

区分的最良近似およびそのプログラム†

山下 眞一郎††*

最良近似式は近似条件を満足する超越連立方程式を解くことにより求める。本論文では補間点を操作して最良近似式を求める方法を述べる。その操作は次のように行う。i) 「誤差関数の0点(補間点)と隣の0点との距離」と「その間にある偏差値」の比例関係を仮定する。ii) 偏差値の「ある種の平均値」を求める。iii) 平均値と偏差値との差を旧0点と新0点の距離に変換する。iv) 新0点から最良近似式を求める。さらに、また、近似区間を分割し、各区間で指定された有理式の最良近似式を求め、全体でも偏差値が等しくなるように分割点を決める方法を述べる。

1. ま え が き

一変数関数の最良近似問題は良く研究され、興味の持てる問題はすべて解決済であると言っても過言ではない。そのような状況にあって、敢えてこの問題を取り上げたのは二つの理由による。一つは、やさしい最良近似式の計算方法を提示することである。もう一つは、計算法を二重に使うことにより、区分的にそれぞれ最良近似式を求め、しかも全体でも最良近似する方法を述べることである。

例えば図1のように近似区間の分割数が $NA=3$ の場合、近似区間 $[x_L, x_R]$ 内に、分割点 γ_i を

$$x_L = \gamma_0 < \gamma_1 < \gamma_2 < \gamma_3 = x_R$$

と取って近似区間を3区間に分割し、この各区間で最良近似し、それぞれの区間の最大偏差値 ρ_1, ρ_2, ρ_3 が等しくなるように、分割点 γ_1, γ_2 および各最良近似式を決める。これが問題の設定であり、これを区分的最良近似問題と呼ぶ。このような最良近似問題は、分割数を1とすれば普通の問題であるが、近似区間の両端で性質の違う関数や $[0, \infty]$ のような場合や特定区間の近傍を高速で計算するための近似式を作成するときなどに現れる。

ここでは一変数関数の被近似関数は高階の微分まで連続な性質の良い関数を仮定する。近似式はあらかじめ次数が指定された多項式または有理式とする。

最良近似式は、最良近似の条件、すなわち、近似式の次数を n 、誤差関数を $E(x)$ とする時、

- 1) $E(x)$ の相隣る偏差値の符号が異なる。

- 2) $|E(x)|$ の偏差点は少なくとも $n+2$ 個ある。

- 3) $|E(x)|$ の偏差値の絶対値はすべて等しい。

による。最良近似式を求める普通の方法は偏差点で条件式

$$E(x_k) = (-1)^k \rho$$

$$E'(x_k) = 0$$

$$k=0, 1, 2, \dots, n+1$$

ただし、 ρ は偏差値

端の微分値は非零のことが多い

を求めて、この超越連立方程式を解く。この超越連立方程式の未知数は $2n+4$ 個ある。最良近似式の係数が $n+1$ 個、偏差点が $n+2$ 個、偏差値が1個の合計 $2n+4$ 個である。方程式の数も同じ個数ある。偏差点で微分値が非零の時は、その個数だけ偏差点が既知となり、式も少なくなる。しかも、一般に、被近似関数が連続でも、区間内に微分値が非零の点があると計算は困難である。

さて、ここで述べる最良近似式を求める方法は誤差関数の零点、いわゆる補間点に注目し、これを逐次修正して最良近似式を求める。また、この考え方を二重に適用することにより、区分的最良近似式を求める。

2. 記号の定義

幾つかの記号と問題を設定して置く。

被近似関数: $f(x)$

$$\text{近似式: } PQ(x) = P(x)/Q(x)$$

$$\text{ただし, } P(x) = \sum_{i=0}^L p_i x^i$$

$$Q(x) = \sum_{i=0}^M q_i x^i$$

$$\text{誤差関数 } E(x) = PQ(x) - f(x)$$

$$[\text{相対誤差の場合 } E(x) = PQ(x)/f(x) - 1.0]$$

$$\text{近似式の次数 } n = L + M$$

$$\text{近似区間の左端 } A$$

† Piecewise Best Approximation and Its Program by SHIN-ICHIRO YAMASHITA (Program Products Division, Numazu Plant, Fujitsu Ltd.).

†† 富士通(株)沼津工場PP事業部

* 現在 日本大学理工学部

Presently College of Science and Technology, Nihon University

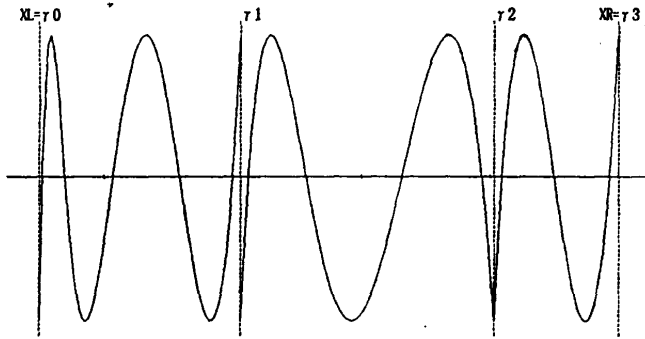


図 1 分割数が3の最良近似誤差

Fig. 1 Error of global B.A. (Best Approximation) by 3 local B.A.

