# 实用计算机数值方法

陈明逵　凌永祥

西安交通大学出版社

# 实用计算机数值方法

陈明逵　凌永祥



西安交通大学出版社

## 内 容 提 要

本书内容包括数值计算和数值分析的基本概念、线性方程组的数值解法、数据近似、数值微积分、非线性方程求解、常微分方程数值解法和最优化计算方法.本书既着重介绍用数字电子计算机求实践中常见问题数值解的有效方法,又对数值计算中可能出现的问题及其处理方法给以足够的重视和分析,并配以较多的数值计算例子,以说明主要概念、方法和理论及其应用.

本书是为运用计算机进行数值计算的科技工作者自学而编写的一本入门性参考书。

(陕)新登字 007 号

实用计算机数值方法

陈明述 凌永祥

责任编辑 叶 涛

*

西安交通大学出版社出版

(邮政编码 710049)

陕西 富平县印刷厂印装

陕西省新华书店经销

*

开本 850×1168 1/32 印张 10.125 字数:256 千字

1992 年 6 月第 1 版 1992 年 6 月第 1 次印刷

印数:1——2000

ISBN7-5605-0487-6/O·85 定价:6.90 元

# 前　言

本书介绍的是进行科学计算(Scientific Computation)的方法.即,在数字电子计算机上对来自科学研究和工程实际中的数学问题,进行数值处理的方法及其有关的问题.这就决定了本书应包含两方面的内容:一是介绍在计算机上对应用中最常见的问题进行数值处理的可靠方法;二是要阐述科学计算中最基本的概念,如问题的性态、算法的稳定性、计算机中数的表示及其运算特点、误差与精度、计算中可能出现的问题及算法组织和描述等,以便读者能将所学方法应用于实际计算.

科学计算要解决的问题非常广泛,使用的数值方法多种多样,作为一本入门性的参考书,不可能包罗万象.这就必须选取那些解最典型问题并经实践证明在计算机上行之有效的方法.因此,本书在取材上力求反映近代数值计算方法的发展,摒弃较陈旧且不适用的内容,加入一些目前已被广泛使用的方法.如,解最小二乘问题的正交化方法、稀疏矩阵的处理、自适应积分方法、最优化方法等.对实际计算中可能出现的问题及其处理方法,书中都给预足够的重视.考虑到读者的情况,不得不舍去一些需要较多数学基础知识的问题和方法,如代数特征值问题和偏微分方程数值解法等.但这并非憾事,因为深入地讨论一部分典型问题比泛泛地叙述所有问题,往往更能使读者深刻领会处理科学计算的主要思想.同时,并不需要读者具备较高的数学修养,凡具备了一般"高等数学"和"线性代数"基本知识的读者,便可顺利地阅读本书.当然,若读者有使用某种程序设计语言在计算机上解题的经验,对阅读本书将会大有帮助.除正文外,书末有三个附录,汇集了阅读本书所需的数学基础知识,供读者查阅.

本书虽然包含了计算方法的传统内容,如线性和非线性方程求解、多项式插值和近似、数值微分和积分、常微分方程数值解等,但在讲述时,是按照最后形成可靠算法的需要来进行的.方法的推

导及有关理论的证明，大都是根据如何构造和使用这些方法而精心组织的．尽可能详细地阐明其意义及处理问题的思想，并以适当的例子和算法描述来帮助读者理解，除第一章外，每章末尾都有一个小结，简要回顾本章所讲内容，并对各种方法预以评介，此外，对当前广泛使用、而本书未予介绍的方法和需要较多数学基础的方法，给以简要介绍并指出参考文献．这样，在学完本书以后，将使读者不但有了进一步学习更多新方法的基础，而且具备了将所学方法应用于实际计算的能力．

学习数值方法的目的主要是应用．因此，如何组织一个算法，将它在计算机上实现，并能分析计算结果，是读者阅读本书的一个重要目的．本书强调了各种方法的算法描述，通过从方法到算法的组织、描述过程，使读者更清楚地了解整个计算过程，加深对所学方法的认识．由于算法描述的目的只在于此，它与具体的程序又有所区别，因此，所有算法描述都力求与书中的方法一致，致使有些算法不是按最优的方式组织的，但稍加改进，都可成为实用的程序．因而本书很适合于所有欲用计算机进行科学计算的人们作为自学参考书．

