計算機計算における累積丸め誤差の推定

田 村 英 之*

Estimation of the Accumulated Round-Off Error in Machine Computation

Hideyuki TAMURA

A simple method is proposed to estimate the accumulated round-off error in computer computation. The error may come from the complicated interaction between values that are to be numerically processed, such as added/subtracted and multiplied/divided. Severe relative error occurs, for example, when a number is subtracted from one of almost equal value. A set of formulae can estimate satisfactorily the limit as well as the standard deviation of errors, and the latter plays a central role in indicating an approximated (absolute) mean value of randomly distributed errors. The effectiveness of the algorithm is demonstrated by treating an example which contains a very complicated numerical processing of power series, such as addition/subtraction, division, development and reversion. Special formulae, which avoid problems of computer over/underflow, are proposed to estimate the standard deviation.

Key Words: Numerical Analysis, Computational Mechanics, Computer, Round-Off Error, Error Estimation, Processing of Power Series, Complete Elliptic Integral of the First Kind, q -Expansion

1. 緒 言

電子計算機を使用して数値計算を行う際に、(数値を処理する1語の)丸め誤差を中心に桁落ちや情報落ちなどが微妙複雑にからまった誤差の累積成長現象が問題となる⁽¹⁾⁽²⁾。簡単な計算では1語の長さ(バイト数)で決まる相対誤差(後述)程度であろうと考えても大過ないが、それでは安心できない場合も少なくない。そのために、プログラムの技巧や工夫の必要や楽しみがあり、誤差の大略を推定して計算結果の信頼性を吟味することもよく行われる。しかし現在のところ局所的、個別的な処理に終わることも多く、誤差の評価のための系統立った実用性の高い計算手順と言うものは今ひとつ未完成のように見受けられる。

一連の計算作業で逐次適用する四則演算の累積が増大すると、この誤差も急増する。別報⁽³⁾で述べた、べき級数の数値処理では、この悪影響がとくに深刻であり、誤差の程度を把握することなしには計算結果が無意味となる場合や、逆に、誤差の推定から計算結果の意味が正しく理解できる[厳密に0や整数値になる(らしい)]こともしばしば経験された。

そこで、四則演算の(主)計算およびこれと連動してその誤差を推定する(補助)計算を、一組の作業として定式化することの必要性が痛感される。本稿は、この意図を実現するための一つの習作である。

2. 誤差の推定

計算機の1語に収容される数値は、とくにその仮数部が(2進ないし16進の)有限桁数で処理される結果、必然的に丸め誤差⁽⁴⁾⁽⁵⁾を伴い、その値は、処理される数値の大きさに依存する一種の確率変数と考えられる。以下、被演算数を X, Y 、それぞれの厳密値を x_0, y_0 、誤差を $\varepsilon(X), \varepsilon(Y)$ 、また(一般に)変数/関数 F の誤差を $\varepsilon(F)$ と記すことにする。そこで

$$X = x_0 + \varepsilon(X), \quad Y = y_0 + \varepsilon(Y) \cdots \cdots (1)$$

に関する四則演算のうち、加減算に対しては、

$$\left. \begin{aligned} X \pm Y &= x_0 \pm y_0 + \varepsilon(X) \pm \varepsilon(Y), \\ \therefore \varepsilon(X \pm Y) &= \varepsilon(X) \pm \varepsilon(Y) \end{aligned} \right\} \cdots \cdots (2)$$

乗算に対しては、

$$\left. \begin{aligned} XY &= x_0 y_0 + x_0 \varepsilon(Y) + y_0 \varepsilon(X) + \varepsilon(X) \varepsilon(Y), \\ \therefore \varepsilon(XY) &= x_0 \varepsilon(Y) + y_0 \varepsilon(X) + \varepsilon(X) \varepsilon(Y) \end{aligned} \right\} \cdots \cdots (3)$$