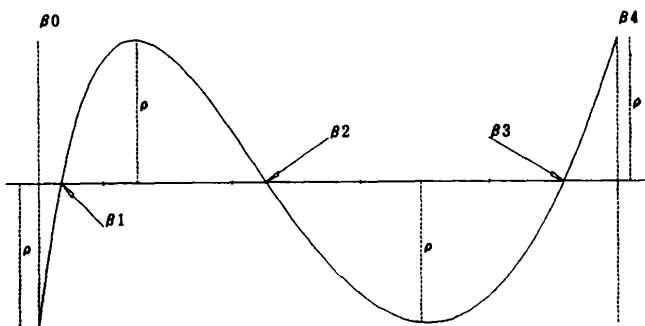


図 2 単区間の最良近似誤差

Fig. 2 Error of best approximation without domain division.

近似区間の右端 B 第 i 補間点 β_i 第 i 区間の幅 S_i 第 i 区間の偏差値 ρ_i

$P(x)$ または $Q(x)$ の係数の 1 個は、固定しなければならない。通常は、 $q_0=1$ とすると便利である。

近似式の次数を n とするとき、近似区間内に両端を含めて補間点 β_i を $n+3$ 点取って

$$A = \beta_0 < \beta_1 < \dots < \beta_{n+1} < \beta_{n+2} = B$$

とする。両端を除きこの β_i の各点で誤差関数が零となり、この点で補間式を求め、最良近似式となるように β_i の点を決める。第 i 区間の幅 S_i は、

$$S_i = \beta_i - \beta_{i-1} \quad (1)$$

$$i=1, 2, \dots, n+2$$

である。

例えば、図 2 のように、一つの区間で、近似式の次数が $n=2$ の場合を考

える。この誤差関数の零点(補間点)は、 $n+1=3$ 個あり、区間の数は $n+2=4$ 個、偏差値 ρ_i は各区間に 1 個対応していると仮定して、 $n+2=4$ 個ある。ここでは、素直な最良近似問題を扱うものとする。

3. 零点または分割点を決める戦略

零点または分割点を決める戦略は、山が高ければ区間を狭め、山が低ければ区間を広めるといものである。その発想は誠に単純である。問題は山を低くするために区間を狭めたり、山を高くするために区間を広める割合である。

例えば図 3 のように分割数 $NA=3$ の場合、それぞれの最大誤差が ρ_1, ρ_2, ρ_3 で、平均(どのような平均を取るかは別にして)を H とすれば、第 1 区間は少し狭め、第 2 区間は広め、第 3 区間は狭める。これを繰り返すことにより、それぞれの最大誤差を等しくすることが期待できる。しかし、区間を狭めたり、広めたりすると、補間の場所が変わり、誤差の様子も変わるかもしれない。このため、山が高ければ区間を狭め、山が低ければ区間を広めるといっても、その方向は良いが、割合は難しい場合もあり得る。その解決のために減速係数 $1 > \alpha > 0$ を使う。

4. 偏差値の平均を決める方法

最良近似式を求めるための誤差関数の零点の修正法は、各区間の偏差値 ρ_i がある平均値よりも大き

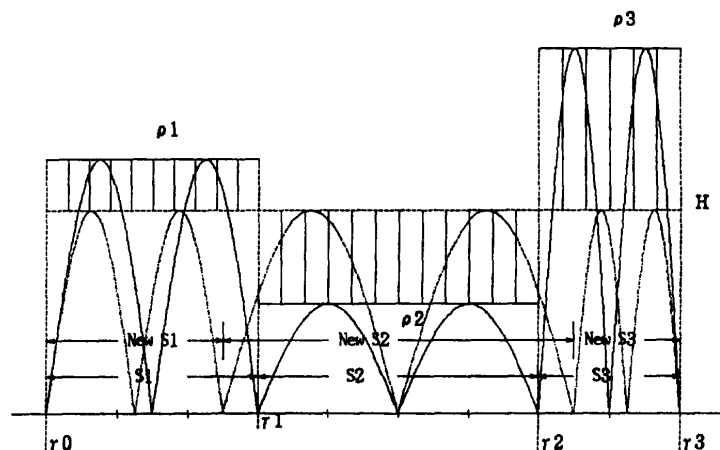


図 3 各区間の最大絶対値誤差を等しくする

Fig. 3 Same maximum absolute error obtained by rearrangement of intervals.

