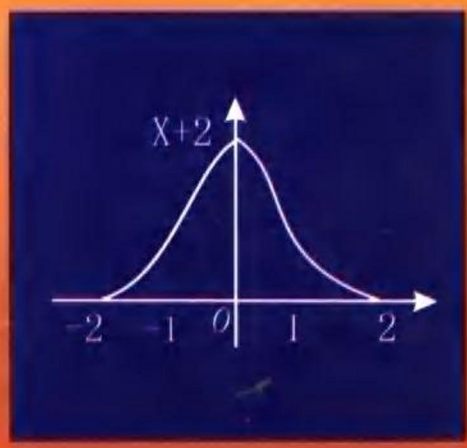
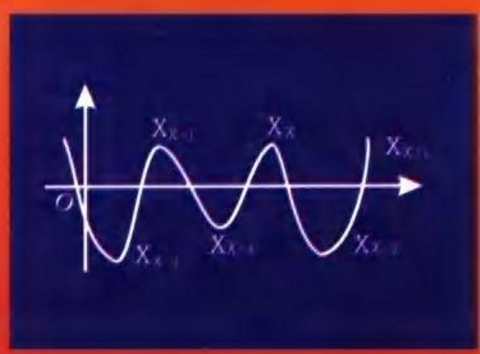
数值计算方法

林成森 编著

（上 册）





科学出版社

数值计算方法

（上册）

林成森 编著

科学出版社

1998

内 容 简 介

本书详细地介绍了计算机中常用的数值计算方法，主要内容包括：误差分析、解非线性方程的数值方法、解线性方程组的直接方法、插值法、数值积分．本书每章末均附有丰富、实用的习题．本书在南京大学数学系和计算机科学系作为教材．

本书可作为高校数学系、计算机系教材；也可供工程技术人员参考．

图书在版编目(CIP)数据

数值计算方法　上册/林成森编著．-北京：科学出版社，1998
ISBN 7-03-006189-6

Ⅰ．数… Ⅱ．林… Ⅲ．数值计算-计算方法 Ⅳ．0241

中国版本图书馆 CIP 数据核字(97)第 17466 号

科学出版社出版
北京东黄城根北街 16 号
邮政编码：100717
北京双青印刷厂 印刷

新华书店北京发行所发行　各地新华书店经售

*

1998 年 3 月第　一　版　开本：787×1092　1/16
1998 年 3 月第一次印刷　印张：15
印数：1－3 600　字数：344 000

定价：24.00 元

前　言

随着计算机的迅速发展，在科学、技术、工程、生产、医学、经济和人文等领域中抽象出来的许多数学问题可以应用计算机计算、求解．本书详细、系统地介绍了计算机中常用的数值计算方法和一些现代数值方法，以及有关理论．本书分上、下两册．上册主要内容有误差分析、解非线性方程的迭代法、解线性方程组的直接方法、插值法、数值积分．下册主要内容有解线性方程组的迭代法、线性最小二乘问题、矩阵特征值问题、解非线性方程组的数值方法、常微分方程初值和边值问题的数值解法、函数逼近．本书例题多，且各章末均附有丰富、实用的习题，其中有少数习题必须用计算机来求解．

本书中的基本数值方法和一些现代方法都用伪程序（pseudocode）给出，以便于在计算机上实现这些算法，读者容易把它们译成 FORTRAN，Pascal，C 语言或其它程序语言．

本书的原稿是作者多年讲授数值计算方法课程的讲义．它在南京大学数学系和计算机系及其他系中作为教材使用过，并经多次整理、修改和补充而最后定稿．本教材授课时间为一学年，如若安排一个学期讲授，可适当删减书中某些章节内容和一些理论．

