http://www.phei.com.cn

计算方法

主　编　张世禄
副主编　陈豫眉　谭代伦

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

计 算 方 法

主　编　张世禄
副主编　陈豫眉　谭代伦

電子工業出版社
Publishing House of Electronics Industry
北京 • BEIJING

内 容 简 介

本书共 11 章，主要介绍数值代数和数值逼近中常用的实用算法。书中提供的算法都是带计算过程和计算条件的数学公式，除解线性方程组的算法因既要考虑复杂性又要考虑内存开销之外，其余算式都与程序语言中语句有一对一的映射关系。

本书可作为计算学科各专业、数学专业及理工科本科生教材，也可作为工科硕士研究生教材。

图书在版编目（CIP）数据

计算方法 / 张世禄主编. —北京：电子工业出版社，2006.6
ISBN 7-121-02539-6

I. 计… II. 张… III. 计算方法 IV. O241

中国版本图书馆 CIP 数据核字（2006）第 040743 号

责任编辑：张燕虹
审　　校：于秀山
印　　刷：北京京科印刷有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编 100036
经　　销：各地新华书店
开　　本：787×1092　1/16　印张：14.5　字数：380 千字
印　　次：2006 年 6 月第 1 次印刷
定　　价：22.00 元

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系。联系电话：（010）68279077。质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

前　言

计算方法虽是数学领域中的一个古老分支，但今天的计算方法已不再是人们手算的方法，而是计算机使用的方法。当今的科学技术计算问题几乎都用计算机来计算结果，但计算机不能识别计算方法，只能识别人们按算法用计算机语言所编写的程序，因此计算方法应考虑程序设计，应考虑程序。

在冯康先生所著的计算方法教材中每个算法都附有一个 ALGOL 程序，中外不少计算方法教材都用计算机语言表述算法的计算过程。本书既用计算机语言描述计算过程，又给出了算法中各计算公式和计算机语言中语句的对应关系。

程序是计算机使用的算法，程序设计就是将非计算机使用的算法翻译成计算机使用的算法。算法表述不同，翻译难度不同。程序运行过程是一动态计算过程，自然语言宜用于描绘自然现象和社会现象，不易翻译成动态计算过程，静态图表也难准确表述动态计算过程，软件工程中介绍了不少动态图表，但是功能各异，互有优劣，难以规范。本书所介绍的所有算法都用带计算过程和计算条件的数学公式表述。除了第 2 章、第 4 章的算法因既要考虑算法复杂性又要考虑内存开销，故使用了不少程序设计技术专用知识，使得同一个数组在不同时刻表示不同含义，因此程序中的语句和算法中算式无明确的对应关系，其他算法和程序中语句都有一对一的关系。

通常的计算方法都是按数学问题分类的。本教材除了按数学问题分类之外，还按程序结构再次分类。设计一类程序虽然不能一劳永逸，但总比一个个设计程序效率更高。

计算方法中除第 2 章解线性方程组的直接法之外，其他算法都是经过一定假定和简化后才得出的，因此本身就是数学模型。教材简明扼要地介绍了各种算法的归纳和推导过程，并对算法本身的收敛性、稳定性进行了证明。书中所给出的不少算法的计算过程和表述形式，以及对定理的证明都颇具特色。书中每个算法都附有计算实例，每个计算实例都给出了计算过程和计算结果，并对计算结果用计算机给予了验证，以保证计算结果正确无误。

为了增强教材的实用性和逻辑性，教材增加了压缩存储迭代法，解 $f(x)=0$ 在 $x\in[a,b]$ 所有根的算法，增加了二重数值积分算法，求导数、2 阶导数算法。除了二重积分计算以外，其余算法都可用手算。

书中符号的使用说明如下：

（1）黑体的大写字母表示矩阵，黑体的小写字母表示向量。

（2）正文和公式中的英文字母用斜体，程序中的英文字母用正体。

（3）ε 表示误差限（或迭代精度），在程序段中用 ep 表示。正文和公式中的 f'，y'，y_0'，y_n' 在程序段中用 fep，yep，yoep，ynep 表示。在程序代码中用 w 表示数学符号 ω。