れば、その区間の幅 S_i を狭め、小さければ広げる方法による。誤差関数 $E(x)$ が近似区間内の任意の点 c のまわりに Taylor 展開できるとし、 $m-1$ 次まで取れば、Lagrange の剰余項は、

$$R_m = (x-c)^m \frac{E^{(m)}(\eta)}{m!} \quad (2)$$

と表せる。 η は区間内の値である。これから類推して、第 i 区間の偏差値 ρ_i は、これを考慮して

$$\rho_i = k_i S_i^{m_i} \quad (3)$$

と仮定する。 k_i は比例定数であり、 m_i は、一般には、 $m_i = m+1$ なる整数である。この仮定の下に、区間幅 S_i を ΔS_i だけ変動したとき、その区間の偏差値 ρ_i が $\Delta \rho_i$ だけ変動したとすれば、 S_i に比べて ΔS_i が小さいとして、すなわち $|\Delta S_i/S_i| \ll 1$ を仮定して、次のような結果を得る。

$$\begin{aligned} \rho_i + \Delta \rho_i &= k_i (S_i + \Delta S_i)^{m_i} \\ &= k_i S_i^{m_i} \left(1 + \frac{\Delta S_i}{S_i} \right)^{m_i} \\ &= \rho_i \left(1 + \frac{\Delta S_i}{S_i} \right)^{m_i} \\ &\doteq \rho_i \left(1 + \frac{m_i \Delta S_i}{S_i} \right) \end{aligned}$$

両辺から ρ_i を引けば、

$$\begin{aligned} \therefore \Delta \rho_i &= \rho_i m_i \Delta S_i / S_i \\ \therefore \Delta S_i &= \frac{S_i \Delta \rho_i}{m_i \rho_i} \quad (4) \end{aligned}$$

となる。一方、区間幅の変動量の代数和は不変であり、その総量は 0 であるから、

$$0 = \sum_{i=1}^{n+2} \Delta S_i = \sum_{i=1}^{n+2} \frac{S_i \Delta \rho_i}{m_i \rho_i} \quad (5)$$

である。最良近似の目標は、すべての偏差値 ρ_i が等しくなることであるから、そのような値を H とすれば、

$$\begin{aligned} H &= \rho_i + \Delta \rho_i \\ \text{または } \Delta \rho_i &= H - \rho_i \quad (6) \end{aligned}$$

でなければならない。この(6)式を(5)式の右辺に代入して、

$$\begin{aligned} 0 &= \sum_{i=1}^{n+2} \frac{S_i (H - \rho_i)}{m_i \rho_i} = \sum_{i=1}^{n+2} \frac{S_i H}{m_i \rho_i} - \sum_{i=1}^{n+2} \frac{S_i}{m_i} \\ \therefore H &= \frac{\sum_{i=1}^{n+2} \frac{S_i}{m_i}}{\sum_{i=1}^{n+2} \frac{S_i}{m_i \rho_i}} \quad (7) \end{aligned}$$

となる。これが求める偏差値の平均である。

5. 簡略法

(3)式と(7)式を簡略にする方法を述べる。それは、前節で $m_i=1$ とする方法である。(3)式は、

$$\rho_i = k_i S_i \quad (8)$$

となり、(7)式は

$$\therefore H = \frac{(B-A)}{\sum_{i=1}^{n+2} S_i / \rho_i} \quad (9)$$

となる。(6)式は変わらず、(4)式は変わる。

$$\Delta \rho_i = H - \rho_i \quad (10)$$

$$\Delta S_i = S_i \Delta \rho_i / \rho_i \quad (11)$$

新しい補間点 $New \beta_i$ は

$$New \beta_i = New \beta_{i-1} + S_i + \Delta S_i \quad (12)$$

$$i = 1, 2, \dots, n+2$$

$$New \beta_0 \equiv \beta_0, \quad New \beta_{n+2} \equiv \beta_{n+2}$$

である。減速係数は ΔS_i に掛け、減速する。

この簡略法で十分に速く収束する。今後の説明は簡単のために簡略法を用いる。

6. 単区間の計算手順

近似区間の両端および誤差関数の零点(補間点)を与えれば、近似式が求められる。(1)式から各区間幅 S_i が定まり、その各区間に対応する偏差値を求めれば、(7)式から偏差値の平均値 H が求められる。(6)式から各偏差値の変動量 $\Delta \rho_i$ が求まり、(4)式から各区間幅の変動量 ΔS_i が求められる。新しい区間幅は、 $S_i + \Delta S_i$ であるから、これから新しい零点の位置が決められる。手順を示せば、以下のとおりである。

1) 近似式の分子、分母の次数 L, M を与える。

2) $n = L + M$ を求める。

3) 誤差関数の初期零点 β_i を

$$\beta_i = \frac{A-B}{2} \cos \left\{ \left(\frac{2i-1}{n+1} \right) \frac{\pi}{2} \right\} + \frac{A+B}{2}$$

$i = 1, 2, \dots, n+1$ のように与える。

4) $Q(x)$ の 0 次の係数 q_0 を 1 と置いて $P(x), Q(x)$ の係数 p_k, q_k に関する $n+1$ 元連立方程式(補間方程式)

$$\sum_{k=0}^L p_k x_i^k - \sum_{k=1}^M q_k \{f(x_k) x_i^k\} = f(x_i)$$

ただし、 $i = 1, 2, \dots, n+1$

を解き、 p_k, q_k を求める。

5) 各区間内に偏差値 ρ_i を求める。

6) すべての偏差値 ρ_i が許容範囲内で互いに一致すれば、4)で求めた近似式を最良近似式とし、計算

を終了する。一致しなければ7)へ行く。

- 7) 零点 β_i を修正するために, (7)式で平均値 H を求め, (6), (4)式を使って新しい零点 $New\beta_i$ は,

$$New\beta_i = New\beta_{i-1} + S_i + S_i(H/\rho_i - 1)\alpha$$

$$i=1, 2, \dots, n+2$$

$$\text{ただし, } New\beta_0 \equiv \beta_0, \quad New\beta_{n+2} \equiv \beta_{n+2}$$

から求める。 α は減速係数である。新しい零点が決まれば4)から反復する。

(7)式または(9)式を使って平均値を求める。これを利用して新しい区間を求める。これを反復すれば良い。

7. 区分的な計算手順

区分的な最良近似法は, 単区間の最良近似法を二重

表 1 最良近似の計算結果
Table 1 Result of best approximation.