である。 $\varepsilon(X), \varepsilon(Y)$ などが確率変数であるとすれば

* 原稿受付 平成2年6月1日。

*1 正員、九州大学工学部 (〒812 福岡市東区箱崎6-10-1)。

$\epsilon(X \pm Y)$, $\epsilon(XY)$ など確率変数となるから

$$\left. \begin{aligned} |\epsilon(X \pm Y)| &\leq \max|\epsilon(X)| + \max|\epsilon(Y)|, \\ |\epsilon(XY)| &\leq |x_0| \max|\epsilon(Y)| + |y_0| \max|\epsilon(X)| \\ &\quad + \max|\epsilon(X)| \max|\epsilon(Y)| \end{aligned} \right\} \dots\dots\dots (4)$$

などにより、演算結果の誤差の限界が与えられる。しかしこれらは、絶対に越える筈がない限界値を示しているから、常時発生する程度の誤差 [代表値あるいは(絶対値の) 平均値] と考えるのは明らかに非現実的な過大評価であろう。そこで、標準偏差ないし分散を誤差の代表値と見ることにすれば、定義⁽⁶⁾により、

$$\sigma^2(X \pm Y) = \sigma^2(X) + \sigma^2(Y) \dots\dots\dots (5)$$

$$\sigma^2(XY) = x_0^2 \sigma^2(Y) + y_0^2 \sigma^2(X) + \sigma^2(X) \sigma^2(Y) \dots\dots\dots (6)$$

ここに、 $\epsilon(X)$, $\epsilon(Y)$ は、各々平均値が 0 で互いに独立と仮定し、(一般に) F の $\epsilon(F)$ の分散を $\sigma^2(F)$ 、標準偏差を $\sigma(F)$ と表す。このような σ が、発生しやすい誤差の値と見られ、絶対値で表した誤差の平均値を近似すると思われる。式(4)、(6)中の厳密値 x_0, y_0 は(厳密には) 不明であるから、実際に計算された(誤差を含んでいる) X, Y で近似せざるを得ない。

除算については、誤差があまり大でなければ、逆数の乗算として処理すればよい。誤差が大であって、それを含む被演算数の値の分布が符号を変えずる範囲にまたがるような場合は、その逆数の誤差は分布の範囲が $-\infty \sim \infty$ と言うことであるから、誤差を論ずるより前に、この除算は精度的に意味がなく危険だとして中止するようプログラムを変更すべきである。

以上の式(4)~(6)を適当に利用すると、丸めの累積は無論のこと桁落ちや情報落ちなど一切を含めて四則演算に関する誤差の成長を追跡できそうである。なお初めて置数するときは 1 語の丸め誤差のみを考え、誤差限界に対しては、1 語の構造⁽⁴⁾⁽⁵⁾から決まる値、およそ(相対誤差) 10^{-5} (単精度)、 10^{-14} (倍精度)、 10^{-33} (四倍精度)とし、標準偏差のほうは、これらをやや下回る値(実験的に調整)とすればよいであろう。

3. 計算例=第 1 種完全だ円積分の q 展開⁽⁹⁾

非常に複雑で膨大な数値処理(べき級数の四則演算、反転などの逐次組み合わせ)を経た結果に対して、簡単な正解があると言う貴重な例題である。計算の性質は悪く、丸めの累積はもちろん桁落ちや情報落ちなど計算機にとっては精度的な無理難題が随所にランダムに数多く盛り込まれている格好のテスト問題である。

まず、第 1 種完全だ円積分、 $K(m)$ は、次のように定義/表示される。

$$\text{定義: } K(m) = \int_0^{\pi/2} d\theta / \sqrt{1 - m \sin^2 \theta} \dots\dots (7)$$

$$\left. \begin{aligned} \text{計算式 (0 型): } K(m) &= (\pi/2) \sum_{n=0}^{\infty} k_n^0 m^n, \\ k_0^0 &= 1, \quad k_n^0 = (1 - 1/2n)^2 k_{n-1}^0, \quad (n > 0) \end{aligned} \right\} \dots\dots (8)$$

$$\left. \begin{aligned} \text{計算式 (1 型): } K(1-m) &= \ln(4/\sqrt{m}) - \ln \sqrt{m} \sum_{n=1}^{\infty} k_n^0 m^n + \sum_{n=1}^{\infty} l_n^1 m^n, \\ l_n^1 &= k_n^0 l_n^0, \quad l_0^0 = \ln 4, \quad l_n^0 = l_{n-1}^0 - 1/n(2n-1) \end{aligned} \right\} \dots\dots\dots (9)$$

