主编　杨大地

科学出版社

北京

内 容 简 介

本书系统地介绍数值计算的基本概念、常用算法及有关的理论分析和应用．全书包含数值计算中的基本问题，如线性方程组的数值解法、矩阵特征值和特征向量的数值解法、非线性方程及方程组的数值解法、插值方法、逼近方法、数值积分、数值微分以及常微分方程初值问题的数值解法等，还介绍了 Matlab 软件在数值计算中的应用．读者可将其中的算法和命令应用于数值实验和工程计算实践中去．各章都给出典型例题并配有一定数量的习题，书后给出了习题答案和提示．

本书基本概念叙述清晰，语言通俗易懂，注重算法的实际应用．可作为理工科大学工程硕士研究生的“数值分析”课程教材，还可作为大学本科及硕士生的学习参考书，同时也可供工程技术人员参考使用．

图书在版编目(CIP)数据

数值分析/王开荣主编．--2 版．—北京：科学出版社，2014.5
工程硕士系列教材
ISBN 978-7-03-040625-5

Ⅰ.①数… Ⅱ.①王… Ⅲ.①数值分析-研究生-教材 Ⅳ.①O241

中国版本图书馆 CIP 数据核字(2014)第 099333 号

责任编辑：邓 静 张丽花 / 责任校对：胡小洁
责任印制：阎 磊 / 封面设计：迷底书装

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

文林印务有限公司 印刷
科学出版社发行 各地新华书店经销

*

2006 年 5 月第 一 版　　开本：720×1000 B5
2014 年 5 月第 二 版　　印张：15 1/2
2014 年 5 月第八次印刷　　字数：312 000

定价：45.00 元

（如有印装质量问题，我社负责调换）

前　言

本书是在第一版的基础上修订再版的，第一版已进行了 7 次印刷．原书在 2006 年 5 月出版至今，重庆大学工程硕士研究生课程已使用了八届，且重庆大学 2006～2009 级学术硕士研究生“数值分析”课程也是使用的该教材，部分兄弟院校也将本书作为教材或主要参考书．作者在充分听取教师和学生使用本教材的宝贵意见后，结合多年来的教学经验，觉得有必要对第一版的部分内容进行修订．

本书第一版的第 1 章～第 4 章、第 10 章由杨大地编写，第 5 章～第 9 章由王开荣编写，全书由杨大地统稿．在此衷心感谢杨大地老师的信任，决定由王开荣教授执笔本书第二版的修订工作．在修订工作中，为了便于理解，对部分内容的叙述进行了适当的简化．本书的其他内容主要变动如下：

为便于更好地理解 1.3.4 节的知识点，增添了例 1.9、例 1.10. 在第 2 章，增添了 Gauss 全主元法求行列式的算法，删去了系数矩阵有误差对方程组解影响的推证．为便于自学理解，将 3.4 节的算法改写为用数学公式和文字说明描述的形式．将 4.1 节、4.2 节的算法进行了改写；增添了定理 4.6 和定理 4.7；删去了原 4.3.2 节．对 5.2.2 节进行了修改．对例 6.5 修改了计算方法. 对第 7 章的内容进行了调整，增添了 7.3.3 节多项式拟合的病态性分析和实用性很强的正交多项式拟合．对 8.5.3 节进行了删减．

感谢使用本书第一版的老师和同学对我们提出的宝贵意见，我们更期望在本书以后的使用中能得到更多的建议，以便改进和完善并得以提升．

目　录

第1章 绪　　论

在科学研究、工程实践和经济管理等工作中，存在大量的科学计算、数据处理等问题．应用计算机解决数值计算问题是工程技术人员应当具备的基本能力．工程实际中的大多数数学模型的完备解(公式解或解析解)很难甚至不能用现有的数学方法求出，需借助于数值分析的方法求出其近似解或者说数值形式的解.