収束するまでの反復回数 = 16			
近似多項式			
P 0	=	6.641933219902010910E-01	
P 1	=	3.871467468886157780E-01	
P 2	=	-1.410594281301297440E-02	
K	偏差点	偏差値	0 点
1	1.0000000000E+00	-3.7234126066E-02	1.0000000000E+00
2	2.5259371955E+00	3.7219012369E-02	1.3406510019E+00
3	6.9428748385E+00	-3.7215205038E-02	4.5249139593E+00
4	1.0000000000E+01	3.7211150593E-02	9.1692794131E+00
5			1.0000000000E+01
最大誤差 =		3.723412606580379E-02	
最小誤差 =		3.721115059331837E-02	

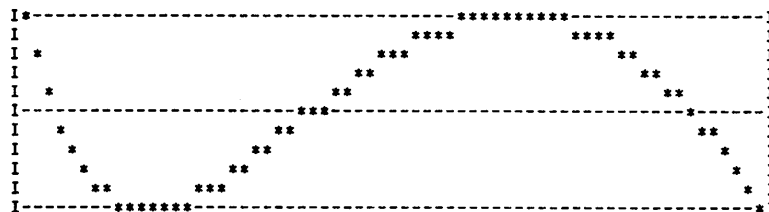


表 2 区分的最良近似の計算結果
Table 2 Result of piecewise best approximation.

*** 第 1 区間 *****			
分子多項式		分母多項式	
P0=	1.531083964024616990E+00	Q0=	1.527854144491143980E+00
P1=	-6.003807333474123760E-01	Q1=	1.000000000000000000E+00
K	偏差点	偏差値	0 点
1	0.0000000000E-01	-2.1139580274E-03	0.0000000000E-01
2	2.0956178170E-01	2.1135746908E-03	5.3265645694E-02
3	7.4842639722E-01	-2.1135582538E-03	4.5960373182E-01
4	1.1250717315E+00	2.1126032946E-03	1.0188541951E+00
5			1.1250717315E+00
最大誤差 =		2.113958027419427E-03	
最小誤差 =		2.112603294643323E-03	
*** 第 2 区間 *****			
分子多項式		分母多項式	
P0=	5.961463184217626890E-01	Q0=	1.882859723940601140E-01
P1=	-1.484493129305139160E-01	Q1=	1.000000000000000000E+00
K	偏差点	偏差値	0 点
1	1.1250717315E+00	-2.1136343192E-03	1.1250717315E+00
2	1.4164617619E+00	2.1133265711E-03	1.1957150573E+00
3	2.2696458588E+00	-2.1134473801E-03	1.7814444016E+00
4	2.9669700127E+00	2.1122348594E-03	2.7529070248E+00
5			2.9669700127E+00
最大誤差 =		2.113634319232931E-03	
最小誤差 =		2.112234859360707E-03	
*** 第 3 区間 *****			
分子多項式		分母多項式	
P0=	6.498192921769717360E-02	Q0=	-2.203096892640085790E+00
P1=	-8.109165612214046590E-03	Q1=	1.000000000000000000E+00
K	偏差点	偏差値	0 点
1	2.9669700127E+00	-2.1130935594E-03	2.9669700127E+00
2	3.4090282638E+00	2.1117318454E-03	3.0592887378E+00
3	5.4027782136E+00	-2.1122041631E-03	4.0556704023E+00
4	1.0000000000E+01	2.1115698799E-03	7.7084659584E+00
5			1.0000000000E+01
最大誤差 =		2.113093559432175E-03	
最小誤差 =		2.111569879884920E-03	

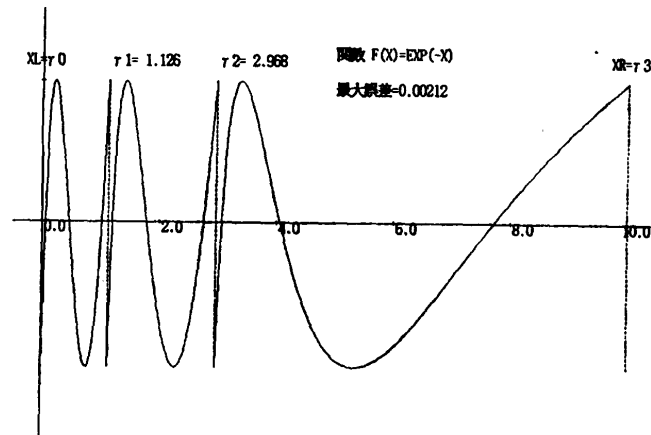


図 4 区分的最良近似の誤差
Fig. 4 Error of piecewise best approximation.