$$q \text{ の定義: } q = \exp[-\pi K(1-m)/K(m)] \dots\dots (10)$$

q の計算式 [(8), (9) $\rightarrow$ (10)]:

$$q = \frac{m}{16} \exp \left[\left(1 - \sum_{n=0}^{\infty} \frac{1}{k_n^0 m^n} \right) \ln 16 - 2 \sum_{n=0}^{\infty} \frac{l_n^1 m^n}{k_n^0 m^n} \right] \dots\dots\dots (11)$$

テスト計算の手順は次の通り:

- (i) 式(11)により、(単純なべき) 級数 $q = q(m)$ を作る (べき級数の除/加減算とべき級数展開)⁽³⁾。
- (ii) $q = q(m)$ の反転級数; $m = m(q)$ を作る (べき級数の反転)⁽³⁾。
- (iii) その級数 $m(q)$ を式(8)の右辺に代入して $K\{m(q)\}$ を作る (べき級数のべき級数展開)⁽³⁾。
- $K\{m(q)\} = (\pi/2)(1 + \sum_{n=1}^{\infty} k_n^0 q^n) \dots\dots\dots (12)$
- (iv) 別に、 $K\{m(q)\}$ の厳密表示は次のように与えられている⁽⁸⁾。これを変形して、式(12)と比較する。

表 1 累積誤差の(実測/推定) 計算例 [式(7)~(13)]