沈祖和、苏维宜、姜东平、王嘉松、孙麟平、赵金熙、王长富等老师对作者完成这本书的编写给予热情的关心和鼓励，罗亮生先生为本书精心绘制了插图，在此表示衷心的感谢．

由于编者水平有限，在本书中尚有错误和不足之处，敬请使用本书的各位老师和读者批评指正．

林成森

1997 年 7 月

目　录

第一章　算术运算中的误差分析初步

§1　数值方法

从自然科学、工程技术、经济和医学等领域中产生的许多数学问题，我们得不到它的解的准确数值，从而需要寻找问题解的近似值的数值方法. 更确切地说，一个**数值方法**是对给定问题的输入数据和所需计算结果之间的关系的一种明确的描述. 例如，我们用 Newton 法(将在第二章 §4 中讨论)计算$\sqrt{3}$. 给定$\sqrt{3}$的一个初始近似值 $x_0(x_0>0)$，由迭代公式

$$x_n=\frac{1}{2}(x_{n-1}+\frac{3}{x_{n-1}}),n=1,2,\cdots$$

产生一个序列 $x_0,x_1,\cdots,x_n,\cdots$. 当然，我们必须在 n 充分大后停止计算，如 $n=m$. 这样，我们取 x_m 作为$\sqrt{3}$的一个近似值，即

$$\sqrt{3}\simeq x_m.$$

为了使一个数值方法在计算机上得到实现，我们需要给出数值方法的一种**算法**，它是算术和逻辑运算的完整描述，按一定顺序执行这些运算，经有限步把输入数据的每一个容许集转换成输出数据. 例如，计算$\sqrt{3}$的 Newton 法的一种算法：

输入　初始近似值 x_0；最大迭代次数 m.

输出　$\sqrt{3}$的近似值 p 或迭代失败信息.

step 1　$p_0\leftarrow x_0$.

step 2　对 $n=1,\cdots,m$ 做 step3－4.

　　step 3　$p\leftarrow(p_0+\frac{3}{p_0})/2$.

　　step 4　若$|p-p_0|<10^{-8}$，则输出(p)，停机，否则 $p_0\leftarrow p$.

step 5　输出('Method failed')；

　　停机.

算法中"←"表示赋值. 若输出"Method failed(方法失败)"，则表明迭代 m 次得到的 x_m，x_{m-1}仍不满足$|x_m-x_{m-1}|<10^{-8}$.

建立一个数值方法(算法)的基本原则应该是(1)便于在计算机上实现，(2)计算工作量尽量小，(3)存贮量尽量小，(4)问题的解与其近似解的误差小.

§2　误差来源

在数值计算中，由于下面一些原因会产生误差.

1. 数据误差　进行数值计算所使用的初始数据往往是近似的. 例如，$\pi=3.14159265\cdots$，

我们只能取有限小数，如取 $\pi\simeq 3.14159$. 有的初始数据可能是从实验或观测得到的. 由于观测手段的限制，得到的观测数据也必会有一定的误差，这种误差称为**观测误差**.

2. 截断误差 求一个级数的和或无究序列的极限时，我们取有限项作为它们的近似，从而产生了误差，这种误差通常称为**截断误差**. 例如，用 e^{-x} 的幂级数表达式

$$e^{-x}=1-x+\frac{1}{2}x^2-\frac{1}{6}x^3+\cdots \tag{2.1}$$

计算 e^{-x} 的值时，常常取级数的开头几项的部分和作为近似公式，如取

$$e^{-x}\simeq 1-x+\frac{1}{2}x^2-\frac{1}{6}x^3. \tag{2.2}$$

这样，用(2.2)式计算得到的 e^{-x} 的值是近似的. 它与 e^{-x} 的准确值之间是截断误差，这是由于截去(2.1)的余项产生的. 又如前述的用 Newton 法计算 $\sqrt{3}$，我们取 $\sqrt{3}\simeq x_n$，其截断误差是 $\sqrt{3}-x_n$.

3. 离散误差 在数值计算中，我们常常使用一个近似公式来求一个问题的解. 例如，求曲边梯形 $abBA$(图 1.1)的面积：

$$S=\int_a^b f(x)dx.$$

若用梯形 $abBA$ 的面积

$$T=\frac{b-a}{2}(f(a)+f(b))$$

作为 S 的近似值，则产生误差 $S-T$. 这种误差称为**离散误差**. 它是由于把连续型问题离散化而产生的.