运用数学方法解决科学研究或工程技术问题，一般的途径为：

实际问题→模型设计→算法设计→程序设计→上机计算→问题的解

其中，算法设计是数值分析课程的主要内容．

数值分析是研究基本数学问题的适合计算机求解的数值计算方法，包含了数值代数(线性方程组的解法、矩阵特征值计算等)、非线性方程的解法、数值逼近、数值微分与数值积分、常微分方程的数值解法等．它的基本理论和研究方法建立在数学理论基础之上，研究对象是数学问题，是数学的分支之一；又与计算机科学有密切的关系．我们在考虑算法时，往往要同时考虑计算机的特性，如计算速度、存储量、字长等技术指标，考虑程序设计时的可行性和复杂性．如果具备了一定的计算机基础知识和程序设计方法，学习数值分析的理论和方法就会更深刻、更实际，选择或设计的算法也会更合理、更实用．

1.1 算　　法

解决某类数学问题的数值计算方法称为数值算法，简称算法，它是求解数学问题过程的完整而准确的描述．为使算法能在计算机上实现，必须将一个数学问题分解为有限次的四则运算，即＋、－、×、÷运算．虽然在算法中也引用一些简单的基本函数运算，但这些函数运算实际上在计算机语言编译系统中也事先转化成了有限次的四则运算．

1.1.1 算法的表述形式

(1) 用数学公式和文字说明描述．这种方式是面向人的算法，符合人们的理解习惯，和算法的推证相衔接，易于学习接受，但离上机应用距离较大．

(2) 用框图描述．这种方式描述计算过程流向清楚，易于编制程序，但对初学者有一个习惯过程．此外，框图描述格式不很统一，详略难以掌握．

(3) 算法描述语言．它是表述算法的一种通用的语言，有特定的表述程序和

语句，独立于计算机的硬件和软件系统，但可以很容易地转换某种实用的计算机语言，同时也具有一定的可读性．

(4) 算法程序．即用计算机语言描述的算法，是面向计算机的算法，它们常组装成算法软件包．本书的算法通常都有现成的程序文本和软件可资利用．但从学习算法的角度看，这种描述方式并不有利．

1.1.2 算法常具有的基本特征

算法要准确而全面地描述整个解题过程，它常具有如下特征：

(1) 算法常表现为一个无穷过程的截断．

例 1.1 计算 $\sin x$，$x\in\left[0, \frac{\pi}{4}\right]$.

解 根据 Taylor 公式

$$\sin x=x-\frac{x^3}{3!}+\frac{x^5}{5!}-\frac{x^7}{7!}+\cdots+(-1)^n\frac{x^{2n+1}}{(2n+1)!}+\cdots \tag{1.1}$$

这是一个无穷级数，只能在适当的地方“截断”，使计算量不太大，同时又能满足精度要求．如取 $n=3$，计算 $\sin 0.5\approx 0.5-\frac{0.5^3}{3!}+\frac{0.5^5}{5!}-\frac{0.5^7}{7!}=0.479426$，根据 Taylor 余项公式，它的误差应为

$$R=(-1)^4\frac{\xi^9}{9!},\qquad \xi\in\left[0, \frac{\pi}{4}\right] \tag{1.2}$$

$$|R|\leqslant\frac{(\pi/4)^9}{362880}=3.13\times 10^{-7}$$

计算结果的六位数字都是准确的．

(2) 算法常表现为一个连续过程的离散化．

例 1.2 计算积分值 $I=\int_0^1\frac{1}{1+x}\mathrm{d}x$.