（4）数学符号中，除%的意义与 C 语言中的意义相同外，其他符号的意义与数学中的意义相同。

本教材第 1 章、第 2 章、第 6 章至第 11 章由张世禄编写，第 3 章、第 4 章、第 5 章由陈豫眉编写，全书所有计算结果都由谭代伦用 C 程序做了验证。

张世禄

目　录

第1章 误 差

就现实而言，计算方法实际上是计算机使用的方法，因为对于工程物理问题，科学技术计算问题很少能通过手算就得到用户认可的结果。

计算方法不追求精确解，只追求满意解，那种认为“计算结果越精确越好，能得到理论解为最好”的观点本身就不科学，计算方法仅用于求满意解，所谓满意解就是在允许的误差范围内的解。精度太高会增大成本且无意义。精确解根本就不存在。理论解也仅仅理论上存在。

1.1 误差、误差计算和误差来源

何谓误差？误差不是失误引起的差，误差是理论值与近似值之间的差异，有误差是正常的、绝对的，无误差虽是理想的，但却是不现实的。

近似值包括测量值和计算值。这里须强调，理论值不是精确值。

1.1.1 误差表示法

误差分为绝对误差和相对误差两种。

绝对误差就是理论值与近似值的差，简称误差。设 x^* 表示理论值，x 表示近似值，$\varepsilon(x)$ 表示绝对误差，则

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

绝对误差可正可负。在工程技术问题中，必须考虑绝对误差的正负。例如在设计活塞和汽缸时，若以汽缸为准，活塞的半径只能比汽缸小，此时 x 必须小于 x^*，即 $\varepsilon(x)$ 必须大于 0。在计算方法中，绝对误差的正负无意义，因而常用其绝对值表示。

一般情况下，用绝对误差表示误差会显得不科学、不准确，绝对误差未能刻画出近似值的精确程度，因此要引入相对误差。

所谓相对误差，就是绝对误差与理论值的比值。若用 $\varepsilon_r(x)$ 表示相对误差，则

$$\varepsilon_r(x)=\frac{\varepsilon(x)}{x^*}=\frac{x^*-x}{x^*} \tag{1-2}$$

在实际计算中，由于理论值 x^* 通常是未知的，因此式（1-2）中的分母常用近似值 x 代替。

1.1.2 误差限

误差限与误差种类有关，误差限也有绝对误差限和相对误差限之分。

设 $\eta(x)$ 为绝对误差限，则

$$\eta(x)=\max|\varepsilon(x)| \tag{1-3}$$

设 $\eta_r(x)$ 为相对误差限，则

$$\eta_r(x)=\max|\varepsilon_r(x)| \tag{1-4}$$

在工程物理计算中，建立物理模型和数学模型后，用户会根据问题的性质和要求给出误差限，然后才由软件人员选择算法，选择算法的重要依据之一就是误差限。对于迭代算法，

迭代精度应不低于绝对误差限，对于数值逼近方面的算法，截断误差（量级）应与绝对误差限一致。相对误差限主要用于对测量仪器仪表的精度选择。相关知识请参考其他有关书籍。

1.1.3 误差计算

误差计算实际上是误差估计，误差本身不可能是准确值，因理论值是不知道的，正因为是估计，所以与一般计算不同，误差计算仅考虑相对误差的计算。

对式（1-2），分母 x^* 用其近似值 x 代替，则

$$\varepsilon_r(x)=\frac{\varepsilon(x)}{x}=\frac{\Delta x}{x}\approx\frac{\mathrm{d}x}{x}=\mathrm{d}\ln x \tag{1-5}$$

式（1-5）取$\Delta x>0$，且 $x>0$。当 $x<0$ 时，考虑$-\Delta(-x)$，此时式（1-5）仍可用。

1. 乘积误差

乘积误差为：

$$\varepsilon_r(x_1x_2)\approx\mathrm{d}\ln x_1x_2=\mathrm{d}(\ln x_1+\ln x_2)\approx\varepsilon_r(x_1)+\varepsilon_r(x_2) \tag{1-6}$$

2. 商的误差

商的误差为：

$$\varepsilon_r\left(\frac{x_1}{x_2}\right)\approx\varepsilon_r(x_1)-\varepsilon_r(x_2) \tag{1-7}$$