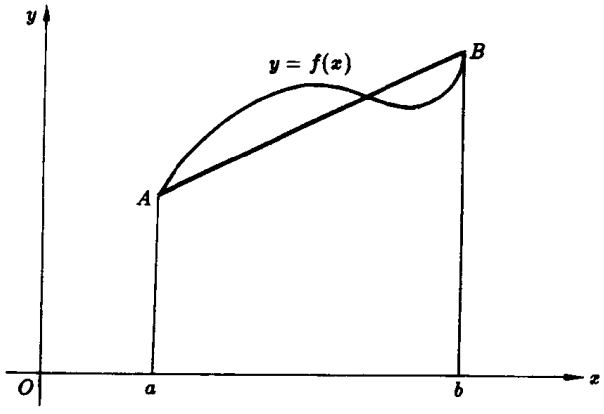


图 1.1

4. 数值计算过程中的误差 由于计算机的字长有限，进行数值计算的过程中，对计算得到的中间结果数据要使用“四舍五入”或其它规则取近似值，因而使计算过程有误差.

在一个数值方法中，通常至少有上述一种误差出现. 在许多数值方法中，会有上述多种误差出现.

§3 绝对误差和相对误差

我们有两种衡量误差大小的方法：一是绝对误差；二是相对误差.

假设某一个量的准确值(真值)为 x,其近似值为 $\bar{x}$. x 与 $\bar{x}$ 的差

$$e_{\bar{x}} = x - \bar{x} \tag{3.1}$$

称为近似值 $\bar{x}$ 的**绝对误差**(简称**误差**). (3.1)式也可改写成

$$x = \bar{x} + e_{\bar{x}}. \tag{3.2}$$

实际上,我们只能知道近似值 $\bar{x}$,而一般不知道准确值 x,但可对绝对误差的大小范围作出估计. 也就是说,可以指出一个正数 ε,使

$$|e_{\bar{x}}| = |x - \bar{x}| \leqslant \varepsilon. \tag{3.3}$$

我们称 ε 为近似值 $\bar{x}$ 的一个**绝对误差界**. 例如,$\pi=3.14159265\cdots$,取 $\bar{\pi}=3.14$,则

$$|\pi-\bar{\pi}|<0.002.$$

绝对误差还不足以刻画近似数的精确程度. 例如 $x=100$(厘米),$\bar{x}=99$,则 $e_{\bar{x}}=1$;而 $y=10000$(厘米),$\bar{y}=9950$,则 $e_{\bar{y}}=50$. 从表面上看,后者的绝对误差是前者的 50 倍. 但是,前者每厘米长度产生了 0.01 厘米的误差,而后者每厘米长度只产生 0.005 厘米的误差. 因此,要决定一个量的近似值的精确程度,除了要看误差的大小外,往往还要考虑该量本身的大小. 我们定义

$$r_{\bar{x}} = \frac{x - \bar{x}}{x} \tag{3.4}$$

为 $\bar{x}$ 的**相对误差**. 因为一个量的准确值往往是不知道的,因此常常将 $\bar{x}$ 的相对误差 $r_{\bar{x}}$ 定义为

$$r_{\bar{x}} = \frac{x - \bar{x}}{\bar{x}}. \tag{3.5}$$

一般说来,我们不能准确地计算出相对误差. 然而,像绝对误差那样,可以估计它的大小范围,即指定一个正数 δ,使得

$$|r_{\bar{x}}|\leqslant\delta.$$

我们称 δ 为 $\bar{x}$ 的一个**相对误差界**.

§4 舍入误差与有效数字

一般说来,一个实数的表示是无限的,例如,

$$\pi = 3.14159265\cdots,$$
$$e = 2.71828182\cdots,$$
$$\sqrt{2} = 1.41421356\cdots,$$
$$\frac{1}{3} = 0.33333\cdots.$$

因此,我们将一个实数 a 表示成无限小数的形式:

$$a = \pm\, a_0a_1\cdots a_m.\, a_{m+1}\cdots a_na_{n+1}\cdots, \tag{4.1}$$

其中 $a_i(i=0,1,\cdots)$都是 $0,1,2,\cdots,9$ 中的一个数字,且 $m\neq0$ 时,$a_0\neq0$.

