数理解析研究所講究録
第153巻 1972年 152-175

有限桁計算における計算誤差と計算限界について

日本大学理工　山下眞一郎

目　次

§0 序

通常，数値計算は有限桁で演算を行う。　このために，必然的に，計算結果の精度は有限である。　そして演算桁数よりも少ない桁数しか計算結果は正しくない。　さらに，公式誤差，入力誤差，算法誤差*などを除く，いわゆる計算誤差しか起り得ない計算に於ては，演算桁数を増大しても，損失桁数**は一定であり，計算結果の精度は，ほぼ，演算桁数に比例することが経験される。　この種の計算の損失桁数は問題と算法(あれ)***に対して固有な量であり，問題と算法が決まれば定まる。

そして，算法が損失桁数に関与しない範囲に於て，算法を工夫しても，計算結果の精度は改善されず，演算桁数の増大以外に精度は改善されない。　計算誤差は主に丸めと桁落ちによって起きる。　前者は演算回数に対して，あまり急激に有効桁数を消失せず，後者は演算回数に関係なく有効桁数を消失する。　このようなわけで，有限桁演算における損

注.

*　例えば，級数計算で，収束しないうちに，計算を打切るための誤差のことである。　このときは，演算桁数を増しても，結果はよくならない。

**　誤差の混入する桁数のことである。

***　計算法と演算の順序の意味である。

失桁数の把握は重大である。　損失桁数の把握は，帰する所，誤差限界を知ることであり，誤差限界を知ることは，また，ある数を他の数と識別する限界を知ることである。　この論文は相対零の概念を導入し，その上限が数の識別限界であることから，相対零の上限を評価して，演算精度，損失桁数，計算誤差などを定量的に定める方法について論ずる。

§1.　絶対零と相対零

普通，零の定義は

(1·1)　$a+b=a;\quad a-b=a$

に於ける b のように，ある値に，加えても，引いても，その値を変えないものとなっている。　このように，従来，零と呼ばれていたものを **絶対零** と呼ぶ。　絶対零の定義は無限桁演算に於ける零の定義であるが，同様に，有限桁演算に於ても，零を定義することができる。　演算を M 進法 L 桁で行う（有限桁演算とは，すべてそのようなものとし，数値例の時は $M=10$ とする）とすれば，(1·1) 式を満す b は次のような大きさを持つ。

(1·2)　$|b| \leqq k * M^{n-L}$

但し，k は **丸め係数** で，例えば，切り捨て，切り上げ演算では 1，$(M/2-1)$ 捨 $(M/2)$ 入 —— 10進法では4捨5入 —— では $1/2$ である。また，$M^{n-1} \leqq |a| < M^n$ とする。

(1.2)式の範囲の値を，M進法L桁の，kを定めるような演算で，aに加えても，aから引いても，その結果はaであるから，零と同じである。　そこで，このような範囲の値をaに対する **相対零** と呼び，絶対零と区別する。

任意の数に絶対零を乗ずると，その結果は常に絶対零になるが，有限な数の演算に於ても，同様に，有限な数の絶対値最小をMINとすれば，aに対する零.bは $|b|<MIN/|a|$ を満す数になる。　これを満す概数bをaに乗ずれば，その結果はいつでも絶対零になる。　このような零の定義によって，有益な結果を導くことができるかも知れないが，ここで論ずるのは，有限演算ではなく，有限桁演算であることに注意されたい，取り扱う数の大きさは制限しないのである。　その理由は，有限の精度で，ある数を他の数と識別するには，加減算に対する零の要請だけで十分だからである。

(1.2)式で定まる|b|の上限 $k*M^{n-L}$ はaの末尾の1単位の大きさに，丸め係数を乗じたものであるが，計算誤差を論ずるのに，末尾の1単位の大小を問題にするほど厳密に考えても実用的な意味がないので，aが直接含まれて，議論が進め易い，次式でこれを代用する。

(1.3)　(aの相対零の上限) $= |a|*M^{-L}$

以下これを $\Delta(a)$ と略記する。

相対零の上限は，ある数と他の数を区別するとき，相対零以上違う数だけが区別できて，2数の差の絶対値が相対零の上限以下となれば，区別がつかないと言う認識の限界を示す量である。　そこで，評価すべき数式の相対零の上限がわかれば，それはその数式の誤差限界となり，その数式の値とから，その数式及び算法の損失桁数がわかり，計算の限界が判明する。

§2. $\sum_{k=1}^{n} x_k$

相対零は加減算に対するものであるから，$\sum_{k=1}^{n} x_k$ の形に帰着する数式が考察の対称となる。

n 個の変数 x_k の和の計算順序を次のように規定する。

$$(2.1)\quad y_0 = 0\ ;\quad y_k = y_{k-1} + x_k\ ;\quad k=1,2,\cdots,n$$
$$y = y_n \equiv \sum_{k=1}^{n} x_k$$

このように，計算順序を指定すれば，和 y の相対零の上限 $\Delta(y)$ は，各加算の段階の相対零の上限の最大を取ればよいから，次のように表わせる。

$$(2.2)\quad e_0 = 0\ ;\quad e_k = \max\left(e_{k-1}, |y_k|, |x_k|\right)\ ;\quad k=1,2,\cdots,n$$
$$\Delta(y) = e_n \times M^{-L}$$

e_n の計算が複雑であるから，2つの略算を加え，3つの形式を定義しよう。

E_I 形式：　$E_I = \max(|x_1|, |x_2|, \cdots, |x_n|)$

E_{II} 形式：　$E_{II} = \sum_{k=1}^{n} |x_k|$

E_{III} 形式：　$E_{III} =$ (2.2)式で定義される e_n)

