数 値 解 析 39－1
（1991. 12. 6）

最適な近似式の形の選択について

浜田 穂積

電気通信大学情報工学科

多項式，連分数を含む広義ｎ次の有理式はｎ＋１通りの形を持っており，それだけ関数ルーチンの設計の自由度が大と言えるが，これらを含みさらに広いクラスを形成する混合形式の近似式は，３次は５通り，４次は８通り（以下フィボナッチ数通り）とさらに多くの形の中から選ぶことができる．しかし，指数的に増加するこの場合は，全ての形について最良近似式を計算してその中から選ぶことは困難になる．そこで簡単な方法で，それぞれの形の最良近似式（を計算したと仮定したとき）の最大誤差を推測する方法を示し，それを基にした関数ルーチン設計の指針を与える．

On the Method for Selecting the Most Suitable Form of the Approximation as the Mathematical Software

HAMADA, Hozumi

Department of Computer Science
University of Electro-Communications
Chofu, Tokyo 182, Japan

Since the rational expression of n-th order in a broad sense has n+1 kinds of the form, the flexibility on designing the routine of the function programs is fairly wide. The mixed-form forms wider class of approximations than the rational expressions. It is difficult, however, to compute minimax approximation of the all kinds of the mixed-form expressions that have the number of exponential order. This paper shows the method for selecting the suitable form among the expressions for designing the function routine as the mathematical software.

1. はじめに

関数の近似値の計算法として最もしばしば用いられるのは多項式である．n 次の多項式が

$$g(x)=b_0+b_1x+\cdots+b_nx^n$$

であるとき，その計算法は

$$z_n=b_n$$
$$z_i=x\times z_{i+1}+b_i \qquad (i=n-1,\ \cdots,\ 1,\ 0)$$
$$g(x)=z_0$$

による．これは Horner 法と呼ばれているが，本質的に掛けて足す $x\times z+b$ を n 回行なう．

一方，古来近似値の計算にしばしば用いられる連分数の，関数項の形式も重要である．これを次の形とする．

$$g(x)=b_0+\frac{x}{b_1+}\cdots+\frac{x}{b_n}$$

このとき，その計算法は

$$z_n=b_n$$
$$z_i=x\div z_{i+1}+b_i \qquad (i=n-1,\ \cdots,\ 1,\ 0)$$
$$g(x)=z_0$$

によるが，これは多項式の場合の Horner 法に類似していることに気付く．そこでこれも区別せず Horner 法と呼ぶことにする．また同様にこの式を (広義)n 次連分数と呼ぶことにする．

しかしながら，(n 次の) 連分数は n 回の除算を必要とし，現今の大型計算機は乗算と比べて長い計算時間を必要とする不利があるため，これを整理して，多項式と多項式の比である有理式を用いるのが普通になっている．さらに有理式は分子と分母の次数を調整できるので，多様な形を近似式として用いることができるメリットがある．上述の多項式等と計算量の点で比較するため加減算の回数に着目すると，有理式の次数は分子と分母の次数の和で規定するのが適している．以上をまとめると，n 次の近似式の演算回数は次の通り．

（i）多項式：加減算 n，乗算 n
（ii）連分数：加減算 n，除算 n
（iii）有理式：加減算 n，乗算 n−1，除算 1

乗除算の和は n 回と，計算法によらない．

これはさておき，多項式と連分数の類似性に着目し，i 回目の乗算あるいは除算をそれぞれ自由に選択する式を混合形式として提案している [2, 3]．n 次の混合形式は 2^n 通り存在すると考えられるが，異なる形の式の中に等価なものの組が有るので，本質的には，0 次 (定数) は 1 個，1 次は 2 個，2 次は 3 個，以下フィボナッチ数個である．n 次有理式は n+1 通りなので，すべての有理式を含む混合形式には，3 次以上で有理式にも属さない形の近似式が存在することも示されている．フィボナッチ数は指数関数的 ($\sim 1.17\times 1.618^n$) に増加するので，高次の混合形式からは選択の範囲が広く，設計の自由度が高いと言える．多様な形式の近似式の中から設計条件を満たす最適な形式を選ぶには，網羅的に全ての形式について最良近似式を計算すればよいが，n+1 通りの有理式の場合は容易であるが，混合形式では膨大な計算量を必要とする．そこで目星をつけた形式の最良近似式の最大誤差を，その計算を行なうことなく推定できれば，関数ルーチン設計の強力な資料となるに違いない．