通常,我们用四舍五入的规则去取一个无限小数的近似值,即若 a 的近似 $\bar{a}$ 取 $n-m$ 位小数时,有

$$\bar{a}=\begin{cases}\pm a_0a_1\cdots a_m.\,a_{m+1}\cdots a_n, & a_{n+1}\leqslant 4;\\ \pm\left(a_0a_1\cdots a_m.\,a_{m+1}\cdots a_n+10^{-(n-m)}\right), & a_{n+1}\geqslant 5.\end{cases}\tag{4.2}$$

此时

$$|a-\bar{a}|\leqslant\frac{1}{2}\times 10^{-(n-m)},\tag{4.3}$$

按此规律产生的误差称为**舍入误差**. 例如，$\pi=3.14159265\cdots$，取三位小数时，$\bar{\pi}=3.142$，$|\pi-\bar{\pi}|<\frac{1}{2}\times10^{-3}$；取五位小数时，$\bar{\pi}=3.14159$，$|\pi-\bar{\pi}|<\frac{1}{2}\times10^{-5}$. 又如，$x=-0.1354$，取二位小数时，$\bar{x}=-0.14$；取三位小数时，$\bar{x}=-0.135$.

假定有一个近似数

$$\bar{a}=\pm a_0a_1\cdots a_m.\,a_{m+1}\cdots a_n,\tag{4.4}$$

其中 $a_i(i=0,1,\cdots,n)$均是 0,1,…,9 的一个数字，且 $m\neq0$ 时，$a_0\neq0$. 若 $\bar{a}$ 的绝对误差满足(4.3)式，且 a_s 是 $\bar{a}$ 的第一位(自左至右)非零数字，则自 a_s 起到最右边的数字为止，所有的数字都叫做 $\bar{a}$ 的**有效数字**，并且说 $\bar{a}$ 是具有$(n+1-s)$位有效数字的**有效数**.

例 1 π 的近似数 $\bar{\pi}=3.1416$ 是具有五位有效数字的有效数，其中 3,1,4,1,6 均为 $\bar{\pi}$ 的有效数字.

例 2 $a=0.120443\cdots$的近似数 $\bar{a}=0.12044$ 是具有五位有 效数字的有效数，其中 1,2,0,4,4 都是有效数字.

例 3 $b=0.03473$ 的近似数 $\bar{b}=0.035$ 是具有二位有效数字的有效数，其中 3,5 为有效数字.

例 4 有效数 30.4 与有效数 30.40 不一样，后者有四位有效数字，而前者只有三位有效数字.

关于有效数字和相对误差的关系，我们有下面的定理.

定理 若形如(4.4)的近似数 $\bar{a}$ 具有 $n+1-s$ 位有效数字，则其相对误差有估计式

$$\left|\frac{a-\bar{a}}{\bar{a}}\right|\leqslant\frac{1}{2a_s}\times10^{-(n-s)},\tag{4.5}$$

其中 $a_s\neq0$ 是 $\bar{a}$ 的第一位有效数字.

证明 首先，若 $s=0$，此时 $a_0\neq0$，由(4.4)知

$$|\bar{a}|\geqslant a_0\times10^m.$$

再由(4.3)式有

$$\left|\frac{a-\bar{a}}{\bar{a}}\right|\leqslant\frac{1}{a_0\times10^m}\times\frac{1}{2}\times10^{-(n-m)}=\frac{1}{2a_0}\times10^{-n}.$$

其次，若 $s\neq0$，$a_s\neq0$，此时 $m=0$，由(4.4)知

$$|\bar{a}|\geqslant a_s\times10^{-s}.$$

再由(4.3)式有

$$\left|\frac{a-\bar{a}}{\bar{a}}\right|\leqslant\frac{1}{a_s\times10^{-s}}\times\frac{1}{2}\times10^{-n}=\frac{1}{2a_s}\times10^{-(n-s)}.$$

这个定理表明，一个近似数的有效数字愈多，其相对误差则愈小，因而精确度愈高. 就例

4，近似数 30.40 的精确度比 30.4 的精确度高. 据定理，前者的相对误差界为$\frac{1}{6}\times10^{-3}$，而后者的相对误差界为$\frac{1}{6}\times10^{-2}$.