これらの間には，$\frac{1}{n}E_{II} \leqq E_I \leqq E_{III} \leqq E_{II} \leqq n E_I$ の関係がある。従って，e_n の代りに E_I を用いれば，最悪の場合、$\frac{1}{n}$ 倍の過小評価となり，E_{II} を用いれば，n 倍の過大評価となり得る。また，見方を変えれば，計算順序を最良に選ぶ場合とそうでない場合，結果の誤差は最悪のとき，n 倍の相違となることを示している。

$y = \sum_{k=1}^{n} x_k$ のM進法での正しい桁数と認識できる上限は $\Delta(y)$ が，y の何桁目に影響するかで定まるから，次式で算定される。

(2.3)　(y の正しい桁数の上限) $= \log_M |y/\Delta(y)|$

ここに，　$= L - \alpha$

(2.4)　$\alpha = \log_M |e_n/y|$

α は演算桁数に依存せず，x_k と加算順序に依存する量であ

り，いわゆる桁落ちの桁数である。　これを(2·3)式のように，演算桁数Lから正しい桁数が消失する桁数だから**損失桁数**と呼ぶことにする。　(2·4)式で，e_nが絶対零の場合には$\chi_k \equiv 0$だから，誤差は入らない。　また，αがLに止ずいたり，$y=0$のときは，yの正しい桁数がないと解釈する。以後に現われる損失桁数の意味はこのような修正を受けるものとする。

$\sum_{k=1}^{n}\chi_k$の相対零の上限は誤差の上限そのものであるが，次のような説明をすれば，直感的な理解が得られる。　すなわち，累算器を持つM進法L桁の計算機を考え，$\sum_{k=1}^{n}\chi_k$はこの累算器に逐次加算されると考える。　そして，χ_kは末尾の1単位程しか誤差を有せず，2数の加算のときは大きい数に桁揃えが行なわれるために，大きい数の末尾の1単位程度が誤差と考え，結果に桁上りがあれば，L桁に丸められ，その末尾の1単位程度が誤差と考える。　それらの誤差の中の最大が$\sum_{k=1}^{n}\chi_k$の誤差の上限であり，相対零の上限である。　$\sum_{k=1}^{n}\chi_k$を精度よく計算するには，e_nを最小になるような加算順序を選ばなければならない。

ある数，例えばAをM進法で$M^{n-1} \leqq |A| < M^n$とすれば，Aの末尾の1単位はM^{n-L}であるけれども，計算式を簡単にし，見通しをよくするために，これを$|A| * M^{-L}$としても，ああ

よその計算としてはさしっかえないだろう。 特に，$M=2$ の2進法の計算機では問題が少ない。 $M^{n-1} \leqq |A| < M^n$ の誤差が M^{n-L} であると言うことは，相対誤差 ε が，$M^{1-L} \geqq \varepsilon > M^{-L}$ となって，数値部の(最初の数字が)小さい数値の相対誤差は大きくなる。 近頃主流の $M=16$ の16進法の計算機の誤差のふるまいは複雑ではないかと思われる。

§3. 入力データが誤差を含む時の考え方

x_k が末尾の1単位程度しか誤差を含まず，残りは計算中の誤差だけとすれば，(2.3)式は $\sum_{k=1}^{n} x_k$ の正しい桁数に対して，よい評価を与える。 もし，x_k が Δx_k の誤差を含めば，A_k, B_k を x_k と同符号に取って，

$$(3.1) \qquad A_k - B_k = x_k \; ; \; |\Delta x_k| = |A_k| * M^{-L}$$

を満すように，x_k を A_k と B_k の2数に分解して考えればよい。 入力データが誤差を含むときの誤差解析は実用上興味ある問題であるが，直接入力誤差を導入して議論すると，説明が複雑になるので，この論文ではこの程度にとどめる。 この問題で注意を要するのは，x_k が独立でなく，$\sum_{k=1}^{n} \Delta x_k$ が消失する，いわゆる 誤差の桁落ち する問題があることである。

このようなとき，誤差を過大に見積ることになるから，注意せねばならない。例えば，3次方程式をCardano法で解くとき，判別式が零に近く，ほとんど誤差のとき，$\sqrt[3]{A+\varepsilon}+\sqrt[3]{A-\varepsilon}$ のような計算が現われる。ε が A の中ほどの桁に影響する大きさであると，$\sqrt[3]{A+\varepsilon}$，$\sqrt[3]{A-\varepsilon}$ のそれぞれは，中ほどの桁に誤差を含むことになる。このとき，$\sqrt[3]{A+\varepsilon}+\sqrt[3]{A-\varepsilon}$ の精度は半分しかないのではなく，ほぼ全桁正しい。それは，$\sqrt[3]{A+\varepsilon}$ と $\sqrt[3]{A-\varepsilon}$ の誤差は独立ではなく，和を取ると打消し合うように働くからである。

§4 乗除算

χ_n が乗除算で

$$\chi_n = \chi_{n1} \textcircled{\scriptsize *1} \chi_{n2} \textcircled{\scriptsize *1} \cdots\cdots \textcircled{\scriptsize *1} \chi_{nm} \tag{4.1}$$

但し $\textcircled{\scriptsize *1}$ は乗算または除算を表わす.