3. 和的误差

和的误差为：

$$\begin{aligned}\varepsilon_r(x_1+x_2)&\approx\frac{\mathrm{d}(x_1+x_2)}{x_1+x_2}=\frac{x_1}{x_1+x_2}\times\frac{\mathrm{d}x_1}{x_1}+\frac{x_2}{x_1+x_2}\times\frac{\mathrm{d}x_2}{x_2}\\&\approx\frac{x_1}{x_1+x_2}\varepsilon_r(x_1)+\frac{x_2}{x_1+x_2}\varepsilon_r(x_2)\end{aligned} \tag{1-8}$$

当+号变成−号时，式（1-8）就成了差的误差计算公式。

从式（1-8）可看出：

（1）当 x_1 和 x_2 异号且绝对值相差极小时，$\varepsilon_r(x_1+x_2)$ 可能相当大，即 $x_1+x_2\approx0$ 时相对误差会极大，因此应尽量避免相近数相减，这也是当 $x\approx0$ 时常使用绝对误差的原因。

（2）当$|x_1|\gg|x_2|$时，$\varepsilon_r(x_1+x_2)\approx\varepsilon_r(x_1)$。

1.1.4 误差来源

误差来源有以下 4 种情况。

1. 模型误差

对于所有工程物理问题，必须通过计算机计算才知道其结果。计算机只能按人们设计的程序计算，在编写程序前都必须建立物理模型和数学模型，物理模型和数学模型都是经过了一定假设和简化才建立起来的。

例如，某研究所为一冶金企业设计了一冶炼过程控制系统，首先假设问题满足热力学第一定律，遵从热传导方程，假设炉体是绝对绝热的，从而建立了物理模型；然后假设炉体是

无限长直圆柱，从而使一复杂的三维热力学问题变成了边界简单的一维热力学问题。

世界上，不存在绝对绝热的材料，也不存在几何中的直圆柱，更无法实现无限长，但是若考虑热量散失，将炉体考虑成有限长几何体，则数学方程就难以建立，即使建立起来了，要想求解一个三维的边界复杂的热传导方程（偏微分方程）至少需要数小时才能得到结果。然而按假设计算，则能在数秒内得到比人工采样和化验更准确、更及时的结果。

问题经假设和简化后，显然已与实际问题有差异，其结果与精确值有差异，这些差异就是模型误差，计算方法不考虑模型误差。由于模型误差总是存在的，因此计算方法里不存在精确解，最多只有理论上的精确解——理论解。

2. 测试误差

对于所有科技计算问题，都须输入数据，输入的数据中一部分是用仪器、仪表以及测量工具测出的，另一部分是直接从传感器（实际上也是测量工具）中经过模数转换得到的。由于所有仪器、传感器本身是有测量精度的，例如某天秤的最小砝码是 0.1mg，那么对于小于 0.1mg 的物体，该天秤就无法准确测出它的质量，因此所测得的数据与实际值之间是有差异的，这类误差称为测试误差。

3. 舍入误差

计算机中的所有数都存放在存储器中，由于存储器的位数是有限的，无论用多少字节存放一个数，它所能容纳的数的位数也是有限的。对于定点数而言，绝对值比它所能表示的最小的数还要小的数就不能表示，对这一部分只能四舍五入或者全部舍去，四舍五入或者全部舍去会带来误差，这类误差与舍入有关，称为舍入误差。