§5 数据误差在算术运算中的传播

设$\bar{x},\bar{y}$分别是初始数据x,y的近似值，即

$$x=\bar{x}+e_{\bar{x}},\quad y=\bar{y}+e_{\bar{y}},$$

其中$e_{\bar{x}},e_{\bar{y}}$分别是$\bar{x},\bar{y}$的绝对误差. 我们来考察用$\bar{x},\bar{y}$分别代替x和y时，计算函数值

$$z=f(x,y)$$

产生的误差，即$\bar{z}=f(\bar{x},\bar{y})$的误差. 假定绝对误差$e_{\bar{x}},e_{\bar{y}}$的绝对值都很小，且$f(x,y)$可微，则$\bar{z}$的误差

$$e_{\bar{z}}=z-\bar{z}=f(x,y)-f(\bar{x},\bar{y})$$

可以近似地表示成

$$e_{\bar{z}}=\left(\frac{\partial f}{\partial x}\right)_{(\bar{x},\bar{y})}e_{\bar{x}}+\left(\frac{\partial f}{\partial y}\right)_{(\bar{x},\bar{y})}e_{\bar{y}},\tag{5.1}$$

而且

$$\begin{aligned}r_{\bar{z}}&=\frac{e_{\bar{z}}}{\bar{z}}=\frac{\bar{x}}{\bar{z}}\left(\frac{\partial f}{\partial x}\right)_{(\bar{x},\bar{y})}\frac{e_{\bar{x}}}{\bar{x}}+\frac{\bar{y}}{\bar{z}}\left(\frac{\partial f}{\partial y}\right)_{(\bar{x},\bar{y})}\frac{e_{\bar{y}}}{\bar{y}}\\&=\frac{\bar{x}}{\bar{z}}\left(\frac{\partial f}{\partial x}\right)_{(\bar{x},\bar{y})}r_{\bar{x}}+\frac{\bar{y}}{\bar{z}}\left(\frac{\partial f}{\partial y}\right)_{(\bar{x},\bar{y})}r_{\bar{y}}.\end{aligned}\tag{5.2}$$

从(5.1)式容易得到，进行算术运算(加、减、乘、除)时，初始数据误差和计算结果中产生的误差之间有下面关系：

(1) $f(x,y)=x\pm y$：

$$e_{\bar{x}\pm\bar{y}}=e_{\bar{x}}\pm e_{\bar{y}};\tag{5.3}$$

(2) $f(x,y)=xy$：

$$e_{\bar{x}\bar{y}}=\bar{y}e_{\bar{x}}+\bar{x}e_{\bar{y}};\tag{5.4}$$

(3) $f(x,y)=x/y$：

$$e_{\bar{x}/\bar{y}}=\frac{\bar{y}e_{\bar{x}}-\bar{x}e_{\bar{y}}}{\bar{y}^2}.\tag{5.5}$$

从(5.2)式，容易得到关系式：

(1) $f(x,y)=x\pm y$：

$$r_{\bar{x}\pm\bar{y}}=\frac{\bar{x}}{\bar{x}\pm\bar{y}}r_{\bar{x}}\pm\frac{\bar{y}}{\bar{x}\pm\bar{y}}r_{\bar{y}};\tag{5.6}$$

(2) $f(x,y)=xy$：

$$r_{\bar{x}\bar{y}}=r_{\bar{x}}+r_{\bar{y}};\tag{5.7}$$

(3) $f(x,y)=x/y$：

$$r_{\bar{x}/\bar{y}}=r_{\bar{x}}-r_{\bar{y}}.\tag{5.8}$$

从(5.5)式我们看到，在作除法运算时，若分母$\bar{y}$的绝对值很小，其计算结果的误差可能

很大. 因此，在数值计算中，要避免绝对值很小的数作分母. 又从(5.6)式看到，接近相等的同号近似数相减时，也将会使计算结果的误差变得很大. 当两个几乎相等的同号数相减时，精度的损失称为**相减相消**.

