数値の表現と誤差

■浮動小数点数 (floating point number)

現在最も普及している表現形式「IEEE754」の概略：

$$x_{\mathrm{f}} = \pm\left(\frac{1}{2^0} + \frac{x_2}{2} + \frac{x_3}{2^2} + \cdots + \frac{x_n}{2^{n-1}}\right) \times 2^m.$$

- 単精度 (single precision) = 4byte = 1+23+8bit：$n = 23+1,\ -126 \le m \le 127.$
  表現できる最小の正数$= 2^{-126} \simeq 1.175\times10^{-38}$, 最大の正数$\simeq 2^{127}\times2 \simeq 3.403\times10^{38}$.
  有効数字桁数：7 桁強 ($2^{24} = 10^{7.224}$ ゆえ).

- 倍精度 (double precision) = 8byte = 1+52+11bit：$n = 52+1,\ -1022 \le m \le 1023.$
  表現できる最小の正数$= 2^{-1022} \simeq 2.225 \times 10^{-308}$,
  最大の正数$\simeq 2^{127} \times 2 \simeq 1.798 \times 10^{308}$.
  有効数字桁数：16 桁弱 ($2^{53} = 10^{15.95}$).

(注) 実際にはもっと複雑. たとえば指数部の空き 2（例：$2^8 = 256 = (127-(-126)+1)+2$）は，0 や特殊な数の表現に予約されている.

■丸め (rounding)

浮動小数点数で表現可能な区間内の数 x に対して,

$$x_{\mathrm{f}} = x(1+\varepsilon_x), \qquad |\varepsilon_x| \le \varepsilon_{\mathrm{M}}$$

「マシンイプシロン」「マシンエプシロン」.

$\varepsilon_{\mathrm{M}} \sim 10^{-7}$（単精度），$\sim 10^{-16}$（倍精度）.

■誤差 (error) の種類

- 離散化誤差 (discretization error)：離散化による誤差
- 打ち切り誤差 (truncation error)：極限を有限で打ち切ることによる誤差
- 丸め誤差 (rounding error)：浮動小数点数で近似することによる誤差
- 桁落ち：近い数の引き算で発生する誤差
- 情報落ち（積み残し）：級数の計算などで発生する誤差

■区間演算 (interval arithmetic)

浮動小数点数をひとつ持つ代わりに，それの含まれる区間

$$x_{\mathrm{f}} \in X = [\underline{x}, \overline{x}]$$

を持ち，区間同士の演算を定義する．例：$X + Y = [\underline{x} + \underline{y}, \overline{x} + \overline{y}]$.
参考：「精度保証付き数値計算」＝浮動小数点演算で通常の数値計算をしつつ，同時にそれの含まれる区間も計算して，数値計算結果の精度を保証する.

■桁落ちとその例

x と y がほぼ等しいときの減算においては，仮数部の大部分が打ち消し合って $x-y$ の相対精度が著しく落ちる．この現象は**桁落ち**と呼ばれている．

例 1 2 次方程式 $ax^2+bx+c=0$ $(a\neq 0)$ の解 (根) は

$$\alpha_+=\frac{-b+\sqrt{b^2-4ac}}{2a},\qquad \alpha_-=\frac{-b-\sqrt{b^2-4ac}}{2a}$$

という公式で与えられる．$a=1$, $b=200$, $c=1$ のとき，真の値は
$\alpha_+=-0.0050001250062\cdots$, $\alpha_-=-199.99499987\cdots$ である．

ところが，2 進 24 桁 (0 捨 1 入), $\varepsilon=2^{-24}=0.596\cdots\times 10^{-7}$ で計算した結果の出力は，
$\tilde{\alpha}_+=-4.9972534\mathrm{E}-03$, $\tilde{\alpha}_-=-199.99501$ となる (ここで，E は 10 のべきを表す)．
α_+ の分子の減算において桁落ちが起きているため，その精度が著しく低下している．

$\alpha_\pm$ を精度良く求めるには，

$$b\geq 0\text{ のとき:}\quad \alpha_+=\frac{-2c}{b+\sqrt{b^2-4ac}},\qquad \alpha_-=\frac{-b-\sqrt{b^2-4ac}}{2a};$$

$$b<0\text{ のとき:}\quad \alpha_+=\frac{-b+\sqrt{b^2-4ac}}{2a},\qquad \alpha_-=\frac{-2c}{b-\sqrt{b^2-4ac}}$$

のように式変形 (**分子の有理化**) をしてから数値を代入すればよい．これにより桁落ちを回避できる．実際，この計算式を用いると，2 進 24 桁 (0 捨 1 入) 計算で，
$\tilde{\alpha}_+=-5.0001251\mathrm{E}-03$, $\tilde{\alpha}_-=-199.99501$ が得られる．

この例は，数学の世界で正しい公式でも丸め誤差のある数値計算の世界では別種の注意が必要であることを端的に示している．

■情報落ちとその例

絶対値の大きな数 x に絶対値の小さな数 y を足す計算 $x+y$ においては，演算結果を浮動小数点数に丸めたものが x に等しくなって，y の情報が反映されないことが起こる．これを**情報落ち**あるいは**積み残し**などと呼ぶことがある．

例 2 無限級数 $S=\sum_{k=1}^{\infty}1/k^2=\pi^2/6=1.6449340668\cdots$ の部分和 $S_n=\sum_{k=1}^{n}1/k^2$ を

$$S_n=\left(\cdots\left(\left(\frac{1}{1^2}+\frac{1}{2^2}\right)+\frac{1}{3^2}\right)+\cdots+\frac{1}{n^2}\right)$$

の形で 2 進 24 桁 (0 捨 1 入) で計算すると，$n\geq N=4096$ に対して $S_n=1.6447253$ となって増加しなくなる．これに対し，

$$S_n=\left(\cdots\left(\left(\frac{1}{n^2}+\frac{1}{(n-1)^2}\right)+\frac{1}{(n-2)^2}\right)+\cdots+\frac{1}{1^2}\right)$$

として小さい項から足すようにすると，精度のよい S の近似値を得ることができる．例えば，$S_{10000}=1.6448341$，$S_{100000}=1.6449241$，$S_{1000000}=1.6449330$ と計算される．

以上 (2013-10-07)