2. 混合形近似式

ここで混合展開の定式化を，前章と違う方法で行なう．展開の第 i 段を次の何れかとする．

$$y_i=a_i\times(y_{i+1}x+1) \qquad (1p)$$
$$y_i=a_i\div(y_{i+1}x+1) \qquad (1c)$$

また関数 f(x) は次の通りである．

$$f(x)=y_0 \qquad (2)$$

近似式はこの $i=0$ からあるところまでで打切ったものと考える．

変更する理由は二つある．

（i）連分数の元の定義式では，演算の最初の項が $x\div b_n$ であるが，これは $x\times b_n$ で良いはずで，不必要な自由度である．また最終的に元の式の逆数の形も含めたい．
（ii）a_i の傾向は収束半径を示し，収束の程度を知るための情報となる．

混合形式は，第 i 段を (1p)，(1c) の何れか自由に選択可能とするものである．各段が何れを取るかを明示するため，左側から作用する演算子

$$P_a^x y=a\times(yx+1)$$

$$C_a^x y = a \div (y\,x+1)$$

を定義する．こうすると多項式は

$$P_{a_0}^x P_{a_1}^x \cdots P_{a_{n-1}}^x a_n$$

連分数は

$$C_{a_0}^x C_{a_1}^x \cdots C_{a_{n-1}}^x a_n$$

と表わすことができる．そしてこのＰ演算子とＣ演算子を任意に混合したものが混合形式である．上の例に見る通り，上下の添字ｘとa_iは冗長であるから，単にＰとＣの列（演算子列）のみによって式の性質を規定することができることは明かである．ｎ次の混合形式は名目上2^n通りあると思われるが，

$$P_u C_v w = u(1+vx/(1+wx))$$
$$= C_u C_{-v}(v+w)$$

という関係が成り立つので，部分列ＣＣはＰＣに置き換えることができる．演算子列中のＣの個数は除算の回数であるから，ＣＣのままよりＰＣに置き換えたものの方を考察の対象にする．このことから本質的に異なる個数はＰとＣからなる長さｎの列のうちＣが連続しないものの個数はフィボナッチ数であるということになる．

この形の最良近似式を計算するには，被近似関数の最初のｎ項は近似式と同じ形に展開して，その剰余項は収束し易い形に展開したものを比較に用いることによって容易に計算可能である．

3. 混合形最良近似式の例

５次の混合形近似式の形は全部で13通りある．$|x| \leqq \pi/4$における相対誤差に関する最良近似式の最大相対誤差は次の通りである．

PPPPP	4.505e-15
PPPPC	1.888e-15
PPPCP	1.200e-16
PPCPP	1.859e-15
PPCPC	2.399e-15
PCPPP	1.427e-15
PCPPC	1.035e-14
PCPCP	5.877e-15
CPPPP	6.908e-11
CPPPC	1.306e-13
CPPCP	1.264e-13
CPCPP	9.783e-14
CPCPC	8.572e-15

これを見ると，PPPCPが特異に誤差が小であり，これが設計条件を満たせば採用したいと思うに違いない．このことが，最良近似式を計算することなしに分からないものであろうか，というのがこの論文の目標である．例えば事前にPPPCPがよいと分かれば，網羅的に最良近似式を計算しないで，必要なものだけ計算することで労力の節約になる．