本书是根据编者多年的教学和科学研究经验编写而成的。本书第六章和第七章由凌永祥编写，其余各章及三个附录均由陈明逵编写．

华中理工大学于寅教授和浙江大学李有法副教授仔细地审阅了全书，提出了不少宝贵的修改意见．西安交通大学计算数学教研室部分同志曾用过我们的讲义，为本书的形成和定稿提供了不少建议和帮助．西安交通大学研究生院和西安交通大学出版社对本书的写作和出版给予了大力支持．在此，谨表诚挚的谢意．

由于水平所限，谬误和不妥之处在所难免，恳请同行专家和广大读者不吝赐教．

编　者

1991.9.于西安交通大学

# 目 录

# 第一章 绪 论

## §1.1 数值计算

自第一台电子数字计算机(以下简称计算机)于1946年问世以来,计算机在科学技术各个领域中广泛使用所取得的成就无可辩驳地说明,科学方法的第三个分支——计算的方法——已经建立起来了.今天,除了传统的理论方法和实验方法以外,计算的方法已渗透到自然科学和社会科学的大多数领域中,研究工作和工程设计更加离不开计算的方法.它将许多领域的科学研究工作由定性阶段迅速地推向定量阶段.

计算机的高速计算能力和一定的逻辑功能使它成为数学应用于各领域最有力的工具.由于计算技术和计算数学的发展,对科学研究和工程实际中非常复杂的问题求数值解,例如从适当的计算结构设计和算法分析到物理、化学、生物、工程乃至社会科学中的很多问题,都可以而且必须用计算机才能获得满意的解答.

利用计算机解决实际问题,大多具有一个共同的方式.通常按以下三步进行.

1)建立数学模型.数学模型用来描述研究对象的某些性质,通常用数学方程来表示.为方便计,模型力求简单,但所有有关因素必须以无二义性的方式加以考虑,所以要作相应的假定.例如,在研究天体运动时,假定行星为质点;在研究经济问题时,模型中因素的依赖关系常假定为线性的.为使所确定的模型有意义,描述模型的数学问题必须是适定的.模型的构造及问题适定性的证明一般可用数学推导方式解决.

2)计算问题的解. 对应用来讲，只证明了问题的适定性是远不够的，必须知道具体的解是什么. 例如，对于空间旅行，理论上可以证明描述太空船轨迹的微分方程解的存在性，但若真要实施这一旅行，如太空船要在月球上着陆，仅有解的存在性结论是不够的，必须将航行的路线(轨迹)具体给出来. 这就要求把微分方程的解求出来.

3)实验验证. 在将计算所得结果应用于实践之前，必须验证所得结果是否符合客观规律，这一步是科学实验的一个重要目的. 若计算结果与实验不相符，必须重新考察前面两步的工作，予以适当修正、重新计算，直到获得满意的解答为止.

计算数学的任务主要是用计算机计算出数学问题的数值解. 从这种意义上说，计算数学是一门辅助性的学科，它是为各具体学科服务的. 它的着眼点不在于纯数学的、公理化的、独立于具体事物的抽象结论，而在于(通常来自其它领域的)具体问题的数值解.

计算数学的任务决定了它的研究内容，这就是寻找求解各种问题的数值方法，并对它们的数值性质进行研究，这种研究称为数值分析. 更具体地说，计算数学首先要构造可计算出各种问题解的数值方法，将其在计算机上实现，求出数值解，然后检查计算解是否可用，即是否符合实际. 为保证计算解的可靠性，需要对方法进行分析. 首先分析方法的可靠性，即，按此方法计算得到的解是否合理，与准确解之差是否很小，以确保计算解有用. 其次，要分析方法的效率，分析比较求解同一问题的各种方法的计算速度，以便使用者根据各自的情况采用高效率的方法，节省人力和物力，取得较好的经济效益. 这样的分析是数值分析的一个重要部分. 应当指出，数值方法的构造和分析是紧密相联不可分割的.

构造数值方法主要借助于数学推导，因此计算数学同数学的关系十分密切. 下面的两个例子示出了如何用数学推导来构造数值方法.