解 如图 1.1 所示，将[0, 1]分为 4 等份，分别计算 4 个小曲边梯形的面积的近似值，然后加起来作为积分的近似值．记被积函数为 $f(x)$，即

$$f(x)=\frac{1}{1+x}$$

取 $h=\frac{1}{4}$，有

$$x_i=ih,\ i=0, 1, 2, 3, 4,\qquad T_i=\frac{f(x_i)+f(x_{i+1})}{2}h$$

所以有

$$I\approx\sum_{i=0}^{3}T_i=0.697024$$

与精确值 0.693147 比较，可知结果不够精确，如需提高精度，可进一步细分区间．

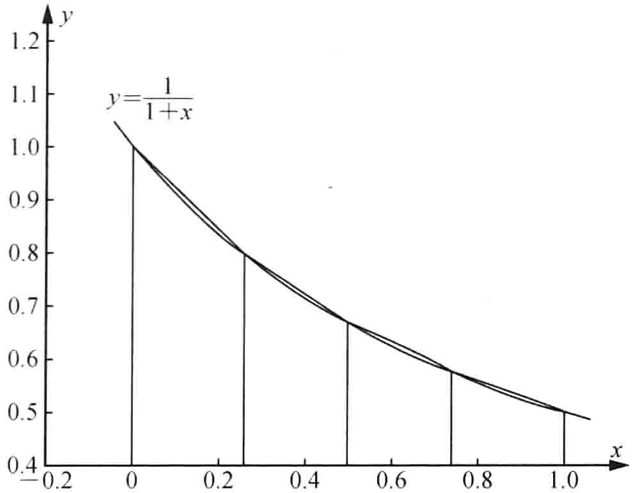


图 1.1

(3) 算法常表现为“迭代”形式.

迭代是指某一简单算法的多次重复，后一次使用前一次的结果. 这种形式易于在计算程序中实现，在程序中表现为“循环”过程.

例 1.3 多项式求值：

$$P_n(x)=a_0+a_1x+a_2x^2+\cdots+a_nx^n \tag{1.3}$$

解 用 t_k 表示 x^k，u_k 表示式(1.3)前 $k+1$ 项之和. 令

$$\begin{cases} t_0=1 \\ u_0=a_0 \end{cases}$$

对 $k=1, 2, \cdots, n$ 反复执行

$$\begin{cases} t_k=xt_{k-1} \\ u_k=u_{k-1}+a_kt_k \end{cases} \tag{1.4}$$

显然 $P_n(x)=u_n$.

对此问题还有一种更好的迭代算法

$$\begin{aligned} P_n(x)&=a_nx^n+a_{n-1}x^{n-1}+\cdots+a_1x+a_0 \\ &=(a_nx^{n-1}+a_{n-1}x^{n-2}+\cdots+a_1)x+a_0 \\ &=((a_nx^{n-2}+a_{n-1}x^{n-3}+\cdots+a_2)x+a_1)x+a_0 \\ &=(\cdots(a_nx+a_{n-1})x+\cdots+a_1)x+a_0 \end{aligned}$$

令

$$\begin{cases} v_0=a_n \\ v_k=xv_{k-1}+a_{n-k} \end{cases} \tag{1.5}$$

显然 $P_n(x)=v_n$.

这两种算法都是将 n 次多项式化为 n 个一次多项式来计算，这种化繁为简的

方法在数值分析中经常使用．算法的计算量如下：

第一种方法：执行 n 次式(1.4)，每次 2 次乘法、1 次加法，共计 $2n$ 次乘法、n 次加法；

第二种方法：执行 n 次式(1.5)，每次 1 次乘法、1 次加法，共计 n 次乘法、n 次加法．

第二种方法运算量少，它是中国宋代数学家秦九韶最先提出的，被称为“秦九韶算法”．

例 1.4 不用开平方计算$\sqrt{a}$($a>0$)的值．

解 假定 x_0 是$\sqrt{a}$的一个近似值，$x_0>0$，则$\frac{a}{x_0}$也是$\sqrt{a}$的一个近似值，且 x_0 和$\frac{a}{x_0}$两个近似值必有一个大于$\sqrt{a}$，另一个小于$\sqrt{a}$，可以设想它们的平均值应为$\sqrt{a}$的更好的近似值，于是有算法

$$x_{k+1}=\frac{1}{2}\left(x_k+\frac{a}{x_k}\right),\quad k=0,1,2,\cdots \tag{1.6}$$

如计算$\sqrt{3}$，取 $x_0=2$，有

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

计算有

$$\begin{aligned}
x_0&=2\\
x_1&=1.75\\
x_2&=1.7321429\\
x_3&=1.7320508\\
&\vdots
\end{aligned}$$

