

簡易型データ解析プログラムの開発と実践 II

宇 野 克 志*

Development and Practice of the Simplified Program for Data Analysis II

Katsushi UNO

要 旨

Lagrange 補間法および多項式近似法を改良して低次数で数値データを処理するためのプログラムの開発を行った。

本報では、前報の非線形最小二乗法プログラムを使用したモデルの想定から相関式を決定するまでの試行の事例を示し、更に、この度開発された方法と対比した。

開発された方法（選点 Lagrange 補間法，項数選択多項式近似法）が簡単なアルゴリズムにも拘らず確実に計算できることを明示した。

Abstract

The program was developed for processing numerical data with lower numbers of order by improving the Lagrange method for interpolation and the multinomial method for approximation.

In this paper, the process for fitting from supposition of the model to determination of the correlative equation was demonstrated by using previous program¹⁾, furthermore, its result was compared with developed methods.

It was shown that their developed methods (Lagrange method for interpolation on selective points, multinomial method for approximation on selective numbers of term) can be calculated certainly in spite of simple algorithm.

1. ま え が き

近年の科学技術の発展により新しい物質の研究開発が盛んである。しかし、これらの研究開発のための実験環境は物質の特異性と相まって次第に極端な領域に拡大し、しかも狭い範囲の厳しい制約を受けるようになっていく。

このため得られるデータが少ないことに加えてバラツキが大きく、実測や文献データをもとにした設計シミュレーションプログラムでは、これらのデータを“如何に精度よくモデル式にあてはめるか”がプログラムソフト全体の良否を左右することになる。前報では、これらのデータ処理を簡便確実に行うツール（道具）として非線形最小二乗法のプログラミングを行った¹⁾。

しかし、モデル式へのあてはめが全てのデータについて有効とは限らない。モデル式の選定に困難を極め、時には見あたらないこともあり、更にモデル式が選定できても、初期値をあてはめてみると演算中に発散したり、収束しても精度が悪かったりなどで実用に供する相関式を得るには幾度となく試行が必要である。

試行の結果、精度のよい相関式が得られないときはこれに対処する方法として、従来から行われてきた Lagrange 補間法と多項式近似法がある。しかし、Lagrange 補間法には周知の通り必ずデータ点を満足する長所があるが、その反面、（データ数－1）次の高次方程式になるため、データ点の少ない区間では振動し、発散するなど避けられない短所がある。また、多項式近似法も次数を上げるとデータ点に接近する確率は高くなるが同じ短所がある。

* 教授 工業化学科

この度、高次数による振動、発散を避けることを目的として、これらの方法の基本となる多項式の次数を可能な限り低次数であてはめるための方策を試みた。

その結果、従来の方法を改良して選点 Lagrange 補間法と項数選択多項式近似法を開発し、これらの方法が低次数でも充分満足なあてはめができることを確かめた。

本報では前報の非線形最小二乗法プログラム¹⁾を使用したモデルの想定から相関式を決定するまでの複雑な試行の事例を紹介し、更に、この度開発した方法と対比した結果について報告する。

2. 開発された方法

2-1) 選点 Lagrange 補間法

従来の Lagrange 補間法を改良して、多項式の任意の次数に応じて補間点近傍の(次数+1)点のデータを選点し、このデータ点に従来の Lagrange 法を使用する。ここで、入力する次数を(入力データ数-1)とすると従来の Lagrange 補間法になる。

2-2) 項数選択多項式近似法

従来の多項式近似法を改良して、多項式の項数と次数を任意に入力し、あてはめには線形最小二乗法を使用する。多項式の係数は Gauss の消去法により行列式を解いて求める。

これらの方法には前報¹⁾の非線形最小二乗法とグラフを描くためのルーチンを接続し、新しいプログラムソフトパック“APPROX”とした。図1に流れ図、文末にプログラムリストを示す。APPROXには従来の多項式近似法を含めて4種の関数近似法がある。これらの方法はメニューにより任意に選択でき、計算後はデータと計算値をディスプレイとプリンタに出力して、あてはめの様子を両面から比較検討できるようにした。更に、任意の数値を入力すると、その推算値をグラフに図示すると同時に数値も記号と同色で表示される。この推算は選択した方法について繰り返し試行できるようにになっている。その後、再びメニューに戻ることができる。なお、二個以上の関数を組み合わせる複雑なモデル式へのあてはめなどのために、各方法は単独でも使用できるようになっている。

