計算の品質保証 : 高精度・多倍長計算特集

渡部, 善隆
九州大学情報基盤研究開発センター

https://doi.org/10.15017/1470173

出版情報：九州大学情報基盤研究開発センター全国共同利用システム広報. 2 (3), pp.91-92, 2009-03. 九州大学情報統括本部広報委員会
バージョン：

計算の品質保証 – 高精度・多倍長計算特集 –

渡部 善隆*

たとえば，偏微分方程式を離散化して数値シミュレーションを行なう場合など，多くの科学技術計算では実数が用いられます．実数（以下，複素数でも同様です）は

$$\frac{1}{3} = 0.33333333333\ldots$$
$$0.1 = 0.00011001100110011\ldots \quad \text{(右辺は 2 進数表記)}$$

などと，一般に無限の桁を持ちます．一方，メモリが限られているコンピュータで扱うことのできる桁数は有限ですので，有限桁の数（もちろん実数です）によって無限桁の実数を近似することになります．この近似表現を「**丸め**」，また，丸めによって生じる誤差を「**丸め誤差**」と呼びます．現在の科学技術計算における実数計算のほとんどは，**浮動小数点演算**と呼ばれる近似計算で行なわれています．

残念ながら，浮動小数点数全体の集合は四則演算に関して閉じていません[1]．したがって，丸めおよび丸め誤差は，浮動小数点同士の加算や乗算などにおいても発生します．四則演算の結果は一般に実数であり，その結果をさらに浮動小数点数で近似する必要があるからです．

浮動小数点演算をコンピュータで行なう限り，丸め誤差は必然的に生じます．問題は計算結果における丸め誤差の影響の見きわめです．

＊＊＊＊＊＊

多くの実用上の問題では，桁数を多くとる（例えば単精度型を倍精度型に変える）こと，および丸め誤差の影響の少ない数値計算アルゴリズムを採用することによって，丸め誤差の影響は無視できると言われています．ただし，計算結果に深刻な丸め誤差の影響がないということ，すなわち数値計算結果の品質の保証のためには,（経験則ももちろん大切なものの）何らかの検証手段があるに越したことはありません．

極端な例をひとつ紹介します．2 元連立 1 次方程式

$$A\boldsymbol{x} = \boldsymbol{b}$$

を考えます．行列 A とベクトル $\boldsymbol{b}$ は次の通りです．

$$A = \begin{pmatrix} 64919121 & -159018721 \\ 41869520.5 & -102558961 \end{pmatrix}, \qquad \boldsymbol{b} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

*九州大学情報基盤研究開発センター　E-mail: watanabe@cc.kyushu-u.ac.jp

[1]有理数は閉じています．ただ，演算が進むと分母・分子がどんどん大きくなるため，科学技術計算全般への適用は困難です．

この解 $\boldsymbol{x}$ の値をスーパーコンピュータ tatara の Fortran 倍精度 (IEEE 標準 754 規格) により Cramer の公式

$$x_1 = a_{22}/(a_{11}a_{22} - a_{12}a_{21}), \quad x_2 = -a_{21}/(a_{11}a_{22} - a_{12}a_{21})$$

を用いて計算した結果 $\tilde{\boldsymbol{x}}$ は

$$\tilde{\boldsymbol{x}} = \begin{pmatrix} 102558961 \\ 41869520.5 \end{pmatrix}$$

となり，翻訳時・実行時のメッセージは何も表示されません．

また，同じ問題を MATLAB 7.6.0 の `A\b` を用いて求めた結果 $\hat{\boldsymbol{x}}$ は

$$\hat{\boldsymbol{x}} = \begin{pmatrix} 106018308.0071325 \\ 4328179.30017831 \end{pmatrix}$$

となります[2]．

ところが，真の解 $\boldsymbol{x}$ は正確に

$$\boldsymbol{x} = \begin{pmatrix} 205117922 \\ 83739041 \end{pmatrix}$$

となり，$\tilde{\boldsymbol{x}}$, $\hat{\boldsymbol{x}}$ は大きな誤差を持つことが分かります．

実は，この問題は，倍精度計算において「**桁落ち**」と呼ばれる精度の損失が起き，その影響が実行結果に深刻な影響を与えるように工夫して設定された意地悪なものです．したがって，このような現象が頻繁に起きていると**過度に心配される必要はありません**．ただ，浮動小数点演算の結果は必ずしも絶対的に信頼できるとは限らないということと，桁落ちは，（例のように）絶対値が近い数同士の減算[3]において発生しやすい，ということは心に留めておくとよいかもしれません．ちなみに，tatara における単精度型の計算では $a_{11}a_{22} - a_{12}a_{21}$ の計算が 0 になり，4 倍精度型の計算では $\boldsymbol{x}$ と同じ値が得られます．

＊＊＊＊＊＊

悪条件の問題に代表される丸め誤差の影響を受けやすい数値計算環境においては，多倍長演算に代表される高精度演算を用いることが解の妥当性・収束性を得るための有効な手段であることが知られています．センターでは，2008 年 8 月 8 日に，計算科学の分野で多倍長・高精度演算を必要とされている方の具体的な要望を汲み取るとともに，実装分野の方からも話題提供をいただきながら，それぞれの交流を図る場になればと，「先駆的科学計算に関するフォーラム 2008〜高精度・多倍長計算〜」を開催しました．詳細は以下を参照願います．

`http://www.cc.kyushu-u.ac.jp/scp/users/forum/program08-08.html`

今回，フォーラムで講演いただいた方から三つの記事を投稿していただきました．

- 4 倍精度, 多倍長精度演算使用例
- 素粒子物理学における多倍長計算
- PC クラスタでの多倍長計算の実装と利用

について，様々な考察・知見が展開されています．みなさまの参考になれば幸いです．

[2]Intel Core2 Extreme CPU X9650 で浮動小数点演算は tatara と同じ型・規格です．

[3]または絶対値が近い異符号数同士の加算．