此算法收敛速度很快，只算三次就能得到 8 位精确数字．

迭代法应用时要考虑是否收敛、收敛条件及收敛速度等问题，今后的课程将进一步讨论．

1.2 误　差

数值计算当然是越精确越好，最好是没有误差，但这一般说来是不可能的．误差总是不可避免的，问题是如何估计误差，并将误差控制在可以接受的范围内．因此，在数值分析中误差分析是十分重要的．

1.2.1 误差的来源

在运用数学方法解决实际问题的过程中，每一步都可能带来误差．

$$
\text{误差}\begin{cases}\text{固有误差}\begin{cases}\text{模型误差}\\\text{测量误差}\end{cases}\\\text{计算误差}\begin{cases}\text{截断误差}\\\text{舍入误差}\end{cases}\end{cases}
$$

固有误差是求解工程问题的数学模型本身所具有的误差，是无法避免的．

模型误差：在建立数学模型时，往往要忽视很多次要因素，把模型“简单化”“理想化”，这时模型就与真实背景有了差距，即带入了误差．

测量误差：数学模型中的已知参数，多数是通过测量得到．而测量过程受工具、方法、观察者的主观因素、不可预料的随机干扰等影响必然带入误差．

计算误差是用数值计算方法求得的近似解与精确解之间的误差，它可以通过选择好的数学模型和计算方法来加以控制．数值分析的目标就是选择较好的计算公式，编制较好的算法和程序，使求解工程应用问题的计算误差被控制在最小的范围内．

截断误差：数学模型常难于直接求解，往往要近似替代，简化为易于求解的问题，这种简化带入的误差称为方法误差或截断误差．

舍入误差：计算机只能处理有限数位的小数运算，初始参数或中间结果都必须进行四舍五入运算，这必然产生舍入误差．

在数值分析课程中不分析讨论模型误差；截断误差是数值分析课程的主要讨论对象，它往往是计算中误差的主要部分，在讲到各种算法时，通过数学方法可推导出截断误差限的公式(如式(1.2))；舍入误差的产生往往带有很大的随机性，讨论比较困难，在问题本身呈病态或算法稳定性不好时，它可能成为计算中误差的主要部分；至于测量误差，可以作为初始的舍入误差看待．

详尽的误差分析是困难的，有时是不可能的．在数值分析中主要讨论截断误差及舍入误差．但一个训练有素的计算工作者，当发现计算结果与实际不符时，应当能诊断出误差的来源，并采取相应的措施加以改进，直至建议对模型进行修改.

1.2.2　误差的基本概念

1. 误差与误差限

定义 1.1　设 x^* 是准确值(一般是不知道的)，x 是它的一个近似值，称

$$e=x-x^* \tag{1.7}$$

为近似值 x 的绝对误差，简称误差．

误差一般无法准确计算，只能根据测量或计算情况估计出它的绝对值的一个上限，这个上限称为近似值 x 的误差限，记为 ε. 即

$$|x-x^*|\leqslant\varepsilon \tag{1.8}$$

其意义是$-\varepsilon \leqslant x-x^* \leqslant \varepsilon$. 在工程中常记为$x=x^* \pm \varepsilon$. 如$l=(10.2\pm 0.05)\mathrm{mm}$, $R=(1500\pm 10)\Omega$等.

2. 相对误差与相对误差限

误差不能完全刻画近似值的精度. 如测量百米跑道产生10cm的误差与测量一个1.2m长度的课桌产生1cm的误差, 就不能简单地认为后者更精确, 还应考虑被测值的大小.

定义1.2 误差与精确值的比值

$$e_r=\frac{e}{x^*}=\frac{x-x^*}{x^*} \tag{1.9}$$

称为x的相对误差. 相对误差是无量纲的量, 常用百分比表示.