4. 最良近似式の概念的計算法

最良近似式は次のようにして，概念的に計算できる．近似式の適用区間は必ずしも$-1 \leqq x \leqq 1$ではないので，一般性を損なうことなく$|x| \leqq \rho$と仮定できる．被近似関数$f(x)$を，ｘを一次変換して$|x| \leqq 1$に写像すると$F(x)$になるものとする．$F(x)$をベキ級数に展開すると

$$F(x) = c_0 + c_1 x + \cdots + c_n x^n + \cdots$$

となるものとする．ｘのｎ次のべきはｎ次以下のチェビシェフ多項式の一次結合で表わされるからこれを全てのｘのベキに代入すると

$$= p_0 T_0(x) + p_1 T_1(x) + \cdots$$
$$+ p_{n+1} T_{n+1}(x) + p_{n+2} T_{n+2}(x) + \cdots$$

となるものとする．このとき$T_n(x)$のところで打ち切った式

$$G(x) = p_0 T_0(x) + p_1 T_1(x) + \cdots + p_n T_n(x)$$

に，先とは逆に$T_i(x)$をｘで表した式を代入し，$|x| \leqq 1$を$|x| \leqq \rho$に逆写像したものを$g(x)$とすればこれは最良近似式に近いと言え，この式の誤差の主要項は$p_{n+1} T_{n+1}(x)$である．もし$i > n+1$で$|p_{n+1}| \gg |p_i|$が成り立てば，誤差を$|p_{n+1}|$と考えても差支えない．ところで上述の事実から，p_{n+1}はc_{n+1}とそれ以上の添字のもののみに依存する．ここでやはり$i > n+1$で$|c_{n+1}| \gg |c_i|$が成り立てば，

$$p_{n+1} \sim 2^n c_{n+1}$$

となるが，もとのx^{n+1}の係数b_{n+1}で表わせば，

$$p_{n+1} \sim 2(\rho/2)^{n+1} b_{n+1}$$

である．これは$\rho \to 0$で漸近的に成り立つ．

5. 最良近似式の最大誤差の推定

前章のp_{n+1}が最良近似式の最大誤差の推定値

を与えるが，これは次のようにして求められる．f(x) を被近似関数，g(x) を着目している n 次の混合形近似式とする．このとき

$$f(x)=g(x)+b_{n+1}x^{n+1}+O(x^{n+2})$$

を満たす x^{n+1} の係数 b_{n+1} を求める．これから

$$|2(\rho/2)^{n+1}b_{n+1}|$$

が推定値である．相対誤差の場合は

$$|2(\rho/2)^{n+1}b_{n+1}/f(0)|$$

とする．具体的には，g(x) が多項式でない限り x^{n+1} 以上の項も含むことを念頭において，f(x)，g(x) の両方をベキ級数に展開する．計算は x^{n+1} 次の項まで計算すれば十分である．そして f(x)−g(x) の最高次の係数を b_{n+1} とする (n 次以下の係数は 0 であるはずである)．

これまでこのような議論においては

$$f(x)=g(x)+O(x^{n+1})$$

のように，剰余項の位数のみを問題にし，係数を問題にしないのが普通であるが，ここでは剰余項の最低次の係数のみは問題にする．

なお前章の仮定から，$i>n+1$ で $|p_{n+1}|\gg|p_i|$ が成り立つことが前提であるので，そうでないときは推定値の評価を誤ることになる．

ここで，推定値と真の誤差，およびその比の例を次に示す．これは先の例と同じ条件である．

演算子列	b_6	真の誤差	比
PPPPP	1.606e-10	4.505e-15	1.043
PPPPC	-6.716e-11	1.888e-15	1.045
PPPCP	-3.717e-12	1.200e-16	1.200
PPCPP	6.490e-11	1.859e-15	1.065
PPCPC	8.510e-11	2.399e-15	1.048
PCPPP	6.993e-11	1.427e-15	0.759
PCPPC	-3.634e-10	1.035e-14	1.059
PCPCP	-2.089e-10	5.877e-15	1.046
CPPPP	-2.1633e-6	6.908e-11	1.187
CPPPC	4.5360e-9	1.306e-13	1.070
CPPCP	4.3973e-9	1.264e-13	1.069
CPCPP	3.4221e-9	9.783e-14	1.063
CPCPC	-3.023e-10	8.572e-15	1.054