のように表現されていても，χ_n の相対誤差は χ_{ni} の各々の相対誤差の和の程度を越えないから，χ_{ni} が末尾の1単位程度の誤差を持つならば，χ_n は高々末尾の m 単位程度の誤差を持つ．宇野先生の研究*によれば，χ_n は通常あまり大きな誤差にならず，$\sqrt{m}$ に比例する程度である。このように，乗除算によって，有効数字を大きく失うことはなく，加減算によってのみ有効数字を大きく失うので，$\sum_{i=1}^{n}\chi_n$ の形が重要な意味を持つことがわかる。

(注) * 宇野利雄：“数値計算論” 岩波書店(昭和16年)；“数値計算” 朝倉書店(昭和38年)

§5　行列の和，差，積和，行列の積

行列A，Bの和,差をCとし，それらの要素をa_{ij}, b_{ij}, c_{ij}とすれば，（A,B,Cはn行m列とする）

(5.1)　$c_{ij} = a_{ij} \pm b_{ij}$; $i=1,2,\cdots n$; $j=1,2,\cdots,m$

であるから，これは2数の和，差と同じである．

2n個の数a_k, b_kの積和

(5.2)　$y = \sum_{k=1}^{n} a_k b_k$

は，$x_k = a_k b_k$と置けば，$\sum_{k=1}^{n} x_k$と同じである．

n行l列の行列$A=(a_{ik})$とl行m列の行列$B=(b_{kj})$の積ABがn行m列の行列$C=(c_{ij})$であるとき，

(5.3)　$c_{ij} = \sum_{k=1}^{l} a_{ik} b_{kj}$; $i=1,2,\cdots,n$; $j=1,2,\cdots,m$

であり，C=ABの要素は積和のそれと同じように評価できる．

§6　逆行列

n次の正規行列をAとし，Aの逆行列の近似値を$\tilde{A}^{-1}$とする．また，$\tilde{A}^{-1}$がかなりよい近似値であるとすれば，

(6.1)　$A\tilde{A}^{-1} = E + e$

但し，Eは単位行列，eの要素e_{ij}は$|e_{ij}| \ll 1$

この式に左から $\tilde{A}^{-1}$，右から $(E+e)^{-1}$ を乗じて

$$(6.2)\qquad \bar{A}^{-1} = \tilde{A}^{-1}(E+e)^{-1}$$
$$= \tilde{A}^{-1} - \tilde{A}^{-1}e + \tilde{A}^{-1}e^2(E+e)^{-1}$$

となり，e^2 の項を無視すれば

$$(6.3)\qquad \bar{A}^{-1} \fallingdotseq \tilde{A}^{-1} - \tilde{A}^{-1}e$$

である．　従って，$\tilde{A}^{-1}$ の精度は $\tilde{A}^{-1}e$ によって判断できる．即ち，別に述べた行列の積に関する考察から，$\Delta(A\tilde{A}^{-1}-E)$ を e と考えればよいから，これに $\tilde{A}^{-1}$ を左から乗ずれば，$\tilde{A}^{-1}$ の精度がわかる．　e の要素の符号は不明であるので，$\Delta(A\tilde{A}^{-1}-E)$ の要素の符号を適当に取った 2^{n^2} 個の行列に $\tilde{A}^{-1}$ を左から乗ずれば，$\bar{A}^{-1}$ を中心とした誤差領域を与える．　しかし，2^{n^2} 個の行列を考えるのは大変だからか，$\tilde{A}^{-1}e$ の計算をするときの積和の各要素で推定する以外にないようである．

§7　連立一次方程式

n 元連立一次方程式

$$(7.1)\qquad \sum_{j=1}^{n} a_{ij}x_j = c_i \;;\; i=1,2,\cdots,n$$

の根 x_j の近似根 $\tilde{x}_j$ を求めて，その1つ1つの精度を判定する．まず，残差である次式の相対零の上限を調べる．

$$(7.2)\qquad y_i = \sum_{j=1}^{n} a_{ij}\tilde{x}_j - c_i \;;\; i=1,2,\cdots,n$$

そのために，計算順序を次のように定める．

(7.3) $f_{i0}=0;\ f_{ik}=f_{ik-1}+a_{ik}\tilde{x}_k;\ k=1,2,\cdots,n$

$$y_i=f_{in}-c_i;\ i=1,2,\cdots,n$$

そうすると，y_iの相対誤の上限$\Delta(y_i)$は次のように表わせる．

(7.4) $e_{i0}=0;\ e_{ik}=\max\left(e_{ik-1},|f_{ik-1}|,|a_{ik}\tilde{x}_k|\right);\ k=1,2,\cdots,n$

$$\Delta(y_i)=\max\left(e_{in},|c_i|,|y_i|\right)*M^{-L};\ i=1,2,\cdots,n$$

従って，y_iの正しい桁数は次のように表わせる．

(7.5) (y_iの正しい桁数の上限) $=\log_M\left(|y_i|/\Delta(y_i)\right)$

ここに，$=L-\alpha_i$

(7.6) $\alpha_i=\log_M\left(\max\left(e_{in},|c_i|,|y_i|\right)/|y_i|\right)$

$\Delta(y_i)$はn次元残差空間の上限の1点を与える。$\Delta(y_i)$の要素の符号を考慮した2^n個の点は残差の上限領域——残差がこの領域内に入れば，残差をこれ以上小さくする目安がなくなる領域——を与える。従って，係数行列Aの逆行列をA^{-1}とし，残差の上限領域を表わす2^n行n列の行列をRとすれば，$A^{-1}R$はx_sを中心とした解の認識の限界領域を表わす。nが小さいと，この領域が求められるが，nが大きいと，2^n個のベクトルを考えるのは大変で，個々の要素の上限しか求められないだろう。しかし，強調しておきたいのは，x_sがあまり正確でなくとも，このような領域は求められることである．