〔プログラムの使用記号〕

THEME\$: 処理内容の名称

AXISX\$: グラフの横軸の名称

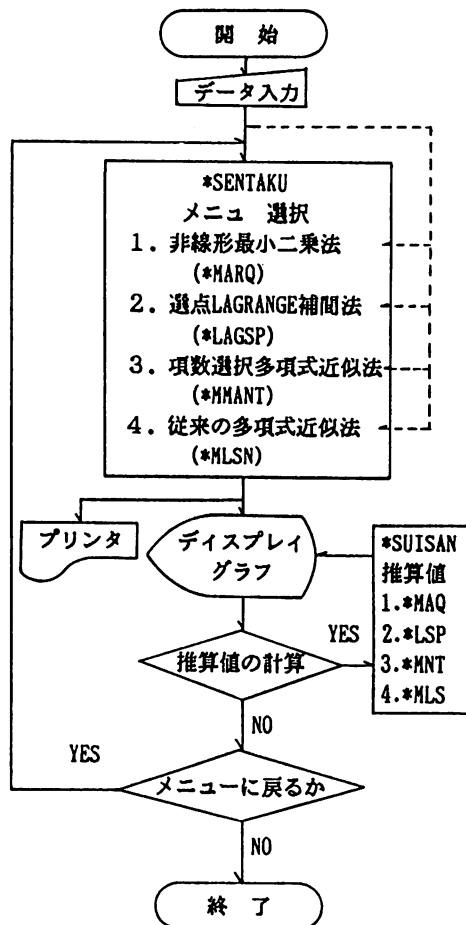


図1 APPROXの流れ図

AXISY\$: グラフの縦軸の名称

NP : 定数の数

PM : モデル式の定数の配列

実行前は初期値, 実行後は収束値

ND : データ数

XD : 独立変数の配列

YD : 従属変数の配列

データは文番号500以降に DATA 文で書き込む。

3. 使用機器

パソコンは NEC PC -9801 RA, ディスプレイは高解像度カラー仕様のものを使用した。Vm以前の機種を除けば、同じような結果が得られるはずである。

4. 処理手順と結果

下記の処理例は化学工業装置設計のための基礎

資料の作成例である。初めに、非線形最小二乗法（* MARQ）を使用したモデル式へのあてはめを行い、次いで今回開発された方法と比較する。なお、グラフを描くための計算点は入力データ数に関係なく全て41点とした。

[处理例-1]

修正 Redlich - Kwong 状態方程式 (略称, MRKEOS) により, 液体混合物の高圧気液平衡関係を相関するには混合物の各純粋成分の MRKEOS 定数 Ω_a と Ω_b が必要である。このうち, 二酸化炭素の定数 Ω_a を文献²⁾のデータより計算

したところ、対臨界温度 T_r との関係は表 1 のようになった。 Ωa を T_r の関数としてモデル式にあてはめ、更に今回開発された方法と比較してみよう。

1-1) 非線形最小二乗法によるモデル式へのあてはめ

(1) $\Omega a \rightarrow Y$, $\text{Tr} \rightarrow X$ としてデータを図示(図5)してみると, かなり複雑な曲線で単純な関数で表現できないことは明白である。

(2) データを変換して曲線の形が或る特定の関数で表し得るか否かを試行してみる。

No.	T_r	Ω_a
1	0.712032	0.432540
2	0.723203	0.431567
3	0.739645	0.430083
4	0.750082	0.428539
5	0.772518	0.426856
6	0.788955	0.425014
7	0.805391	0.423064
8	0.821828	0.420963
9	0.838264	0.418799
10	0.854701	0.416428
11	0.887574	0.411564
12	0.904010	0.409167
13	0.920447	0.406419
14	0.933320	0.401802
15	0.966193	0.400713
16	1.000000	0.412644