に使用すれば良い。その方法はプログラムの点には単純である。すなわち、単純変数は一次元配列に、一次元配列は二次元配列に記憶する。すべての単区間の最良近似式が求まったとき、各区分の最大偏差値を単なる偏差値と考えれば良い。なお、詳細はプログラムを参照されたい。

8. プログラムおよび計算例

プログラム A (図 5) は単区間の計算手順をプログラム化したものである。その実行条件は次のとおりである。

- 1) 被近似関数は $f(x) = \sqrt{x}$ とする。
- 2) 区間の両端は $[1, 10]$ とする。
- 3) 近似式の分母多項式の次数 = 0
分子多項式の次数 = 2 とする。
- 4) 近似式の型は一般の有理式とする。
- 5) 減速率 $ALPH = 0.3$ とする。
- 6) 印刷頻度は毎回とする。
- 7) 収束判定は相対 10^{-3} とする。

これを翻訳して実行すると少し修正したが表 1 の結果が得られる。その誤差曲線は図 2 である。

プログラム B (図 6) は区間 $NA=3$ の場合をプログラム化したものである。その実行条件は次のとおりである。

- 1) 被近似関数は $f(x) = e^{-x}$ とする。
- 2) 区間の両端は $[0, 10]$ とする。
中間の初期値を $r_1 = 1.0$, $r_2 = 3.0$ とする。
- 3) 全部の近似式の分母多項式の次数 = 1 とする。
分子多項式の次数 = 1 とする。

```

C*****
C 最良近似のプログラム A
!C*****
      INCLUDE 'COM.FOR'
      EXTERNAL F,E
CC 最終結果を格納するファイル
      OPEN(7,FILE='WW')
      AAA=' ' !C タイトル
      A=1.0 !C 区間の左端
      B=10.0 !C 区間の右端
      L=2 !C 分子の次数
      M=0 !C 分母の次数
      N=L+M !C 有理式の次数
      NREP=1 !C 印刷頻度
      NSKEY=0 !C 一般次数
      ALPH=0.3 !C 減速率
      CALL SHOKI !C 初期値
      CALL CONBES !C 最良近似
      RETURN; END
C*****
      FUNCTION F(X) !C 被近似関数
      REAL*8 F,X
      F=SQRT(X)
      RETURN; END
C*****

```

図 5 最良近似のプログラム A
Fig. 5 Program A of best approximation.

- 4) 近似式の型は一般の有理式とする。
- 5) 減速率 $ALPH = 0.3$ とする。
- 6) 減速率 $BETA = 0.3$ とする。
- 7) 印刷頻度は毎回とする。
- 8) 収束判定は相対 10^{-3} とする。