計算誤差の把握が直接的に役立つ良い例は反復計算法に於て収束を判定するときである。n元連立一次方程式に於て

よく使われる そのような方法は Gauss－Seidel法であろう。 すなわち，

$$(7\cdot7)\quad \begin{cases} x_i^{(k)} = x_i^{(k-1)} + \Delta x_i^{(k)} \;;\; \Delta x_i^{(k)} = y_i^{(k)}/a_{ii} \\ y_i^{(k)} = c_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i}^{n} a_{ij} x_j^{(k-1)} \\ \text{但し} \quad x_i^{(0)} = 0 \;;\; i = 1,2,\cdots,n \;;\; k = 1,2,3,4,\cdots \end{cases}$$

に於て，$y_i^{(k)}$ のすべての値が相対零となるとき，Gauss－Seidel法は収束と考えるべきである。 それ以外は発散か，もしくは反復回数が不足していることになる。 収束判定の実際的方法は，相対零の上限が過大に評価されるE型形式である次式を使い，すべての式が $|y_i^{(k)}| \leqq \Delta(y_i^{(k)})$ となった時収束とみなすと良いだろう。

$$(7\cdot8)\quad \Delta(y_i^{(k)}) = \left\{ |c_i| + \sum_{j=1}^{i-1} |a_{ij} x_j^{(k)}| + \sum_{j=i}^{n} |a_{ij} x_j^{(k-1)}| \right\} * \bar{M}^{-L} \;;\; i = 1,2,\cdots,n$$

その時，解の限界は $\Delta(x_i^{(k)}) \leqq |\Delta(y_i^{(k)})/a_{ii}|$ となるから，

$$(7\cdot9)\quad (x_i \text{ の正しい桁数の上限}) = \log_M \left| x_i^{(k)} / \Delta(x_i^{(k)}) \right|$$

ここに，

$$= L - \alpha_i$$

$$(7\cdot10)\quad \alpha_i = \log_M \left(\left(|c_i| + \sum_{j=1}^{i-1} |a_{ij} x_j^{(k)}| + \sum_{j=i}^{n} |a_{ij} x_j^{(k-1)}| \right) / |x_i^{(k)} a_{ii}| \right)$$

この α_i は大きめの値であるが，根の近傍に於ては高々 $\log_M(n+1)$ 程度である。

§8 多項式

n次の多項式

(8.1) $\quad f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$

の値は $Z_k = a_k x^k$ と置けば，$\sum_{k=0}^{n} Z_k$ となって、和の形に帰着する。計算の順序を次のように規定しよう。

(8.2) $$\begin{cases} f_{n+1} = 0; \quad Z_k = a_k x^k; \quad f_k = f_{k-1} + Z_k; \quad k = n, n-1, \cdots 1, 0 \\ f(x) = f_0 \end{cases}$$

このような計算に対する $f(x)$ の相対誤の上限は次のように表わせる。

(8.3) $$\begin{cases} e_{n+1} = 0; \quad e_k = \max(e_{k+1}, |f_{k+1}|, |Z_k|); k = n, n-1, \cdots, 1, 0 \\ \Delta(f(x)) = e_0 * M^{-L} \end{cases}$$

従って，$f(x)$ の正しい桁数は次のようになる。

(8.4) $\quad (f(x)$ の正しい桁数の上限$) = \log_M |f(x)/\Delta(f(x))|$

ここに， $\quad = L - \alpha$

(8.5) $\quad \alpha = \log_M |e_0/f(x)|$

通常，(8.2)式のような計算回数の多い方法は取らないで，次のような手順による。

(8.6) $$\begin{cases} f_{n+1} = 0; \quad f_k = a_k + x f_{k+1}; k = n, n-1, \cdots, 1, 0 \\ f(x) = f_0 \end{cases}$$

このような計算手順に対する相対誤の上限は次のように表わせる。

(8.7) $$\begin{cases} e_{n+1} = 0; \quad e_k = \max(|x e_{k+1}|, |a_k|, |x f_{k+1}|, |f_k|); k = n, n-1, \cdots, 1, 0 \\ \Delta(f(x)) = e_0 * M^{-L} \end{cases}$$

これらの相対誤の上限の評価は多少複雑であるから，過大評価（但し高々 $(n+1)$ 倍）となる次式を使うとよい。

(8.8) $$\Delta(f(x)) = \sum_{k=0}^{n} |a_k x^k| \times M^{-L}$$

これは.