例 1.1　用数学的方法推导出计算实二次方程

$$ax^2 + bx + c = 0 \tag{1-1}$$

根的方法.

**解** 方程(1-1)在 $a \neq 0$ 的条件下等价于方程

$$x^2 + \frac{b}{a}x + \frac{c}{a} = 0$$

因对任何 $x, a, b$ 和 $c$ 都有

$$\begin{aligned} x^2 + \frac{b}{a}x + \frac{c}{a} &= x^2 + \frac{b}{a}x + \left(\frac{b}{2a}\right)^2 + \frac{c}{a} - \left(\frac{b}{2a}\right)^2 \\ &= \left(x + \frac{b}{2a}\right)^2 + \frac{c}{a} - \left(\frac{b}{2a}\right)^2 \end{aligned}$$

故 $x$ 是方程(1-1)根的充要条件是

$$\left(x + \frac{b}{2a}\right)^2 = \left(\frac{b}{2a}\right)^2 - \frac{c}{a}$$

从而，当 $\left(\frac{b}{2a}\right)^2 - \frac{c}{a} \geqslant 0$，即 $b^2 - 4ac \geqslant 0$ 时，方程(1-1)有实根

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

而当 $b^2 - 4ac < 0$ 时，方程(1-1)有一对共轭复根

$$x = -\frac{b}{2a} \pm i\frac{\sqrt{|b^2 - 4ac|}}{2a} \tag{1-3}$$

因此，计算方程(1-1)的根就应该首先判断 $b^2 - 4ac$ 的符号，然后确定选用式(1-2)还是式(1-3)进行计算. 这样，我们就借助于数学推导得到计算实系数二次方程根的一种方法.

**例 1.2** 构造一个求解 $n$ 元线性方程组

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = \beta_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = \beta_2 \\ \quad\vdots \qquad\qquad\qquad \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = \beta_n \end{cases} \tag{1-4}$$

的方法.

**解** 线性方程组(1-4)可以写成矩阵形式

$$Ax = b \tag{1-4'}$$

式中

$$A=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

是 $n$ 阶矩阵，称为方程组的系数矩阵.

$$x=(x_1,x_2,\cdots,x_n)^{\mathrm{T}},\qquad b=(\beta_1,\beta_2,\cdots,\beta_n)^{\mathrm{T}}$$

是 $n$ 维向量，分别称为**未知向量**和**右端向量**. 将矩阵 $A$ 按列分块成

$$A=(a_1\ a_2\ \cdots\ a_n)$$

这里，$a_j$ 是 $A$ 的第 $j$ 列，即

$$a_j=(a_{1j},a_{2j},\cdots,a_{nj})^{\mathrm{T}},\qquad j=1,2,\cdots,n$$

以 $b$ 代替 $A$ 的第 $i$ 列，并根据式(1-4')，便有

$$(a_1\ a_2\ \cdots\ a_{i-1}\ b\ a_{i+1}\ \cdots\ a_n)=(a_1\ a_2\ \cdots\ a_{i-1}\ Ax\ a_{i+1}\ \cdots\ a_n)$$

从而

$$(a_1\ a_2\ \cdots\ a_{i-1}\ b\ a_{i+1}\cdots\ a_n)=A(e_1\ e_2\ \cdots\ e_{i-1}\ x\ e_{i+1}\ \cdots\ e_n) \tag{1-5}$$

式中，$e_j$ 是 $n$ 阶单位矩阵的第 $j$ 列，即

$$e_j=(0,\cdots,0,1,0,\cdots,0)^{\mathrm{T}},\qquad j=1,2,\cdots,n$$

第 $j$ 个

将等式(1-5)两边取行列式便得

$$\begin{aligned}\det(a_1\ \cdots\ a_{i-1}\ b\ a_{i+1}\ \cdots\ a_n)&=\det(A)\det(e_1\ \cdots\ e_{i-1}\ x\ e_{i+1}\ \cdots\ e_n)\\&=x_i\det(A)\end{aligned}$$

因此，当 $A$ 非奇，即 $\det(A)\neq 0$ 时

$$x_i=\frac{\det(a_1\ \cdots\ a_{i-1}\ b\ a_{i+1}\ \cdots\ a_n)}{\det(A)} \tag{1-6}$$

对 $i=1,2,\cdots,n$ 均成立. 上式通常写成形式

$$x_i=\frac{\begin{vmatrix} a_{11} & \cdots & a_{1i-1} & \beta_1 & a_{1i+1} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2i-1} & \beta_2 & a_{2i+1} & \cdots & a_{2n} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ a_{n1} & \cdots & a_{1,i-1} & \beta_n & a_{1,i+1} & \cdots & a_{nn} \end{vmatrix}}{\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}}, \quad i=1,2,\cdots,n \tag{1-6'}$$

这就是计算方程组(1-4)解的公式，称为 Cramer 法则.