このような条件の下で計算したのが表 2 である。そ

```

C*****
C 最良近似のプログラム B
!C*****
INCLUDE 'COM.FOR' !C 共通変数定義
EXTERNAL F,E
REAL*8 AA(5),EEMAX(5),EMIN(5)
REAL*8 PP(10,5),QQ(10,5),XXZ(10,5),
-   XXS(10,5),XXM(10,5)
REAL*8 EEX(10,5),SS(5)
INTEGER LL(5),NM(5)
OPEN(7,FILE='TABLE')
NA=3 !C 区間の分割数
C r 0, r 1, r 2, r 3 分割点
AA(1)=0.0; AA(2)=1.0
AA(3)=3.0; AA(4)=10.0
C 第 i 区間の分子の次数
LL(1)=1; LL(2)=1; LL(3)=1
NM(1)=1; NM(2)=1; NM(3)=1
NREP=10 !C 印刷の頻度
NSKEY=0 !C 近似式は一般有理式
ALPH=0.3; BETA=0.3 !C 減速率
123 CONTINUE
DO 110 JJ=1,NA !C 区間番号を指定
A=AA(JJ); B=AA(JJ+1) !C 区間右端
L=LL(JJ); N=NM(JJ); M=L+N !C 次数
CALL SHOKI !C 初期値
DO 41 K=1,M+3
41 XYZ(K,JJ)=XZ(K) !C 各零点を格納
110 CONTINUE
CC 以下各区間の最良近似を行う
DO 111 JJ=1,NA !C 区間番号を指定
A=AA(JJ); B=AA(JJ+1) !C 左右区間
L=LL(JJ); N=NM(JJ); M=L+N !C 次数
CC 区間毎の零点を取出
DO 40 K=1,M+3
40 XZ(K)=XYZ(K,JJ)
CALL CONBES !C 区間の最良近似
EEMAX(JJ)=EMAX !C 区間の最大誤差
WRITE(6,*) '区間番号=',JJ,
-   ' 最大誤差=',EMAX
DO 31 K=1,M+3 ; 31 XYZ(K,JJ)=XZ(K)
DO 32 K=1,M+3 ; 32 XM(K,JJ)=XM(K)
DO 33 K=1,M+3 ; 33 EX(K,JJ)=EX(K)
DO 34 K=1,M+3 ; 34 XS(K,JJ)=XS(K)
111 CONTINUE
CC 区間毎の最良近似が求まる
EMAX=0.0; EMIN=1.0E10
DO 54 K=1,NA
C 全体の最大誤差を求める
IF(ABS(EEMAX(K)).GT.EMAX)EMAX=EEMAX(K)
C 各区間の最大誤差の最小を求める
IF(ABS(EEMAX(K)).LT.EMIN)EMIN=EEMAX(K)
54 CONTINUE
C 偏差値が全て等しくなったか?
IF(EMAX-EMIN.LT.EMAX*0.01)GOTO 990
C まだ収束していない
H=AA(NA+1)-AA(1) !C 全距離
IF(NA.EQ.1)GOTO 123 !C 区間が1の場合
CC 区間が2以上の処理
S=0.0
DO 51 K=1,NA
51 SS(K)=AA(K+1)-AA(K) !C 区間の距離
DO 52 K=1,NA
52 S=S+SS(K)/EEMAX(K)
H=H/$ !C 各区間の偏差値の平均
CC 偏差値と平均から区間を変更する
DO 53 K=1,NA-1
AA(K+1)=AA(K)+SS(K)+
-   SS(K)*(H/EEMAX(K)-1.0)*BETA
53 CONTINUE
GOTO 123 !C 反復する
CC 全て収束した
990 CONTINUE
CC 区間毎に復元して印刷
DO 410 JJ=1,NA
A=AA(JJ); B=AA(JJ+1)
L=LL(JJ); N=NM(JJ); M=L+N
EMAX=EEMAX(JJ)
DO 431 K=1,M+3 ; 431 XZ(K)=XYZ(K,JJ)
DO 432 K=1,M+3 ; 432 XM(K)=XM(K,JJ)
DO 433 K=1,M+3 ; 433 EX(K)=EX(K,JJ)
DO 434 K=1,M+3 ; 434 XS(K)=XS(K,JJ)
CALL PRINT(6)
410 CONTINUE
WRITE(7,FMT='(1H,15X,A)')
- '表1 区分的最良近似の計算結果'
WRITE(7,FMT='(1H,15X,A)')
- 'Table 1 Results of piecewise
- best approximation'
READ(5,FMT='(A)') AAA
RETURN; END
C*****
FUNCTION F(X) !C 近似関数
REAL*8 F,X
F=EXP(-X)
RETURN; END
C*****

```

図 6 最良近似のプログラム B
Fig. 6 Program B of best approximation.

```

C*****
C* 最良近似の共通プログラム AB
C*****
C* 最良近似のサブルーチン集である
C* 切り離してライブラリとする
C*****
C
C*****
C* 各ルーチンに共通な変数を定義する
C** COM.FOR **
C*****
IMPLICIT REAL*8 (A-H,O-Z)
REAL*8 A,B,EMAX,EMIN,EPS,
- XZ(30),XS(30)
REAL*8 P(30),Q(30),Z(30,31),
- XM(30),EX(30),NW(30)
CHARACTER*55 AAA
COMMON/VR/P,Q,Z,XM,EX,NW,XZ,XS,A,B,
- EMAX,EMIN,EPS,ALPH
COMMON/VI/ L,M,N,NREP,NSKEY
COMMON/VC/ AAA
C*****
CC 【単純変数】
C* A : 区間の左端
C* B : 区間の右端
C* EMAX: 誤差の絶対値の最大値
C* EMIN: 誤差の絶対値の最小値
C* EPS : 偏差値の許容度
C* ALPH: 補間点の補正の加速率
C* L : 近似式の分子多項式の次数
C* M : 近似式分母多項式の次数
C* N : 近似式の次数 ; N=L+M
C* NREP: 反復回数に対する
C* 印刷頻度を指示する
C* NSKEY: 有理式近似の形を指示する
C* NSKEY=0: 分子分母は一般的多項式
C* NSKEY=1: 分子は奇数多項式
C* 分母は偶数多項式
C* NSKEY=2: 分子は偶数多項式
C* 分母は偶数多項式
CC 【配列変数】
C* P : 近似式の分子多項式
C* Q : 近似式分母多項式
C* Z : 連立一次方程式用
C* 2次元配列
C* XZ : 両端を含む0点;
C* XZ(1)=A;XZ(N+3)=B;
C* XZ(2) ~ XZ(N+2) は0点
C* XS : 0点間の距離;
C* XS(K)=XZ(K+1)-XZ(K);
C* K=1 ~ N+2
C* XM : 偏差点; XM(1) ~ XM(N+2)
C* EX : 偏差値; EX(1) ~ EX(N+2)
C* NW : 使用せず
C*
C*****
CC 最良近似式を求める共通主プログラム
C*****
SUBROUTINE COMBES
EXTERNAL F,E
INCLUDE 'COM.FOR'
CC CALL SHOKI !C 初期値
KREP=0 !C 収束までの反復回数
10 CONTINUE
KREP=KREP+1 !C 反復回数を数える
CALL SIKI !C 補間の式を立てる
CALL LINIA !C 式を解く
CALL HOSEI !C 式を取出す
CALL KYOKU !C 偏差点を求める
CC 印刷
IF(KREP.EQ.(KREP/NREP)*NREP)THEN
CALL GRAPH(E,F,6); CALL PRINT(6)
ENDIF
CALL ZERHO !C 零点の修正
WRITE(6,67)
- '反復回数=',KREP,' EMAX=',EMAX
CC 偏差値が等しくなければ反復する
IF(EMAX-EMIN.GT.EPS*EMAX)GOTO 10
67 FORMAT(1H ,A,I5,A,1PE13.5)
CC 収束したので印刷する
WRITE(7,*) AAA
WRITE(7,67)
- '収束するまでの反復回数=',KREP
CALL PRINT(7)
CALL GRAPH(E,F,7)
RETURN; END
C*****
FUNCTION GFUN(X) !C 補間の関数
REAL*8 F,X,GFUN
GFUN=F(X)
RETURN; END
C*****
FUNCTION SFUN(X) !C 変数変換
REAL*8 X,SFUN
SFUN=X
RETURN; END
C*****
FUNCTION E(F,X) !C 標準誤差関数
REAL*8 E,F,X,PQ
E=F(X) - PQ(X)
RETURN; END
C*****