n (exact)	--- k_n^0 --- (computed)	$\epsilon \pm$	σ
0	1.00000000000000000000000000000000	0.50E-32	0.15E-33
1	0.00000000000000000000000000000000	0.12E-31	0.27E-33
2	4.00000000000000000000000000000000	0.96E-30	1.14E-31
3	0.7888800052210118051172856528279E-30	0.55E-28	0.37E-30
4	3.99999999999999999999999999999974	0.27E-28	0.81E-29
5	7.99999999999999999999999999999987	0.12E-27	0.7E-27
6	0.24253270025755272884102326652411E-26	0.51E-23	0.36E-24
7	0.20052654812081366156372264178E-25	0.21E-21	0.75E-25
8	3.99999999999999999999999999999984	0.85E-20	0.15E-23
9	3.99999999999999999999999999999982	0.34E-18	0.32E-22
10	7.99999999999999999999999999999992	0.13E-16	0.67E-21
11	0.258315048012616051158386270729E-19	0.53E-15	0.14E-19
12	0.159242194083808463744844301277E-18	0.21E-13	0.30E-18
13	8.00000000000000000000000000000000	0.82E-12	0.63E-17
14	0.12858189102587965300826218440E-16	0.32E-10	0.13E-15
15	0.1623701173314291440369561315828E-14	0.13E-08	0.29E-14
16	3.99999999999999999999999999999999	0.50E-07	0.61E-13
17	8.00000000000000000000000000000000	0.20E-05	0.13E-11
18	4.00000000000000000000000000000000	0.77E-04	0.28E-10
19	-0.24829205358400940895080564606250E-09	0.30E-02	0.61E-09
20	8.00000000000000000000000000000000	0.12	0.13E-07
21	0.926669917240123504887300000000E-07	4.7	0.29E-04
22	-0.394308242407166992187500000000E-05	0.18E-03	0.62E-05
23	0.989437103271483750000000000000E-04	0.72E-04	0.14E-05
24	-0.288118591308937500000000000000E-02	0.28E-06	0.30E-02
25	12.034041259765625000000000000000	0.11E-08	0.65E-01
26	6.32641601562500000000000000000000	0.33E-09	1.4
27	15.84765625000000000000000000000000	0.17E-11	31.
28	-627.500000000000000000000000000000	0.67E-12	0.69E+03
29	1232.000000000000000000000000000000	0.24E-14	0.19E+05
30	-490128.0000000000000000000000000000	0.10E-16	0.33E+06
31	8387677.0000000000000000000000000000	0.40E-17	0.74E+07
32	-7766016.0000000000000000000000000000	0.14E-19	0.14E+09
33	132024112.00000000000000000000000000	0.62E-20	0.34E+10
34	-6274643106.000000000000000000000000	0.24E-22	0.81E+11
35	1928524201984.0000000000000000000000	0.96E-23	0.18E+13
36	-10525622665216.00000000000000000000	0.38E-25	0.40E+14
37	-861204193771208.00000000000000000000	0.15E-27	0.89E+15
38	17538035096748032.000000000000000000	0.58E-28	0.20E+17
39	1040243552992362496.0000000000000000	0.23E-30	0.44E+18
40	2689564626634915840.0000000000000000	0.89E-31	0.99E+19
41	23895854354571082424.0000000000000000	0.35E-33	0.22E+21
42	-695021615373536915968.00000000000000	0.14E-35	0.49E+22
43	15802161147354266271744.000000000000	0.54E-36	0.11E+24
44	6521524702160226663530496.0000000000	0.21E-38	0.25E+25
45	2490219428347665077272576.0000000000	0.83E-39	0.55E+26
46	-100812085846072900572133952.00000000	0.33E-41	0.12E+28
47	22204760763738851331696367616.0000	0.13E-43	0.27E+29
48	-1033297012244294106721567575424.0	0.50E-44	0.60E+30
49	1910203070115635936839382224896.	0.20E-46	0.13E+32
50	-28547514165920445904574637670E+33	0.78E-47	0.30E+35
51	0.155284691693785487828947193021E+35	0.30E-49	0.67E+34
52	-0.1698935985821247499014881381607E+36	0.12E-51	0.15E+36
53	0.86310499159181819128040267190E+37	0.47E-52	0.35E+37
54	0.833628711553575571469974218007E+37	0.18E-54	0.75E+38
55	0.14763821914507248985671954425402E+40	0.72E-55	0.17E+40
56	-0.242792500142745913482439578501E+41	0.28E-57	0.38E+41
57	0.495196765998380196668069764664E+42	0.11E-59	0.85E+42
58	-0.467810586062616798965147464239E+44	0.44E-60	0.19E+44
59	0.242792500142745913482439578501E+45	0.17E-62	0.45E+45
60	0.30807925022920205613336405319120E+47	0.67E-63	0.97E+46
61	-0.5377138275423812176675016034561E+48	0.26E-65	0.22E+48
62	0.3350077862194823786554992753575E+49	0.10E-67	0.49E+49
63	0.9598778234018310737093967992581E+50	0.41E-68	0.11E+51
64	0.40308835341310273620687193794640E+52	0.16E-70	0.25E+52

図1 累積誤差の(実測, 推定)計算例(表1対応)

$$K\{m(q)\} = (\pi/2)(1 + 2 \sum_{n=1}^m q^{n^2})^2 \quad \dots\dots\dots (13)$$

係数 k_n^q を, $n=0 \sim 64$ について数値計算した結果と厳密値との比較を表1と図1に示した。その際, 式(4)による誤差限界と式(5), (6)による誤差の標準偏差, それぞれの推定計算値を $\epsilon_{\pm}$, σ^* と記して併記している。なお計算機の語長は四倍精度を使用しており, 数値を初期置数する際の $\epsilon_{\pm}$, σ^* (の初期値) は, それぞれ, 相対誤差 10^{-33} , 3×10^{-35} とおいた。

すべての, べき級数演算の計算処理は, 次数番号の小さな項から大なる方へと逐次進行するから, ここに示す結果についても, 次数の大なる項のほうが, 多数の演算累積を経た計算結果と見て良い, このことを反映して, 項の次数の増大とともに, 実測/推定いずれの誤差も急増し, しかも膨大な値に達していることがわかる。図2は, 実測誤差を σ^* で除した結果であり, σ^* による誤差推定の有効性を示している。

以上の結果から, 標準偏差による誤差の推定は, 実用上十分有効に機能すると考えられ, 実際に生起する誤差は(例外的な0を除いて), 推定値 σ^* の $1/10 \sim 10$ 倍程度と見ればよいようである。反面, 誤差限界の推定値 $\epsilon_{\pm}$ のほうは(確かに誤差の上限を示しているが) やや利用価値が乏しいように思われる。