例 1 求方程

$$ax^2+bx+c=0, a\neq 0$$

的两个根 x_1, x_2 的二次公式是

$$x_1=\frac{-b+\sqrt{b^2-4ac}}{2a}$$

和

$$x_2=\frac{-b-\sqrt{b^2-4ac}}{2a}.$$

若 $b>0$ 且 $b^2>>4ac\geqslant 0$，计算 x_1 则会产生相减相消. 为了避免相减相消，我们可将分子有理化，把二次公式改写成

$$x_1=\frac{-2c}{b+\sqrt{b^2-4ac}}.$$

例 2 假定某一计算过程中，需要计算表达式 $1-\cos x$. 当 $x\simeq 0$ 时，直接进行计算会导致相减相消. 然而，我们可以利用恒等式

$$1-\cos x=2\sin^2\frac{x}{2},$$

计算 $2\sin^2\frac{x}{2}$ 来代替计算 $1-\cos x$.

§6 机器误差

6.1 计算机中数的表示

现在，我们假定提供给计算机的数 x 只是有限位小数. 这样，数 x 可以表示成

$$x=\pm 10^J\sum_{k=1}^{t}d_k10^{-k}, \tag{6.1}$$

其中 J 是整数，$d_1, d_2, \cdots, d_t$ 都是 $0,1,2,\cdots,9$ 中的一个数字. 例如

$$312.74=10^3(3\times 10^{-1}+1\times 10^{-2}+2\times 10^{-3}+7\times 10^{-4}+4\times 10^{-5}),$$
$$-0.012=-10^{-1}\times(1\times 10^{-1}+2\times 10^{-2}).$$

若记

$$a=\sum_{k=1}^{t}d_k10^{-k}=0.d_1d_2\cdots d_t, \tag{6.2}$$

则

$$x=\pm a\times 10^J. \tag{6.3}$$

例如

$$312.74=0.31274\times 10^3,\quad -0.012=-0.12\times 10^{-1}.$$

(6.1)或(6.3)式是通常的数的十进制系统计数法，其中的 10 称为十进制系统的**基数**.

在计算机中，还广泛采用二进制，八进制和十六进制系统表示数的方法，它们的基数分别为2，8和16.

一般地，一个 p 进制数 x 可以表示成

$$x = \pm\, p^J \sum_{k=1}^{t} d_k p^{-k}, \tag{6.4}$$

其中，$d_k(k=1,2,\cdots,t)$都是 $0,1,\cdots,p-1$ 中的一个数字. (6.4)式或写成

$$x = \pm\, a \times p^J, \tag{6.5}$$

其中

$$a = \sum_{k=1}^{t} d_k p^{-k} = 0.d_1 d_2 \cdots d_t. \tag{6.6}$$

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**，它用来确定该数的小数点的位置.

在各种计算机中，有各自规定的字长 t，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.

如果阶 J 是一个固定不变的常数，我们便称(6.4)或(6.5)为**定点**表示. 在定点表示中，通常取 $J=0$ 或 $J=t$，这就是说，将小数点固定在数的最高位的前面或者固定在最低位的后面. 若小数点固定在数的最高位前面，则计算机中参与运算的数的绝对值必须小于1.

如果阶 J 是可以变化的，我们则称(6.4)或(6.5)为**浮点**表示. 一个数可以有不同的浮点表示. 例如，5360可以表示成

$$0.5360 \times 10^4,$$

也可以表示成

$$0.0536 \times 10^5.$$

为了避免发生这种情形，我们将浮点表示规格化，规定在浮点表示中尾数的第一位数字 d_1 非零. 这种浮点表示的数，称为**规格化浮点数**. 从而，对于十进制规格化浮点数，其尾数满足关系式：

$$0.1 \leqslant a < 1;$$

对于二进制规格化浮点数，其尾数满足关系式：

$$\frac{1}{2} \leqslant a < 1.$$

按(6.4)规定的浮点数的全体组成的集合记作 F，再假定当 $x \neq 0$ 时，$d_1 \neq 0$，则称 F 为**规格化浮点数系**，它的基数为 p. 规格化浮点数系 F 是一个离散的有限集合. 例如，若 $p=2,t=3,L=1,U=2$，则相应的规格化浮点数系 F 中的规格化浮点数具有形式

$$x = \pm a \times 2^J = \pm 2^J (d_1 \times 2^{-1} + d_2 \times 2^{-2} + d_3 \times 2^{-3}),$$