```

図 7
Fig. 7

```

C* 初期値の設定
C*****
SUBROUTINE SHOKI
  INCLUDE 'COM.FOR'
  EPS=1.0E-3
C 最初の0点を決める
  NP=2.0*ATAN(1.0D0)/(N+1)
  A1=0.5*(A-B); A2=0.5*(A+B)
  DO 19 K=2,N+2
    XZ(K)=A1*COS((2*K-3)*NP)+A2
19 CONTINUE
  XZ(1)=A; XZ(N+3)=B; Q(1)=1.0
  RETURN; END
C*****
C 補間式を立てる
C*****
SUBROUTINE SIKI
  INCLUDE 'COM.FOR'
  DO 20 I=1,N+1
    U=XZ(I+1) !C 補間点
    FU=QFUN(U) !C 補間値
    U=SFUN(U) !C 変数変換
    IF(NSKEY.EQ.0)THEN
      U2=U; Z(I,1)=1.0 !C 一般
    ELSEIF(NSKEY.EQ.1)THEN
      U2=U*U; Z(I,1)=U !C 奇関数
    ELSEIF(NSKEY.EQ.2)THEN
      U2=U*U; Z(I,1)=1.0 !C 偶関数
    ENDIF
    IF(L.EQ.0)GOTO 21
    DO 22 J=1,L !C 分子多項式
      Z(I,J+1)=Z(I,J)*U2
22 Z(I,J+1)=Z(I,J)*U2
21 IF(M.EQ.0)GOTO 23
    Z(I,L+2)=-FU*U2
    IF(M.EQ.1)GOTO 23
    DO 24 J=2,M !C 分母多項式
      LJ=L+J
24 Z(I,LJ+1)=Z(I,LJ)*U2
23 Z(I,N+2)=FU
20 CONTINUE
  RETURN; END
C*****
C 近似式を補正する
C*****
SUBROUTINE NOSEI
  INCLUDE 'COM.FOR'
  IF(M.EQ.0)GO TO 41
  DO 42 I=1,M !C 分母取出し
    LI=L+I
42 Q(I+1)=Z(LI+1,N+2)
41 DO 40 I=1,L+1 !C 分子取出し
40 P(I)=Z(I,N+2)
    RETURN; END
C*****
C 偏差点, 偏差値を探す
C*****
SUBROUTINE KYOKU
  EXTERNAL F,E
  INCLUDE 'COM.FOR'
  REAL*8 X(9),Y(9)
  EMAX=0.0; EMIN=1.0E10
  DO 15 K=1,N+2
    DD=XZ(K+1)-XZ(K); N=DD/8.0D0
    DO 16 I=1,9
      X(I)=XZ(K)+H*(I-1); Y(I)=E(F,X(I))
16 CONTINUE
    II=1
    DO 17 I=2,9 !C 最大を探す
      IF(ABS(Y(I)).GT.ABS(Y(II)))II=I
17 CONTINUE
    XN(K)=X(II); EX(K)=Y(II)
    IF(II.EQ.1.OR.II.EQ.9)GOTO 13
    DF=Y(II+1)-Y(II-1)
    DDF=Y(II+1)-2.0*Y(II)+Y(II-1)
    XN(K)=XN(K)-0.25*N*DF/DDF !C 2次逆補間
    EX(K)=E(F,XN(K))
13 CONTINUE
    IF(ABS(EX(K)).GT.EMAX)EMAX=ABS(EX(K))
    IF(ABS(EX(K)).LT.EMIN)EMIN=ABS(EX(K))
CC WRITE(6,66) K,XN(K),EX(K)
66 FORMAT(I5,1P2E25.15)
15 CONTINUE
CC WRITE(6,67)' MAX-MIN',EMAX,EMIN
67 FORMAT(A/1PE25.15/1PE25.15)
  RETURN; END
C*****
C 補間点を決める
C*****
SUBROUTINE ZERHO
  INCLUDE 'COM.FOR'
  REAL*8 S,SV,M,DS
  DO 60 I=1,N+2 !C 距離
60 XS(I)=XZ(I+1)-XZ(I)
    S=0.0
    DO 61 I=1,N+2
61 S=S+XS(I)/(ABS(EX(I)))
    H=(B-A)/S !C 偏差値の平均
    DO 62 I=1,N+1 !C 補間点修正
      DS=XS(I)*(H/ABS(EX(I))-1.0)
62 XZ(I+1)=XZ(I)+XS(I)+DS*ALPH
  RETURN; END
C*****