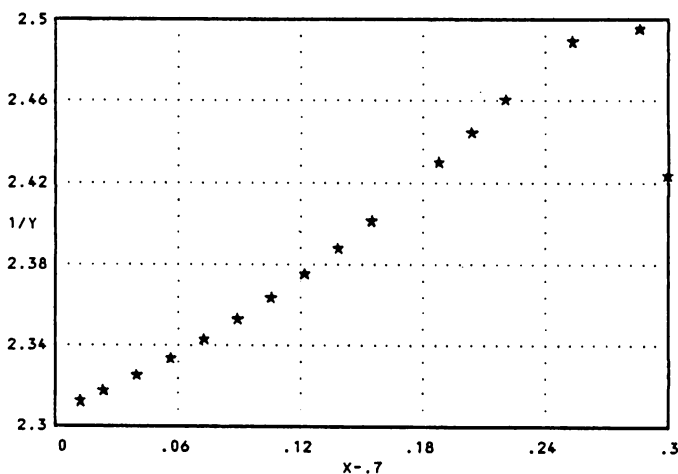


図2 修正 Redlich-Kwong 状態方程式の対臨界温度 T_r と Ω_a の関係 (CO_2)

```

200 DEF FNY(X)=(PM(1)*X+PM(2))*PM(3)+PM(4)
410 FOR J=1 TO ND : YD(J)=1/YD(J) : XD(J)=XD(J)-.7 : NEXT J
420 GOSUB *KEKKA : GOSUB *JUST : GOSUB *GRAPH5 : INPUT "ND=";ND : CLS 3
430 GOSUB *MARQ : GOSUB *JUST : CLS 3 : GOSUB *GRAPH2 : END
520 DATA "X-.7","1/Y", 16
600 DATA 4, 1, .1, .1, .1, .2,

```

***** 最速解 (D.W.Marquardt) *****

繰り返しの演算の回数 = 72

関数を呼び出した回数 = 445

残差の平方和 = 9.98826E-08

PM(1) = .637371

PM(2) = .642931

PM(3) = 6.06234
PM(4) = 8.03212

PM(λ) = 2.23818

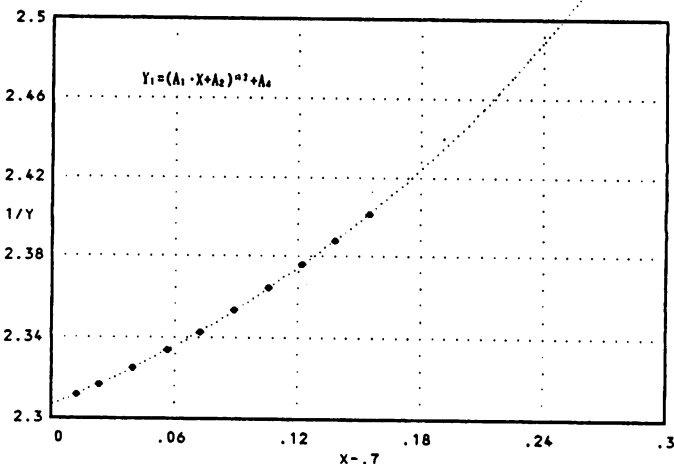


図3 修正 Redlich-Kwong 状態方程式の対臨界温度 T_r と Ω_a の関係 (CO_2)

(3) 試行の結果、 $Y = 1/Y$ 、 $X = X - 0.7$ なる変換を施すと図2のようになり、成書³⁾によると左側のかんりの部分は文番号200の関数で表せる。文番号600はその初期値である。全データ数=16のうち、左側のデータ数=10について* MARQによるあてはめを行うと図3のような最適解が得られた。

この場合には首尾よく収束したが、時にはモデ

ル式に確信があっても収束しないことがある。その時は先ず初期値を変えて試行する。それでもなお収束しない時は文番号2020の制御変数をかえてみる。すなわち、FLAM (収束勾配) と DX (数値微分の幅) で、両値とも0.1~0.00001の範囲である。文末のプログラムリストは標準値で、この数値を変えることは余り薦めない。

(4) (3)で求めた式からデータを差し引いた残余

```
210 DEF FNY1(X)=(.637371*X+.642931)^6.06234+2.23818
230 DEF FNY(X)=PM(1)/(PM(2)*X+PM(3))+PM(4)
410 FOR J=1 TO ND: YD(J)=1/YD(J): XD(J)=XD(J)-.7: NEXT J
440 FOR J=1 TO ND: YD(J)=FNY1(XD(J))-YD(J):XD(J)=.3-XD(J):NEXT J: GOSUB *SHIFT
450 GOSUB *KEKKA: GOSUB *JUST: GOSUB *GRAPH5: INPUT "ND=":ND:CLS 3
460 GOSUB *MARQ: GOSUB *JUST:CLS 3: GOSUB *GRAPH2:END
630 DATA "1-X","1/Y", 16
630 DATA 4, 1., 1., 1., 1., .2
***** 最適解 (D.W.Marquardt) *****
繰り返し演算の回数 = 14
関数を呼び出した回数 = 85
残差の平方和 = 1.07501E-06
PM( 1 ) = .265528
PM( 2 ) = 169.555
PM( 3 ) = 1.668609
PM( 4 ) = -.0112715
```