其中当 $x \neq 0$ 时，$d_1=1$，d_2,d_3 为0，1中的一个数字，$J=-1,0,1,2$. 因此，d_1,d_2,d_3 只可能有下列四种排列：

d_1	d_2	d_3
1	0	0
1	0	1
1	1	0

$$1 \qquad 1 \qquad 1$$

从而尾数 a 只可能为下列 4 个数：

$$(0.100)_2=1\times2^{-1}+0\times2^{-2}+0\times2^{-3}=\frac{4}{8},$$

$$(0.101)_2=1\times2^{-1}+0\times2^{-2}+1\times2^{-3}=\frac{5}{8},$$

$$(0.110)_2=1\times2^{-1}+1\times2^{-2}+0\times2^{-3}=\frac{6}{8},$$

$$(0.111)_2=1\times2^{-1}+1\times2^{-2}+1\times2^{-3}=\frac{7}{8},$$

故 F 中有 33 个浮点数如下：

$J=-1$	$J=0$	$J=1$	$J=2$
$\pm\frac{1}{4}$	$\pm\frac{1}{2}$	±1	±2
$\pm\frac{5}{16}$	$\pm\frac{5}{8}$	$\pm\frac{5}{4}$	$\pm\frac{5}{2}$
$\pm\frac{3}{8}$	$\pm\frac{3}{4}$	$\pm\frac{3}{2}$	±3
$\pm\frac{7}{16}$	$\pm\frac{7}{8}$	$\pm\frac{7}{4}$	$\pm\frac{7}{2}$

以及数 0.

6.2 浮点运算和舍入误差

规格化浮点数系 F 是一个离散的有限集合. 在使用计算机进行数值计算时，初始数据可能不在 F 中，于是要用 F 中的数来近似地表示相应的数据. 如果初始数据或中间结果 x 的绝对值大于 F 中的最大正数，就会使计算机发生**溢出**现象. 因此，在以后的讨论中，我们将假定对给定的初始数据或中间结果不致于发生这种现象. 若 $x\in F$，我们要用 F 中最接近于 x 的浮点数 x_R 作为 x 的近似值. 这样，可按(4.2)四舍五入的规则取 x_R，即若

$$x=\pm a\times10^J, \tag{6.7}$$

其中

$$a=0.d_1\cdots d_td_{t+1}\cdots d_n,\quad 0\leqslant d_i\leqslant 9(i=1,\cdots,n),d_1>0, \tag{6.8}$$

则取

$$\bar{a}=\begin{cases}0.d_1-d_t, & \text{若 } 0\leqslant d_{t+1}\leqslant 4;\\ 0.d_1\cdots d_t+10^{-t}, & \text{若 } d_{t+1}\geqslant 5,\end{cases} \tag{6.9}$$

$$x_R=\pm\bar{a}\times10^J. \tag{6.10}$$

于是

$$\left|\frac{x_R-x}{x}\right|=\left|\frac{\pm\bar{a}\times10^J-(\pm a)\times10^J}{\pm a\times10^J}\right|$$
$$=\left|\frac{\bar{a}-a}{a}\right|.$$

据(4.3)式，并注意到 $a\geqslant10^{-1}$，便有

$$\left|\frac{\bar{a}-a}{a}\right|\leqslant\frac{1}{2a}\times10^{-t}\leqslant\frac{1}{2}\times10^{-t+1}=5\times10^{-t},$$

即有

$$\left|\frac{x_R-x}{x}\right|\leqslant5\times10^{-t}.$$

令

$$\frac{x_R-x}{x}=\varepsilon,$$

则得到关系式

$$x_R=x(1+\varepsilon),|\varepsilon|\leqslant5\times10^{-t}. \tag{6.11}$$

对于二进制系统数

$$x=\pm a\times2^J, \tag{6.12}$$

其中$\frac{1}{2}\leqslant a<1$，且

$$a=0.d_1\cdots d_td_{t+1}\cdots d_n,d_i=0\text{ 或 }1(i=2,\cdots,n),d_1=1, \tag{6.13}$$