(8.9) $$\left|\sum_{k=0}^{n} a_k x^k\right| \leqq \sum_{k=0}^{n} |a_k x^k| * M^{-L}$$

を満足するとき，M進法L桁の演算では$f(x)$が零であると見做すべきことを意味する。多項式の零点を求めるとき，この右辺は過大に評価してあるので，零点の近傍で，これを満足するようなxが必ず求められ，条件が厳しいためにloopするようなことにならない。

(8.8)式は(8.2),(8.6)式にそれぞれ対応して

(8.10) $$e_{n+1}=0;\ e_k = e_{k+1} + |z_k| \text{ または } e_k = |x e_{k+1}| + |a_k|;\ k=n,n-1,\cdots,1,0$$

のように，e_0を求めれば，$\Delta(f(x)) = e_0 * M^{-L}$ である。

§9 べき級数

べき級数

(9.1) $$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n + \cdots$$

の計算は，項数を一定に定めれば，多項式と同じであるが，収束を判定しつつ計算するとすれば，任意の級数では困難であるから，条件がよい，各項の絶対値が単調減少する級数と仮定しよう。このような級数の計算は$b_k = a_k/a_{k-1}$と係数を変形した，次のような計算手順が考えられる。

(9.2) $$\begin{cases} f_0 = a_0;\ z_0 = a_0;\ z_k = z_{k-1} * b_k * x;\ f_k = f_{k-1} + z_k;\ k=1,2,\cdots \\ f(x) = f_\infty \end{cases}$$

収束は

(9.3) $e_0=|a_0|$; $e_k=\max(|f_{k-1}|,|Z_k|,|f_k|)$; $k=1,2,\cdots$

とするとき

(9.4) $|Z_k|<e_k*M^{-L}$

を満足すれば，収束とみなせばよい．　正項級数では，新しい項を加えても，和が変らないことで収束とみるのが実用的であるが，交項級数ではそれでは計算量が増えるので，(9.3)式の代りに，各項の絶対値の和を求める次式を使って

(9.5) $e_0=|a_0|$; $e_k=e_{k-1}+|Z_k|$; $k=1,2,\cdots$

(9.6) $(k+1)|Z_k|<e_k*M^{-L}$

収束を判定すればよい。　(9.6)式はほぼ，e_{k-1}に$k|Z_k|$を加えても値が変らないことで判定すればよいことを示している。

§10　高次代数方程式

実係数で実根の場合を考えると，関数値の評価法については§8で論じた通りである。　Newton法を考えれば，根の近傍で，$\Delta(f(x))=e_0*M^{-L}$とし，$f'(x)$が1桁でも正しければ，根の補正限界は$|\Delta(f(x))/f'(x)|$である。　従って，根の正しく求められる桁数は次式のようになる。

(10.1) (根の正しく求められる桁数) $=\log_M|x/\{\Delta(f(x))/f'(x)\}|$
$=L-\alpha$

ここに，

(10.2) $\alpha=\log_M|e_0/(xf(x))|$

この α の値によって，代数方程式を特徴づけることができる。従来の“近接根があると計算が困難である”と言うような定性的な特徴づけ方が，“この方程式のこの根は，損失桁数が何桁である”と言うように，定量的に特徴づけることができる。

一般に，近接根を有する時，反復法による求根法は収束が遅くなるが，$f(x)$ と $f'(x)$ がある程度精度を失うとき，その近似根の点に原点を移動するとよい。原点移動の計算は極端に損失桁数を増大しないから，精度はあまり変らず，収束の速さが速くなる。

§11 数値例

◎ 連立一次方程式の例題 (1)

$$AX = C;\ A=\begin{bmatrix} 1 & 1/2 \\ 1/2 & 1/3 \end{bmatrix};\ C=\begin{bmatrix} 5 \\ 3 \end{bmatrix};\ X=\begin{bmatrix} 2 \\ 6 \end{bmatrix};\ A^{-1}=\begin{bmatrix} 4 & -6 \\ -6 & 12 \end{bmatrix}$$

近似値を $\hat{X} = (2,6)^T$ とすれば，$\Delta(y) = (5,3)^T \times 10^{-L}$ となる．

従って，$\Delta y = \begin{bmatrix} 5, & 5, & -5, & -5 \\ 3, & -3, & 3, & -3 \end{bmatrix} \times 10^{-L}$；$A^{-1}\Delta y = \begin{bmatrix} 2, & 38, & -38, & -2 \\ 6, & -66, & 66, & -6 \end{bmatrix} \times 10^{-L}$

Δy は残差の上限領域を与えるから，$A^{-1}\Delta y$ は X の誤差の上限領域，すなわち，計算限界を与える．これを別図に示す．$\hat{X}$ は X を中心としたこの四辺形内に入り，x_1 と x_2 は互に関連して動く．従って，個々の精度を論ずるのは，あまり意味があるように思えない．例えば，絶対値で，x_1 に 0.035，x_2 に 0.060 の誤差を入れた $X_1 = [2.035, 5.940]^T$，$X_2 = [2.035, 6.060]^T$ の残差を求めると，$R_1 = [+0.0050, -0.0025]^T$，$R_2 = [+0.065, +0.0375]^T$ となる．根の精度としては，X_1 も X_2 も同じであるが，残差は1桁の違いがある．

170

Xの誤差の上限領域.