PPPCP の場合は b_6 が想像以上に具合よくキャンセルしすぎたために比が若干大になっている．また PCPPP の場合は b_6 のみキャンセルが十分でなかったと推測される．

6. 最適な近似式の選択法

(1) 数式処理を用いて，必要な次数までのすべての本質的な混合形式の剰余の主要項の係数を網羅的に計算する．これは最良近似式を計算するより簡単である．またこれは最良近似式の評価規準である絶対誤差か相対誤差かに依存しないし，近似式の適用範囲にも依存しない．

(2) 絶対誤差か相対誤差かの評価規準と近似式の適用範囲は設計条件で決まっているのが普通であるので，前項で計算された各形式について最大誤差の推定値を計算する．それを参考にして形式の目星をつける．

(3) 演算子列から四則演算それぞれの演算量を計算し，設計条件を満たすものを選ぶ．以下に演算子列のみから判断できる事実を列挙する．

(i) C の個数は除算の回数である．同様に P の個数は乗算の回数である．加減算の回数は次数に等しい．奇関数以外で，演算子列の左端が C のとき乗算を 1 回追加する．

(ii) 偶関数には乗算を 1 回追加する．あらかじめ x の 2 乗を計算するためのものである．奇関数にはさらに乗算を 1 回追加する．最後に x を乗じるためのものである．

(iii) 以下の条件を満たす場合は有理式に変形することができて，除算の回数を減らしその分乗算に変更できる．条件を満たさないものを無理に有理式に変形すると，演算回数を無用に増加させる結果となる．条件とは，もし演算子列の左端が C であるときはこれを P に置き換える．このときなお C が存在するときは，右端は C であること．かつ左右を C で囲まれた P の列の長さはすべて 1 であること，である．

(4) 前項の手続きで選んだ形式の最良近似式を計算して，設計条件を正しく満たしているかをチェックする．もし満たしていない場合は，それに近いものを調べて探す．

最良近似式の計算は以下の考え方にしたがつて行なう．被近似関数 f(x) はつじつまの合う fr(x) を仮定して

$$y_{n+1}=fr(x) \qquad (3)$$

$$y_i = a_i \times (y_{i+1} x + 1) \quad (4p)$$
$$y_i = a_i \div (y_{i+1} x + 1) \quad (4c)$$
$$f(x) = y_0, \quad (5)$$

近似式は同じく

$$z_{n+1} = 0 \quad (6)$$
$$z_i = a_i \times (z_{i+1} x + 1 + d_i) \quad (7p)$$
$$z_i = a_i \div (z_{i+1} x + 1 + d_i) \quad (7c)$$
$$g(x) = z_0 \quad (8)$$

と定義されているとする ($i = n, \cdots, 1, 0$). ただし, $i = n$ のときだけは P 要素をとる方が望ましい. 最良近似式の計算は, これら $n+1$ 個の値 d_i ($i = 0, 1, \cdots, n$) を求めることである.

7. 近似式の設計例

次に, $|x| \leqq \pi/4$ における IEEE 倍精度の正接のプログラムの設計を例にして示す. この場合は, 相対誤差が $2^{-54} (= 5.55 \times 10^{-17})$ であればよいとすれば, 6 次の式の中にあることが分かっているとして検討を行なうことにしよう. 上の条件を満たす 21 通りのうち 6 次の最良近似式は次の 4 通りである.