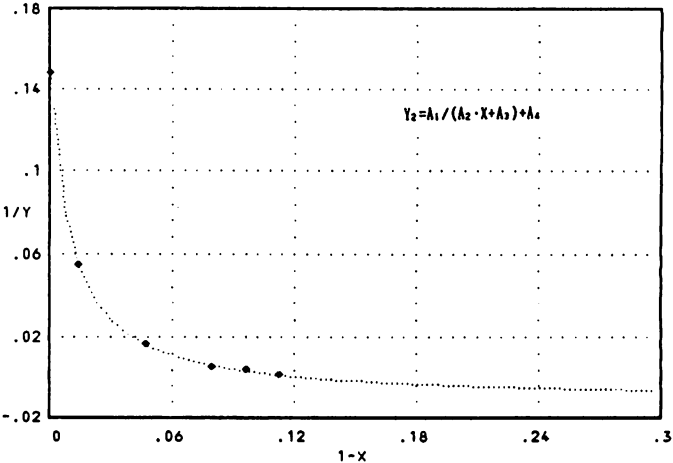


図4 修正 Redlich-Kwong 状態方程式の対臨界温度 Tr と Ωa の関係 (CO_2)

```
270 DEF FNY(X)=1/((PM(1)*X+PM(2))^PM(3)-PM(4)/(PM(5)*X+PM(6))+PM(7))
490 GOSUB *MARQ: GOSUB *SUISAN:END
540 DATA "X","Y", 16
670 DATA 7, .637371, .196771, 6.06234, .265528,-169.555, 171.224, 2.22691
***** 最適解 (D.W.Marquardt) *****
繰り返し演算の回数 = 21
関数を呼び出した回数 = 200
残差の平方和 = 6.81429E-08
PM( 1 ) = .662697
PM( 2 ) = .166287
PM( 3 ) = 5.60053
PM( 4 ) = .286347
PM( 5 ) = -169.327
PM( 6 ) = 171.084
PM( 7 ) = 2.23656
```

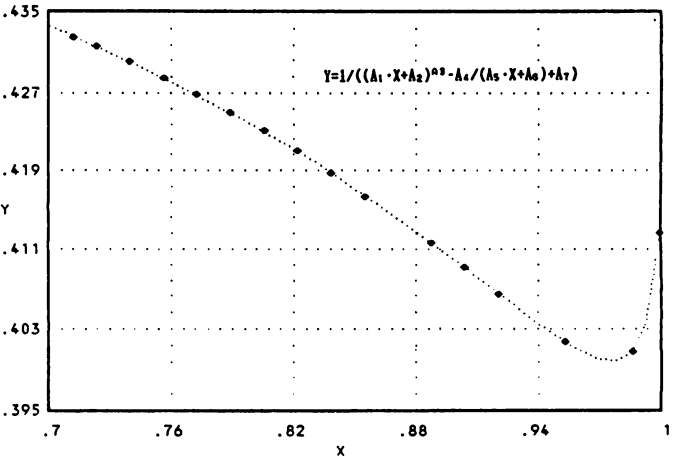


図5 修正 Redlich-Kwong 状態方程式の対臨界温度 Tr と Ωa の関係 (CO_2)

表 2

次数	誤差
3	.8461E-04
4	.8105E-04
5	.6587E-04
6	.5955E-04
7	.6189E-04
8	.2869E-04
9	.2849E-04
10	.6100E-02
11	.6241E-03
12	.2361E-04
13	.3020E-04
14	.3566E-04
15	.1662E-04
16	.1672E-04
17	.1958E-04
18	.2214E-04
19	.7236E-05
20	.8751E-05
21	.1310E-04
22	.4589E-04
23	.7805E-05
24	.3753E-05
25	.4433E-05
26	.1703E-04
27	.1580E-04
28	.3395E-04
29	.2975E-03
30	.1137E-04