- 問題

$$\begin{bmatrix} 1 & 1/2 \\ 1/2 & 1/3 \end{bmatrix}\begin{bmatrix} 2 \\ 6 \end{bmatrix}=\begin{bmatrix} 5 \\ 3 \end{bmatrix}$$

- 係数の逆行列

$$A^{-1}=\begin{bmatrix} 4, & -6 \\ -6, & 12 \end{bmatrix}$$

- 固有値

$\lambda_1 = 1.267\cdots$ $\quad \lambda_1^{-1} = 0.7888\cdots$

$\lambda_2 = 0.06574\cdots$ $\quad \lambda_2^{-1} = 15.21\cdots$

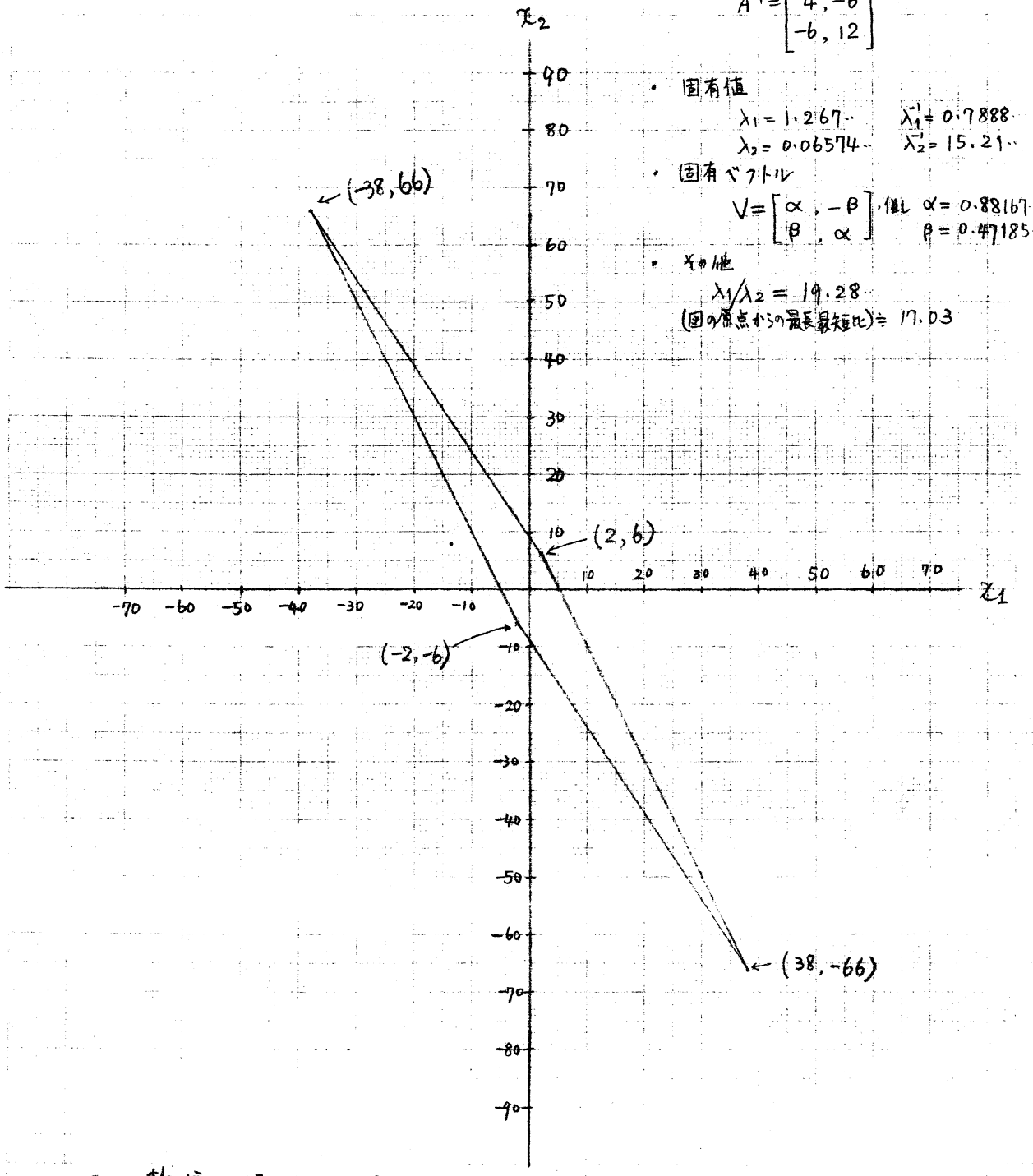
- 固有ベクトル

$$V=\begin{bmatrix} \alpha, & -\beta \\ \beta, & \alpha \end{bmatrix}$$ ただし $\alpha = 0.88167$, $\beta = 0.47185$

- その他

$\lambda_1/\lambda_2 = 19.28\cdots$

(図の原点からの最長最短比) $\fallingdotseq 17.03$



図は $10^{\frac{1}{4}}$ 倍に拡大してある.

⊙ 逆行列の例題

$$A=\begin{bmatrix}100, & 99\\ 99, & 98\end{bmatrix};\quad A^{-1}=\begin{bmatrix}-98, & 99\\ 99, & -100\end{bmatrix}$$

$$AA^{-1}=\begin{bmatrix}-9800+9801, & 9900-9900\\ -9702+9702, & 9801-9800\end{bmatrix},\quad e=\begin{bmatrix}9801, & 9900\\ 9702, & 9801\end{bmatrix}\times 10^{-L}$$

eの要素の符号を適当に取ったものを $\Delta(AA^{-1}-E)$ とすれば，誤差の限界 $\Delta(A^{-1})$ は

$$\Delta(A^{-1})=A^{-1}\Delta(AA^{-1}-E)=\begin{bmatrix}\ominus 960498 \triangle\pm 960498, & \diamond 970200 \boxplus 970299\\ \oplus 970299 \triangle\mp 970200, & \diamond 980100 \boxplus 980100\end{bmatrix}\times 10^{-L}$$