上面两个例子给出了用数学推导构造求解数学问题的方法，并得到解的公式. 但并不是一切问题的解都能用一个代数式表示出来. 例如，Galois 证明了五次以上的代数方程不可能用公式求解. 事实上，在以后各章将会看到，大多数问题的解不可能用公式来表示. 例如，计算函数 $f(x)$ 的积分

$$I=\int_a^b f(x)dx \tag{1-7}$$

只有在极少数情况下才能用 Newton-Leibniz 公式计算，而在一般情况下，取 $h=(b-a)/n$，$x_k=a+kh(k=0,1,\cdots,n)$，用

$$S=\sum_{k=1}^{n} f(x_k)h \tag{1-8}$$

计算，无论 $n$ 多么大，都只能获得定积分(1-7)的近似值. 用式(1-8)进行计算就是求定积分(1-7)的一种(近似)数值方法.

又如计算初等函数 $e^x$ 在给定点处的值，它不能用一个简单的代数式进行计算. 一种近似的替代是用其 Taylor 展开式

$$e^x=\sum_{k=0}^{\infty}\frac{x^k}{k!} \tag{1-9}$$

的前面若干项，即

$$S_n=\sum_{k=0}^{n}\frac{x^k}{k!} \tag{1-10}$$

无论用式(1-8)来计算定积分(1-7),还是用式(1-10)来计算 $e^x$ 的值,都是将无穷项的尾部截去而得到一个可以计算的有限项之和.这类方法称为近似替代法,它是数值计算中常用的一类方法,常常用在一些有关复杂函数的计算问题中.

上述的几个例子都说明,如何用数学手段构造求解问题的方法(或公式),但所获得的前两个方法同后两个方法是不同的:前两个是精确的,而后两个是近似的.

## §1.2 数值方法的分析

对数值方法进行分析的目的是要评价方法的优缺点,以便从各种方法中挑选出好的方法,以供使用.可以从不同的观点来评价一个方法的"好坏",但我们着重强调两点:一是方法的可靠性,二是方法的计算效率.所谓方法的计算效率是指解一个问题所花的代价,主要以在计算机上运行此方法所用的时间来衡量.为分析方便,通常以计算机完成操作

$$a+b*c\Rightarrow$$

所需的时间作为一个时间单位,称为**浮点运算**,①简记为 flop (*floating point operations*).我们用方法所需的浮点运算数目来度量方法的效率.所需浮点运算数越少,效率越高.

**例 1.3** 设 $A_1$、$A_2$、$A_3$ 和 $A_4$ 分别是 $10\times20$、$20\times50$、$50\times1$ 和 $1\times100$ 的矩阵.试比较按不同运算顺序求矩阵乘积

$$P=A_1A_2A_3A_4$$

的几种方法的效率.

**解** 根据矩阵乘法的结合律,可按不同的运算顺序计算矩阵

① 这里所说的浮点运算指计算机完成一次浮点加法、一次浮点乘法和少量的辅助性操作(如存取数据、下标查找等)所需的时间.因为在大多数的计算中,加减法的数目大约与乘除法的数目相同.

乘积 $P$. 我们以下面三种顺序作为例子.

1) $P=[(A_1A_2)A_3]A_4$

2) $P=A_1[A_2(A_3A_4)]$

3) $P=[A_1(A_2A_3)]A_4$

按这三种不同的计算顺序，分别需要 11 500、125 000 和 2 200flop. 显然，第三种方法效率最高.