相对误差也常不能准确计算, 而是用相对误差限来估计. 相对误差限

$$\varepsilon_r=\frac{\varepsilon}{|x^*|}\geqslant\frac{|x-x^*|}{|x^*|}=|e_r| \tag{1.10}$$

由于实际上x^*不知道, 常用x代x^*作分母, 用$\varepsilon_r=\frac{\varepsilon}{|x|}$表示相对误差限.

例1.5 在刚才测量的例子中, 若测得跑道长为$(100\pm 0.1)\mathrm{m}$, 课桌长为$(120\pm 1)\mathrm{cm}$, 则

$$\varepsilon_r^{(1)}=\frac{0.1}{100}=0.1\%$$

$$\varepsilon_r^{(2)}=\frac{1}{120}=0.83\%$$

显然后者比前者相对误差大.

1.2.3 有效数字

定义1.3 若近似值x的误差限ε是它某一位数字的半个单位, 就称x准确到该位数字, 且从这一位数字起直到前面第一个非零数字为止的所有数字称为x的有效数字.

通常所说的l位有效数字是指从左端第一位非零数字往右数至第$l+1$位数字, 并对第$l+1$位数字进行四舍五入而得到的近似数. 如$\pi=3.14159265\cdots$则3.14和3.1416分别有3位和5位有效数字. 而3.143相对于π也只能有3位有效数字.

在更多的情况下并不知道准确值x^*. 如果认为计算结果的各数位可靠, 将它四舍五入到某一位, 这时从这一位起到前面第一个非零数字共l位, 由于四舍五入的原因, 它与计算结果之差必不超过该位的半个单位. 习惯上说将计算结果保留l位有效数字. 如计算机上得到方程$x^3-x+1=0$的一个正根为1.32472, 保留4位有效数字的结果为1.325, 保留5位有效数字的结果为1.3247.

相对误差与有效数字的关系十分密切．定性地讲，相对误差越小，有效数字越多，反之亦正确．定量地讲，有如下两个定理．

定理 1.1 设近似值 $x=\pm 0.a_1a_2\cdots a_n\times 10^m$，有 n 位有效数字，则其相对误差限

$$\varepsilon_r\leqslant\frac{1}{2a_1}\times 10^{-n+1} \tag{1.11}$$

证明略．

定理 1.2 设近似值 $x=\pm 0.a_1a_2\cdots a_n\cdots\times 10^m$ 的相对误差限

$$\varepsilon_r\leqslant\frac{1}{2(a_1+1)}\times 10^{-n+1} \tag{1.12}$$

则它至少有 n 位有效数字．

证明

$$|x|\leqslant(a_1+1)\times 10^{m-1}$$

$$|x-x^*|=\frac{|x-x^*|}{|x|}\times|x|\leqslant\frac{1}{2(a_1+1)}\times 10^{-n+1}\times(a_1+1)\times 10^{m-1}=0.5\times 10^{m-n}$$

由定义 1.3 知 x 有 n 位有效数字．

对有效数字的观察比估计相对误差容易得多，如监视到有效数字在算法中某一步突然变少，便意味着相对误差在这一步突然扩大，这就是计算出问题的地方．

例 1.6 计算$\frac{1}{759}-\frac{1}{760}$，视已知数为精确值，用 4 位浮点数计算．

解

$$原式=0.1318\times 10^{-2}-0.1316\times 10^{-2}=0.2\times 10^{-5}$$

结果只剩 1 位有效数字，有效数字大量损失，造成相对误差的扩大．若通分后再计算

$$原式=\frac{1}{759\times 760}=\frac{1}{0.5768\times 10^6}=0.1734\times 10^{-5}$$

就得到 4 位有效数字的结果，从此例可知相近数字相减会扩大相对误差．

1.3 数值运算时误差的传播

当参与运算的数值带有误差时，结果也必然带有误差，问题是结果的误差与原始误差相比是否会扩大．

1.3.1 一元函数计算误差的传播

设 x 是 x^* 的近似值，则结果误差 $e(f(x))=f(x)-f(x^*)$，用 Taylor 展开式分析

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