```

図 7
Fig. 7

```

C 有理式を計算する
  FUNCTION PQ(X)
    << 省略 >>
C 多項式を計算する
  FUNCTION POL(A,N,X)
    << 省略 >>
C 連立1次方程式を解く
  SUBROUTINE LINIA
    << 省略 >>
C 印刷ルーチン
  SUBROUTINE PRINT(IO)
    << 省略 >>
C 出力ルーチン
  SUBROUTINE OUT(IO)
    << 省略 >>
C グラフを描く
  SUBROUTINE GRAPH(E,F,IO)
    << 省略 >>
C グラフを描く
  SUBROUTINE CURVE(N,R,G,AMAX,IO)
    << 省略 >>

```

図 7 最良近似のプログラム AB

Fig. 7 Program AB of best approximation.

の誤差曲線は図 4 である。

プログラム AB (図 7) はプログラム A および B に共通なプログラムである。新しい関数に対して最良近似を行う時は、プログラム A および B を書き換えれば、プログラム AB は書き換える必要がない。なお、プログラムは自家用の FORTRAN で書かれている。このプログラムの見方を少し説明する。

- 1) 最初に ! 記号が来るまで読み捨てる。
- 2) 1 行のマイナス記号は継続行である。
- 3) 行の途中の !C 以下は読み捨てる。
- 4) 文は 2 行以下に書く。
- 5) 行の途中の ; 記号は文を分ける。

9. ま と め

収束速度は、(3), (4) 式の仮定の適合度に依存し、まだ、いろいろ検討の余地がある。例えば、 m_i を区間によって変えてみたが、うまくいく場合もあり、時には、かえって収束が遅くなる場合もあり、一長一短であった。また、関数によっては、前述の初期零点の取り方によると、各区間に極大値が 1 個あるという仮定に反する場合があった。そのような関数の最良近似問題は計算が困難である。ここでは極大値は簡単な山登り法で求めたが、この改良は計算時間の短縮に大きく寄与する。本法は考え方が単純で、プログラムも単純な利点がある。

本論文の本質的な点は故高橋秀俊先生のヒントによる。先生が何気なく与えられたヒントは誠に貴重な結果であった。ここに心中より感謝の意を表す。

参 考 文 献

- 1) Hastings, C.: *Approximations for Digital Computers*, Princeton University Press (1955).
- 2) Hart, J. F. (ed.): *Computers Approximations*, John Wiley & Sons, Inc. (1968).
- 3) 宇野利雄: 数値計算, pp. 192-202, 朝倉書店 (1963).
- 4) 一松 信: 近似式, 竹内書店 (1963).
- 5) 高田 勝, 春海佳三郎 (編): 数値計算の手順と実際, コロナ社 (1984).
- 6) 山下眞一郎: 有理式の最良近似式を求めるプログラム, 情報処理, Vol. 10, No. 6, pp. 442-446 (1969).
- 7) 山下眞一郎: 偶関数の最良近似有理式を求めるプログラム, 情報処理, Vol. 10, No. 6, pp. 372-376 (1971).
- 8) 山下眞一郎: 奇関数の最良近似有理式を求めるプログラム, 情報処理, Vol. 10, No. 6, pp. 577-581 (1971).
- 9) 山下眞一郎: 区分的最良近似について, 京都大学・数理解析研究所・講究録 373, 数値計算のアルゴリズムの研究 (1978. 11).
- 10) 山下眞一郎: 新しい最良近似式の計算法について, 第 21 回情報処理学会全国大会論文集, pp. 941-942 (1980).

(平成 4 年 1 月 10 日受付)

(平成 4 年 6 月 12 日採録)

山下眞一郎 (正会員)

昭和 12 年宮崎県えびの市生。昭和 31 年鹿児島工業高等学校卒業。

昭和 49 年日本大学理学博士。昭和 32 年 8 月有隣電機精機(株) (日本最初の計算センタ) に入社。計算機の

保守および受託計算に従事。高次代数方程式・最良近似などを研究。昭和 38 年 FACOM 231 ALGOL コンパイラ開発。昭和 40 年 4 月現富士通(株)に移籍。昭和 42 年 5 月 FACOM 230-60 FORTRAN コンパイラ開発。昭和 50 年から科学計算ライブラリ SSL 開発。昭和 48.5~53.4 本会会誌編集委員。昭和 53.5~56.4 本誌論文編集委員。昭和 56.5~58.4 本会数値解析研究会幹事。平成 4 年 4 月日本大学理工学部教授に就任。日本数学会、日本応用数学会各会員。