例如，某计算机在某一确定软件支持下，定点数绝对值不能小于 10^{-7}，舍取方式为四舍五入，则

$$0.1+10^{-8}=0.1$$

$$0.1+5\times10^{-8}=0.100\,000\,1$$

这里用等号的原因是，我们看到的就是 0.100 000 0 和 0.100 000 1。

对于浮点数也同样存在舍入误差。

舍入误差看来很小，但当算法不当时，经过传递和放大，则可能变得相当大，下一节将专门介绍。

4. 截断误差

在不考虑测试误差、舍入误差的前提下，理论值与计算值之间的差，称为截断误差。截断误差是算法本身的误差，故截断误差也称为先天误差。例如：

$$e=\sum_{i=0}^{\infty}\frac{1}{i!}=1+\frac{1}{2!}+\cdots+\frac{1}{n!}+\cdots$$

取前 n 项和作为 e 的近似值

$$e\approx\sum_{i=0}^{n}\frac{1}{i!}=1+\frac{1}{2!}+\cdots+\frac{1}{n!}$$

这样计算出的近似值和理论值之间存在误差 $\sum_{i=n+1}^{\infty}\frac{1}{i!}$。这种类型的误差就是截断误差。

1.1.5 有效数字及与相对误差之间的关系

对于近似值，还需要知道它的准确程度，因此引入有效数字这个概念。

【定义】 若近似值 x 与理论值 x^* 的误差不超过某位数字的半个单位，十进制下可表示为 $|x^*-x|\leqslant\frac{1}{2}\times10^{-n}$，则称近似值 x 准确到小数点后第 n 位，或者称从小数点后第 n 位数字到 x 最左边的第一个非零数字全部为有效数字。

【例 1】 已知 e = 2.718 28…，问下列 e 的近似值有几位有效数字？

（1）2.7　　（2）2.718 3

【解】 （1）$|e-2.7|=0.018\,28\cdots<\frac{1}{2}\times10^{-1}$

则近似值 2.7 准确到小数点后第 1 位，所以从小数点后第 1 位数字到最左边第 1 个非零数字全部为有效数字，因此 2.7 作为 e 的近似值有 2 位有效数字，分别为 2 和 7。

（2）$|e-2.718\,3|=0.000\,02\cdots<\frac{1}{2}\times10^{-4}$

则近似值 2.718 3 准确到小数点后第 4 位，所以从小数点后第 4 位数字到最左边第 1 个非零数字全部为有效数字，因此 2.718 3 作为 e 的近似值有 5 位有效数字，分别为 2，7，1，8，3。

【注 1】 如果近似值 x 是由理论值 x^* 四舍五入得到的，则其误差不超过末位数字的半个单位，因此自左向右的第一个非零数字到由四舍五入得到的最末一个数字均为近似值 x 的有效数字。通常情况下，默认近似值 x 由四舍五入得到。

【例 2】 指出下列近似值的有效数字并估计其绝对误差限。

1.02　　1.020　　0.035　　381.70

【解】 由于默认上述近似值均由四舍五入得到，故自左向右的第 1 个非零数字到最末一个数字均为近似值的有效数字，因此上述近似值的有效数字分别为 3，4，2，5。

由于四舍五入得到的近似值的误差都不超过末位数字的半个单位，因此上述近似值的误差限分别为 $\frac{1}{2}\times10^{-2}$，$\frac{1}{2}\times10^{-3}$，$\frac{1}{2}\times10^{-3}$，$\frac{1}{2}\times10^{-2}$。

【注 2】 如果近似值 x 用浮点数表示为

$$x=\pm0.x_1x_2\cdots x_p\cdots x_n\times10^m \tag{1-9}$$

式（1-9）中，$x_1\neq0$，其余数为 0～9 中任何数字。如果

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

即近似值 x 的误差限不超过 10^{m-p} 位的半个单位，则近似值 x 准确到 10^{m-p} 位。当 $p=n$ 时，近似值 x 具有 n 位有效数字，对于 0.035 应写成 0.35×10^{-1}，才能算出舍入误差限。

对于理论值 x^* 和它的近似值 x 而言，近似值的有效位数越多越精确。有效数字和相对误差限的关系由下面两方面决定。

（1）若由式（1-9）表示的近似值 x 具有 n 位有效数字，显然

$$0.x_1\times10^m\leqslant x\leqslant0.(x_1+1)\times10^m$$

于是有

$$|\varepsilon_r(x)|=\left|\frac{x^*-x}{x}\right|\leqslant\frac{\frac{1}{2}\times10^{m-n}}{0.x_1\times10^m}=\frac{1}{2x_1}\times10^{1-n} \tag{1-10}$$

（2）形如式（1-9）的近似值 x，若其相对误差 $\varepsilon_r(x)$ 满足

$$|\varepsilon_r(x)| \leqslant \frac{1}{2(x_1+1)} \times 10^{1-n} \tag{1-11}$$

则近似值 x 至少具有 n 位有效数字。

因为 $|x^* - x| = |\varepsilon_r(x)| \cdot |x^*| \leqslant \frac{1}{2(x_1+1)} \times 10^{1-n} \times (x_1+1) \times 10^{m-1} = \frac{1}{2} \times 10^{m-n}$