一个方法的效率高低，不仅仅是一个经济问题，而且反映了一个算法是否实际可行. 例 1.2 中所述的计算 $n$ 元线性方程组解的 Cramer 法则对较大的 $n$ 是一个不可行的方法. 因为要计算 $n+1$ 个行列式. 若按行列式的定义计算，每个行列式的值都由 $n!$ 项之和组成，每项是 $n$ 个数的乘积，故计算一个 $n$ 阶行列式的运算量约为 $(n-1)(n!)$flop，从而，计算 $n$ 阶线性方程组的解的总运算量是

$$N=(n+1)(n-1)(n!)+n$$

当 $n=20$(并不太大)时，$N$ 约为 $9.707\times10^{20}$. 如此大的计算量即使在一台速度为每秒一亿次的计算机上也要算 30 多万年！这个方法显然是不可行的. 若用 Gauss 消去法(见 §2.1)，解这样一个方程组约需 3 060 flop，即使在一台小型计算机上，也可在一秒钟之内完成. 由此可见，分析方法的效率是十分有意义的.

数值分析的一个更重要并且相对而言较困难的任务是数值方法的可靠性分析(数值稳定性分析). 数学家在解决问题时，都要先假定所论数的范围，一般都假定计算在实数范围进行. 这一假定大大简化了问题的讨论. 但当我们在一台计算机上解题时，运算不可能在实数系中进行. 因为实数系是无限的：实数范围无限，即它包含无限大的数(正数和负数)；实数又是稠密的，即任何两个实数间都包含无限多个实数. 相反，任何计算机上的数都是有限的. 计算机只能表示一给定有限区间中有限个不连续的数. 这就使得在计算机上的计算与在实数系内的计算有很大的差别.

### 1.2.1 计算机上数的运算

任何一个 $\beta$ 进制、具有 $t$ 位有效数字的实数 $x$ 总可以表示成

$$x = \pm \left( \frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \cdots + \frac{d_t}{\beta^t} \right) \times \beta^l \tag{1-11}$$

其中，$d_j$ 是满足

$$1 \leqslant d_1 < \beta$$
$$0 \leqslant d_j < \beta, \quad j = 2,3,\cdots,t$$

的整数. $\beta^l$ 称为**指数部分**，$d_1 d_2 \cdots d_t$ 称为**尾数**. 通常用的是 $\beta=10$ 的所谓十进制数. 而在计算机中用的是二进制数，即 $\beta=2$.

若用有 $t$ 位尾数的 $\beta$ 进制数，就只需记录 $l$ 和 $\pm d_1 \cdots d_t$，这就是在计算机中表示的所谓浮点形式的数. 在现代计算机中，数都是以二进制浮点形式表示的(在早期的计算机中，有不少是定点形式的). 此外，由于计算机字长有限，指数 $l$ 也总是有限的，即满足 $L \leqslant l \leqslant U$，这里，$L$ 和 $U$ 分别称为**指数 $l$ 的下界与上界**.

按上述表示方法，计算机中能表示的全体数的集合称为**计算机的浮点数系**，以符号 $F(\beta,t,L,U)$ 记之. 显然，尾数部分的位数 $t$ 越大，能表示数的准确度越高. 因此，$t$ 常称为**数系 $F(\beta,t,L,U)$ 的精度**. 浮点数系除了硬件实现方便以外，它还有另外的优点，就是在其可表示数的范围之内，它在实数系 $R$ 中的相对密度是均匀的. 例如，数系 $F(10,4,-2,3)$ 能表示数的范围是 $\pm[0.001, 999.9]$ 及数零. 在此数系中，数 865.54 表示成 $0.865\,5\times10^3$，而 0.865 54表示成 $0.865\,5\times10^0$. 其相对误差均是 0.005%.

为了在数系 $F(10,4,-2,3)$ 中表示数 865.54，必须舍去其最后一位数字，这就产生了误差. 一般来说，一个实数用计算机数系中浮点数表示时，要以最接近于它的一个浮点数来代替它，因而产生误差. 此外，在浮点数系中，四则运算是非封闭的，即浮点数系中的任意两个数的和、差、积或商不一定仍在此数系中. 为了使经过算术运算产生的浮点数能以同一浮点数系中的数表示，必须用接近于它的一个浮点数来代替，这也要产生误差.

浮点数运算结果越出浮点数系有两种情况：

1)结果的指数 $l$ 可能不在范围 $[L,U]$ 之内. 例如，在数系