4. 標準偏差(ノルム)の計算

誤差限界の計算, 式(4)は問題ないとして, 標準偏差のほうは(σ ではなく)分散 σ^2 の形で一貫処理することにより, 量質ともに式(4)とほぼ同等の作業となる(毎回 σ の自乗/平方根を繰り返すのは自明な無駄)。しかし σ^2 のレベルで処理するのはアンダフロー/オーバーフローが起こり易い⁽⁷⁾ 難点がある。

図2 累積誤差の(実測, 推定)計算例[表1対応, (実測誤差)/(標準偏差の推定値)]

これら問題の解決策として, 式(5), (6)におけるノルム形の計算を次のアルゴリズムで処理することを提案しておきたい[式(6)では, これを2度使う]。 a , b とも非負とし(自明の) $a+b=0$ の場合を例外として,

$$\sqrt{a^2 + b^2} = (a+b) \sqrt{1 - 2ab/(a+b)^2} \quad \dots\dots\dots (14)$$

平方根の計算は, 最良近似多項式⁽¹³⁾により, 精度低下との妥協で若干の高速化が期待できる。たとえば,

$$\begin{aligned} \sqrt{a^2 + b^2} &\approx (a+b) \\ &\times [1 - \{1.005277 + (0.391499 + 1.09114y)y\}y], \end{aligned} \quad \left. \begin{aligned} \text{ただし } y &= ab/(a+b)^2 \quad (0 \leq y \leq 1/4), \\ \text{相対誤差 } &\pm 0.008\% \quad (y=0 \text{ で誤差 } 0) \end{aligned} \right\} \quad \dots\dots\dots (15)$$

同様の1次式では相対誤差0.8%, 2次式で0.08%である。

5. 結 語

計算機(定語長)計算に特有の現象である“1語の丸めに起因する誤差の累積成長”を予測し, 計算結果の信頼性を推定するためのアルゴリズムを提案し, 例題によって, その妥当性を検討した。なお, この例題は, 累積丸め誤差の恐ろしさを如実に示すデモンストレーションとして, またそれを推定評価するアルゴリズム/プログラムのテスト問題として貴重であろう。

本稿は(緒言でも触れたように)別報⁽³⁾の, べき級数演算の数値処理作業に対する副作業の定式化を意図したものであるが, 着想開陳の域を遠く出ないものとなってしまう。紹介した例題に関するテストはますますの出来ばえと自賛したいが検討を要する問題も多く潜在しているに違いないから, さらに多くの数値実験や例題試算を重ね改良を続けることにしたい。

なお, この副計算プログラムは, すでに著者らによる報文二, 三^{(10)~(12)}でも試験的に使用しており, その

ほうもそれなりの成績と見られるが、多少懐疑的な結果⁽⁷⁾もないわけではない。さらに、この副作業に要する計算時間とプログラム量は、主計算のその数十倍程度になること⁽⁷⁾も宿命的問題であろう。これら検討事項のために続報を予定しておきたい。

文 献

- (1) 高田, 機械計算法, (1981), 7-10, 養賢堂.
 - (2) 伊理, 数値計算の常識, (1988), 1-25, 共立出版.
 - (3) 田村, 機論, 56-531, C (1990), 3154.
 - (4) 文献(1)の4ページ.
 - (5) 文献(2)の3ページ.
 - (6) 機械工学便覧, (1986), A2-57/B3-33.
 - (7) 田村・深田, 九大工学集報, 61-1, (1988), 45.
 - (8) 友近, 楕円関数論 (復刻版), (1973), 135/239, 現代工学社.
 - (9) 田村・深田, 九大工学集報, 61-2, (1988), 149.
 - (10) 田村・李, 機論, 55-509, C (1989), 13.
 - (11) 田村・李, 機論, 55-512, C (1989), 823.
 - (12) 田村・李, 機論, 56-525, C (1990), 1155.
 - (13) 赤坂, 数値計算, (1969), 107, コロナ社.
-