由注 2 可知：近似值 x 至少具有 n 位有效数字。

1.2 算法选择

计算方法、应用数学、数据结构为程序设计提供了不少算法，解决同一问题可用多种算法。这些算法中，有的精度高，有的精度低；有的能得到满意的结果，有的不能；有的耗费时间多，有的耗费时间少。下面介绍算法选择原则。

1.2.1 正确性

正确性不是指公式是否正确，它包含下述 4 方面的内容。

1. 误差在误差限内

由于还未正式涉及计算方法中的算法，故所有例子最多只用到数学分析中的例子。

【例 1】 计算 $e^{-x} = \sum_{i=0}^{n} (-1)^i \frac{x^i}{i!}$，$x \in [0,1]$，误差不大于 10^{-6}。

【解】 由数学知识知，上式截断误差为：

$$|\varepsilon(x)| = \frac{e^{-\xi}}{(n+1)!} \leqslant \frac{1}{(n+1)!} \leqslant 10^{-6}$$

由上式可确定 n=10。

若 n=7，虽算出一个 e，但该值不正确，不满足精度要求。

绝大多数情况下，$\varepsilon(x)$无法用数学公式表示，因此常使用结果比较法（迭代法使用方法），即比较相邻两次的计算结果，取其差，当差的绝对值小于ε 时，计算结束。

设 $e^{(n)}(x) = \sum_{i=0}^{n} (-1)^i \frac{1}{i!}$，$\varepsilon = 10^{-6}$，则计算公式为：

$$|e^{(n)}(x) - e^{(n-1)}(x)| = \frac{1}{n!} < 10^{-6}，\ n=0,1,\cdots \tag{1-12}$$

式（1-12）对应一个 while 型循环，循环条件为 $\frac{1}{i!} > \varepsilon$，该式所对应的程序如下：

```
main()
{ float s,ep,p;
  int i;
  scanf("%f",&ep);
  p=1.0;
  i=0;
  s=1;
```

```
    while(p>ep)
    { i++;
      p=p/i;
      s=s+p;
    }
    printf("s=%f,i=%d",s,i);
}
```

程序运算结果为：s=2.718 282, i=10。

注意

程序中循环前 p=1.0，表示 $\frac{1}{0!}$。将 p 设成实型变量，且直接计算 $\frac{1.0}{i!}=\frac{1.0}{i(i-1)!}$，不是先计算 $i!$，再计算 $\frac{1.0}{i!}$，这不仅仅是节省计算量，而是考虑到了溢出，因为 i 较大时，$i!$可能很大，$i!$ 很容易超出整型数甚至长整数的范围，产生溢出，另外若将 p 设定为整型数，须知当 i>1 时，$\frac{1}{i}$=0。

对于本例不是将 n 取得越大越好，太大无意义。

2．必须在指定时间内算出结果

正确不仅仅表现为结果正确，还必须满足时间要求，必须在指定时间内算出结果。前面提到的冶炼（过程）控制系统，若按三维热传导方程编写程序，即使编出了程序，并建立好了控制系统，但由于算法复杂性太高，结果得出时金属液的状态早已与算前一刻大不一样，从而错过了时机。再举一例，战斗机上都配备了“大气数字计算机”，该计算机能在很短时间内算出敌我双方飞机位置、速度（马赫数）、压力和温度，作出飞行选择和攻击选择，所选择算法就必须在指定时间内算出相应参数。

这里须指出，不是任何系统都对高效率感兴趣，在一般商务管理程序（例如机票销售系统）中，3s 显示票价和 10s 显示票价并无本质差异，这类软件的可维护性、界面才是设计应优先考虑的问题。

3．收敛于理论解

这方面的内容将在以后的具体章节里加以介绍。

4．数值稳定

稳定性内容将在以后的章节里加以详细介绍，下面所介绍的内容实际上也与稳定性有关。算法选择时，还应考虑内存开销，本书不直接介绍这方面的内容。

1.2.2 算法选择的一般常识

商务、金融、管理软件所涉及的程序使用的算法大都与计算方法无关，但不少程序要用到算法选择的一般常识。