で，○,△,◇,□の中の符号をそれぞれ独立に組合せた16通りの ΔA^{-1} を得る。数値は 10^{-L} を乗ずるのを省略すると，次の8組と，この8組の要素の符号を変えた8組である。

$$\begin{bmatrix}1920996, & 99\\ -1940499, & 0\end{bmatrix},\begin{bmatrix}-1920996, & 99\\ 1940499, & 0\end{bmatrix},\begin{bmatrix}1920996, & 1940499\\ -1940499, & -1960200\end{bmatrix},\begin{bmatrix}-1920996, & 1940499\\ 1940499, & -1960200\end{bmatrix}$$

$$\begin{bmatrix}0, & 99\\ 99, & 0\end{bmatrix},\begin{bmatrix}0, & 99\\ -99, & 0\end{bmatrix},\begin{bmatrix}0, & 1940499\\ 99, & -1960200\end{bmatrix},\begin{bmatrix}0, & 1940499\\ -99, & -1960200\end{bmatrix}$$

A^{-1} に $\Delta(A^{-1})$ の誤差が入った時，すなわち，$\tilde{A}^{-1}=A^{-1}+\Delta(A^{-1})$ のとき $A\tilde{A}^{-1}-E=e$ となる。10進L桁の計算では e は有意な値ではない。$\Delta(A^{-1})$ の要素の絶対値最大は約 $2\times 10^{6-L}$ である。A^{-1} の要素の絶対値は約 10^{2} であるから，$\Delta(A^{-1})/A^{-1}\doteqdot 2\times 10^{4-L}$ で，これから損失桁数は約4.3桁である。

⊙ 連立一次方程式の例題 (2)

$$AX=C;\quad A=\begin{bmatrix}100,99\\99,98\end{bmatrix};\quad C=\begin{bmatrix}1\\-1\end{bmatrix};\quad X=\begin{bmatrix}-197\\199\end{bmatrix};\quad A^{-1}=\begin{bmatrix}-98,99\\99,-100\end{bmatrix}$$

$$AX=\begin{bmatrix}-19700+19701\\-19503+19502\end{bmatrix}\quad から\quad e=\begin{bmatrix}19701\\19503\end{bmatrix}\times 10^{-L}=\begin{bmatrix}e_1\\e_2\end{bmatrix}\times 10^{-L}$$

e の要素の符号を適当に取ったものを $\Delta(AX-C)$ とし、X の誤差を $\Delta(X)$ とすれば、$\Delta(X)=A^{-1}(AX-C)$ であるから、

$$\Delta(X)=A^{-1}\begin{bmatrix}e_1, e_1, -e_1, -e_1\\e_2, -e_2, e_2, -e_2\end{bmatrix}\times 10^{-L}$$

$$=\begin{bmatrix}+99, -3861495, +3861495, -99\\+99, +3900699, -3900699, -99\end{bmatrix}\times 10^{-L}$$

$X^*=X+\Delta(X)$ としたとき、AX^* と C は L 桁計算では区別がつかない。これは、x_1 と x_2 の増減の方向が同じならば、約 10^{2-L} 動かしてもよいが、違う方向に動かすとすれば、約 $4\times 10^{6-L}$ 動かしてよいことを示している。$\Delta(X)$ と X の要素の比から損失桁数は約4.3桁である。具体的に $L=6$ を与えると次のようになる。A^{-1}, X の近似値 $\hat{A}^{-1}$, $\hat{X}$ をそれぞれ次のように取る:

$$\hat{A}^{-1}=\bar{A}^{-1}+\begin{bmatrix}-1.921, 1.941\\1.941, -1.960\end{bmatrix}=\begin{bmatrix}-99.921, 100.94\\100.94, -101.96\end{bmatrix}$$

$$\tilde{X} = X + \begin{bmatrix} -3.937 \\ 3.977 \end{bmatrix} = \begin{bmatrix} -200.937 \\ 202.977 \end{bmatrix}$$

そうすれば，次のような値が得られる．

$$\tilde{R} = A\tilde{X} - C = \begin{bmatrix} -20093.7\square\square + 20094.7\square\square - 1.0 \\ -19892.\overset{8}{\square}\square\square + 19891.7\square\square + 1.0 \end{bmatrix} = \begin{bmatrix} 0.0 \\ -0.1 \end{bmatrix}$$

この $\tilde{R}$ の値は，予定よりも5倍大きい．　それは，a の相対零を近似的に $|a| \times M^{-L}$ としたからである。　正確には，

$k_1 \cdot |a| \cdot M^{-L} < \Delta(a) \leq k_2 \cdot |a| \cdot M^{1-L}$ であり，$|a|$ が下限に近いほど，$\Delta(a)$ は $k_2 \cdot |a| \cdot M^{1-L}$ に近ずく。　そして今の場合がそれである．

$k = 1/2$ も考え入れて，$e = \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix}$ が今の場合，より正確であろう。　このようにすると，

$$\Delta(X) = \tilde{A}^{-1} \begin{bmatrix} 0.1, & 0.1, & -0.1, & -0.1 \\ 0.1, & -0.1, & 0.1, & -0.1 \end{bmatrix} = \begin{bmatrix} +0.1019, & -20.0861, & +20.0861, & -0.1019 \\ -0.1020, & +20.2900, & -20.2900, & +0.1020 \end{bmatrix}$$