PCPCPC	1.886e-17
CPPCPC	3.242e-17
CPCPPC	3.469e-17
CPCPCP	4.491e-17

これらの内下 2 つは有理式に変形すべきでない式であって, 上 2 つが候補である. $\tan x$ は $\pi/2$ を周期とする周期関数で, 偶数番周期では $\tan x$ を, 奇数番周期では $\cot x$ を計算するのであるから, ある偶関数 $f(x)$ について, 偶数番周期では $x/f(x)$, 奇数番周期では $-f(x)/x$ とすれば対称性が保たれて美しい. それには第 1 あるいは第 2 の形が良いことになる. 第 2 の場合は, u を x^2 として $f(x)$ を次の式で近似することを意味している.

$$f(u) = b_0 + b_1 u + b_2 u^2 + u^3 / (b_3 + b_4 u + u^2 / (b_5 + b_6 u))$$

この計算は次のステップで行なわれる.

$$u = x \times x$$
$$y_5 = u \times b_6 + b_5$$
$$y_4 = u \div y_5 + b_4$$
$$y_3 = u \times y_4 + b_3$$
$$y_2 = u \div y_3 + b_2$$
$$y_1 = u \times y_2 + b_1$$
$$y_0 = u \times y_1 + b_0$$

これを見ると, $f(x)$ の計算に必要な演算量は, 加減算 6 回, 乗算 5 回, 除算 2 回である. しかし除算の回数を最小限にしたい要求があれば, 上式の下線部を有理式にすれば乗算 6 回, 除算 1 回となる. $\tan x$ の計算全体としては加減算 6 回, 乗算 6 回, 除算 2 回である. $f(x)$ は全体として有理式にすることができるけれども, 下線部のみと指定したのは, 多項式であれば適用できる丸め誤差を最小に抑える最終ステップの微調整がここでも適用できるからである. すなわち $b_0 \fallingdotseq 1$ の場合は, $b_0 = 1 + \varepsilon$ として, 最終ステップを

$$y_0 = (u \times y_1 + \varepsilon) + 1$$

とする方法である. 奇関数の場合にはこの後で x を掛ける分をまとめて

$$y_0 = (u \times y_1 + \varepsilon) \times x + x$$

とする. 一方全体を有理式にすると, この方法は使えず, かつ分子, 分母両方とも誤差を含んだ値であれば結果の相対誤差がこれらの和となる. 特に困るのは 16 進計算機の場合で, たいへん苦労する. この微調整は加減算 1 回の損失であるが, 十分これを補う効果がある.

8. おわりに

これまで, 近似式の形式の一般形は有理式であること, また計算式としても有理式を用いることが一般と考えられてきた. しかしながら, 最良近似式を精度よく計算することはかなり困難であること [1], 有理式形の近似式での近似値の計算で (特に 16 進計算機で) 高精度を確保することが困難であること [2,3] が分かってきた. これらの問題点をすべて解決するのが混合形式である. 混合形式は同じ次数でも多くの異なる形式を持つので選択の範囲が広くなる. これら多くの形式の中から最適な形を選ぶ方法を具体的に示した.

また, 混合形式が場合によっては除算を多く必要とするという問題点を解決するため, 部分的に

有理式に変形することもよい場合があり，変形によって計算量を増加させるか否かを決定する方法も示した．結果的に有理式を用いる場合も，最良近似式は混合形式で計算し，完成した近似式を有理式に変形する方が，誤差の点で容易である．概括的に言えば，有理近似式は何等取柄のない計算法であると言える．

参考文献

[1] 浜田穂積：有理式近似および連分数近似の最良化について．情報処理，Vol. 19, No. 11 (Nov. 1978), pp.1065-1071

[2] 浜田穂積：混合形近似式の最良化と関数値の計算．数理解析研究所講究録 585(Feb. 1986), pp.26-41

[3] 浜田穂積：混合形近似式による関数値の計算．コンピュータ・ソフトウェア，Vol. 3, No. 2(Apr. 1986), pp.30-36