则取

$$\bar{a}=\begin{cases}0.d_1\cdots d_t, & \text{若 } d_{t+1}=0;\\ 0.d_1\cdots d_t+2^{-t}, & \text{若 } d_{t-1}=1,\end{cases} \tag{6.14}$$

$$x_R=\pm\bar{a}\times2^J. \tag{6.15}$$

类似于十进制系统，我们有

$$x_R=x(1+\varepsilon),|\varepsilon|\leqslant2^{-t}. \tag{6.16}$$

有的计算机采用只“舍”而不“入”的断位方法. 此时，(6.11)和(6.16)式中 ε 的界的估计分别为

$$|\varepsilon|\leqslant10^{1-t}$$

和

$$|\varepsilon|\leqslant2^{1-t}.$$

采用浮点表示的数，怎样进行算术运算？进行两数相加(或相减)时，首先要**对阶**，即把两数的小数点对齐，使它们的阶相等. 对阶的方法是，把阶小的数的尾数右移，每移一位其阶就加“1”直到两数的阶相等. 然后，将对阶后的两数相加(或相减). 浮点数运算加，减分别记作$\oplus$，$\ominus$.

例 1 假设所使用的计算机是采用断位的方法，$t=5$，$p=10$. 若 $x=0.31249\times10^2$，$y=0.82718\times10^2$，则

$$x\oplus y=0.11396\times10^3.$$

例 2 假设所使用的计算机是采用舍入的方法，$t=5$，$p=10$. 若 $x=0.21062\times10^{-5}$，$y=0.12345\times10^{-3}$，则

$$x \oplus y = 0.00211 \times 10^{-3} + 0.12345 \times 10^{-3}$$
$$= 0.12556 \times 10^{-3}.$$

例 3 假设在 $p=10, t=3, L=U=5$ 的断位计算机上对数 0.0438, 0.0693 及 13.2 做加法运算，那么

$$(0.438 \times 10^{-1} \oplus 0.693 \times 10^{-1}) \oplus 0.132 \times 10^{2}$$
$$= 0.113 \times 10^{0} \oplus 0.132 \times 10^{2}$$
$$= 0.133 \times 10^{2},$$
$$(0.132 \times 10^{2} \oplus 0.693 \times 10^{-1}) \oplus 0.438 \times 10^{-1}$$
$$= 0.132 \times 10^{2} \oplus 0.438 \times 10^{-1}$$
$$= 0.132 \times 10^{2}.$$

由此可知，对于浮点运算，通常的运算规律不再成立. 准确和为 13.3131.

例 4 假设在 $p=10, t=4$ 的断位计算机上求数 $x=0.12378$ 与 $y=0.12362$ 的差，则有

$$x_R \ominus y_R = 0.1237 \times 10^{0} \ominus 0.1236 \times 10^{0}$$
$$= 0.1000 \times 10^{-3}.$$

x_R 的相对误差是

$$\frac{x - x_R}{x_R} = 6.467 \times 10^{-4},$$

y_R 的相对误差是

$$\frac{y - y_R}{y_R} = 1.618 \times 10^{-4}.$$

但 $x-y=0.00016$，$x_R \ominus y_R$ 的相对误差是

$$\frac{0.00006}{0.0001} = 0.6.$$

x_R 与 y_R 很接近，$x_R \ominus y_R = 0.0001$ 只有一位有效数字，产生了相减相消. $x_R \ominus y_R$ 的相对误差是参与运算的近似数的相对误差的 10^3 倍.

在作乘法运算时，就不必对阶.

现在，我们来考察计算机中浮点数的算术运算的舍入误差. 假设 x, y 都是规格化浮点数，$x, y \in F$. 我们用

$$fl(x+y), fl(x-y), fl(x \times y), fl(x/y)$$

分别表示得到准确的 $x+y, x-y, x \times y$ 和 x/y 后按上述舍入规则进行舍入的结果，即

$$fl(x+y) = (x+y)_R, \quad fl(x-y) = (x-y)_R,$$
$$fl(x \times y) = (x \times y)_R, \quad fl(x/y) = (x/y)_R.$$

就上述例 2，

$$x + y = 0.0021062 \times 10^{-3} + 0.12345 \times 10^{-3}$$
$$= 0.1255562 \times 10^{-3}.$$

因此 $x+y \in F$，而

$$fl(x+y) = (x+y)_R = 0.12556 \times 10^{-3}.$$

对于具有双精度累加器的计算机，这样的浮点运算是可以做到的.