となる．　従って，$X = \tilde{X} + \Delta(X)$ として，X_1, X_2, X_3, X_4 を次のように求めて，残差を示すと次のようになる．

$$X_1 = \begin{bmatrix} -200.835 \\ +202.875 \end{bmatrix},\ X_2 = \begin{bmatrix} -221.023 \\ +223.267 \end{bmatrix},\ X_3 = \begin{bmatrix} -180.851 \\ +182.687 \end{bmatrix},\ X_4 = \begin{bmatrix} -201.039 \\ +203.079 \end{bmatrix}$$

$$R_1 = \begin{bmatrix} -20083.5\square\square + 20084.6\square\square - 1.0 \\ -19882.\overset{7}{\square}\square\square + 19881.\overset{8}{\square}\square\square + 1.0 \end{bmatrix} = \begin{bmatrix} +0.1 \\ +0.1 \end{bmatrix} \quad ; R_i = AX_i - C \quad i = 1, 2, 3, 4.$$

$$R_2 = \begin{bmatrix} -22102.3\square\square + 22103.4\square\square - 1.0 \\ -21881.\overset{3}{\square}\square\square + 21880.\overset{2}{\square}\square\square + 1.0 \end{bmatrix} = \begin{bmatrix} +0.1 \\ -0.1 \end{bmatrix}$$

$$R_3=\begin{bmatrix}-18085.100+18086.013-1.0\\ -17904.249+17903.348+1.0\end{bmatrix}=\begin{bmatrix}-0.1\\ +0.1\end{bmatrix}$$

$$R_4=\begin{bmatrix}-20103.900+20104.802-1.0\\ -19902.898+19901.798+1.0\end{bmatrix}=\begin{bmatrix}-0.1\\ -0.2\end{bmatrix}$$

$A(X+\delta X)=C+\delta C$ とするとき，$\|\delta X\|/\|X\| \leqq \|A\|\cdot\|A^{-1}\|\cdot\|\delta C\|/\|C\|$ であることが知られている。 $\|A\|\cdot\|A^{-1}\|$ はいわゆる条件数である。 C と δC を適当に選ぶと等号も成立する。 この問題の条件数は Faddeeva に従って，ノルムを定義すると次のようになる。

$\|A\|_{\mathrm{I}}=\max_i \sum_{k=1}^{n}|a_{ik}|$ ； $\|A\|_{\mathrm{I}}\cdot\|A^{-1}\|_{\mathrm{I}}=199\times199=39601$

$\|A\|_{\mathrm{II}}=\max_k \sum_{i=1}^{n}|a_{ik}|$ ； $\|A\|_{\mathrm{II}}\cdot\|A^{-1}\|_{\mathrm{II}}=199\times199=39601$

$\|A\|_{\mathrm{III}}=\sqrt{\lambda_1}$ ；但し λ_1 は A^TA の最大固有値； $\|A\|_{\mathrm{II}}\cdot\|A^{-1}\|_{\mathrm{II}}=39206.\cdots$

よく“条件数が大きいと C の微小変化が X の大きな変化となる”と言われるが，それはあくまで，“条件数が大きいと C のノルムの微小変化が X のノルムの大きな変化となり得る”と言うべきである。 個々の要素は条件がよくとも悪い結果が得られることがある。 次の例題がそれを示している。

◎ 連立一次方程式の例題 (3)

$$AX=C\ ;\ A=\begin{bmatrix}100,-1\\99,+1\end{bmatrix};\ C=\begin{bmatrix}-99\\-98\end{bmatrix};\ X=\begin{bmatrix}-197/199\\+1/199\end{bmatrix};\ \bar{A}^{-1}=\begin{bmatrix}1,&1\\-99,&100\end{bmatrix}$$

$$AX=\frac{1}{199}\begin{bmatrix}-19700-1\\-19503+1\end{bmatrix};\ e=\frac{1}{199}\begin{bmatrix}19700\\19503\end{bmatrix}\times10^{-L};\ \bar{A}^{1}C=\begin{bmatrix}-99&-98\\9801&-9800\end{bmatrix}\times199^{-1}$$

$$\Delta(X)=A^{-1}\Delta(AX-C)=\begin{bmatrix}39203,&197,&-197,&-39203\\0,&-3900600,&3900600,&0\end{bmatrix}\times39601^{-1}\times10^{-L}$$

$$\fallingdotseq\begin{bmatrix}1,&1/200,&-1/200,&-1\\0,&-100,&100,&0\end{bmatrix}\times10^{-L}$$

$$\|A\|_E\cdot\|A^{-1}\|_I=101\times1=101\ ;\ \|A\|_E\cdot\|A^{-1}\|_E=199\times\frac{101}{199}=101\ ;\ \|A\|_m\cdot\|A^{-1}\|_m\fallingdotseq99.5$$

以上の数値によって，条件数は約100だから，前例題に比べて，条件は悪くないと言うべきであろう。確かに，$\Delta(X)$を見ても悪くないように見える。しかし，$\max|\Delta x_1/x_1|$, $\max|\Delta x_2/x_2|$から，各々の損失桁数は約0桁，約4.3桁である。ノルムからの，より解が得られるはずと言うのはx_1に現われている。$[x_1,1]^T/x_2$は前例題の解X^Tを表わすが，$[x_1,1]^T$が正しくても，x_2が悪いために，この例題のようにして解いても，結局，前例題と同じ結果しか得られない。一般に，条件数が小さいとよい解が出るが，それはあくまでノルムとしてよい解となるのであって，各々の解の相対精度がすべてよいわけではなく，何個かがよくて，他は悪いと言うことがあり得る。