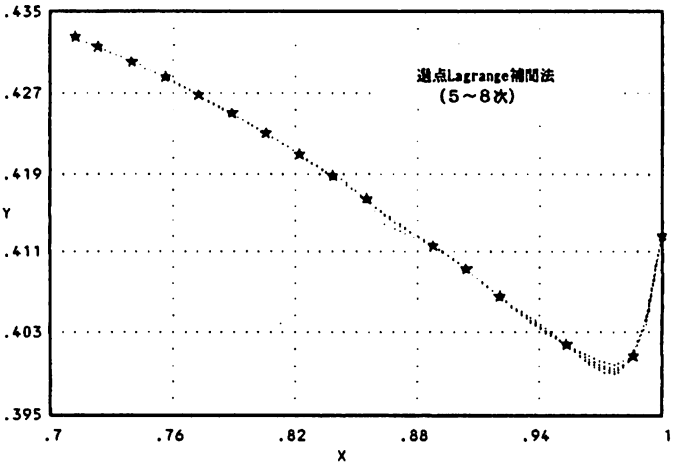


図 6 修正 Redlich-Kwong 状態方程式の対臨界温度 T_r と Ωa の関係 (CO_2)

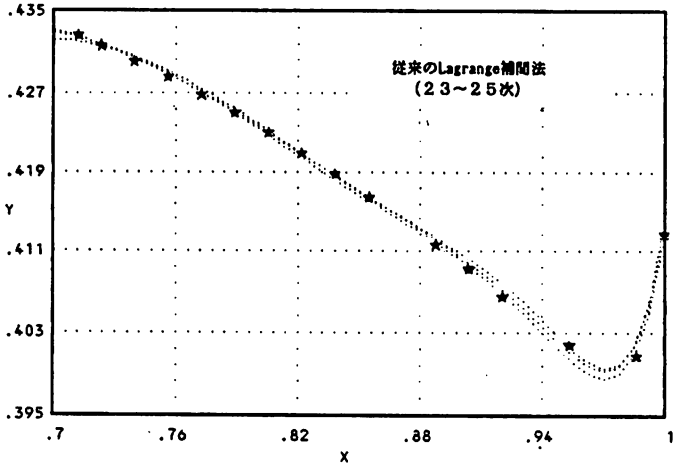


図 7 修正 Redlich-Kwong 状態方程式の対臨界温度 T_r と Ωa の関係 (CO_2)

表 3

No.	温度	体積
1	0	28.6
2	10	22.4
3	15	20.1
4	20	18.3
5	25	16.7
6	30	15.4
7	40	13.2
8	50	11.4
9	60	9.8
10	80	6.0
11	100	0.0

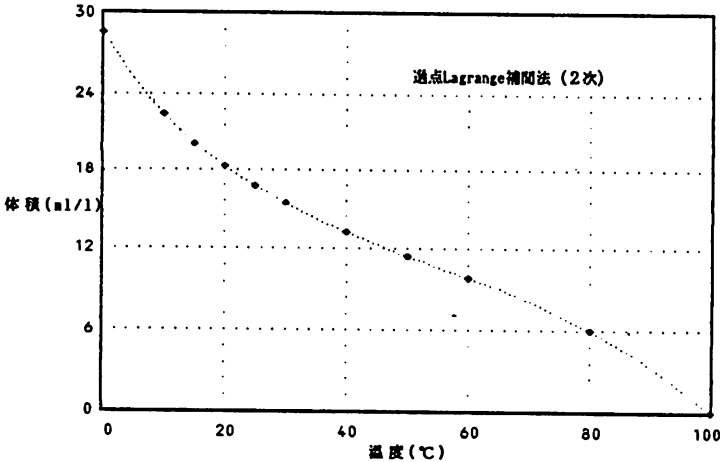


図 8 大気圧 (760mm Hg) と平衡にある水中の溶存空気

について $X = 0.3 - X$ なる変換を行うと、左側の部分は文番号230の関数で表せる。変換後はデータの並びの方向が逆になるので * SHIFT により再び昇順に並び換えてから、文番号630を初期値として左側のデータ数=6について、前回同様、* MARQ によるあてはめを行った。図4が収束結果である。

(5) ここで、(3)および(4)で得られた二つの関数を組み合わせると文番号270になり、更に得られた定数を関数に合わせて整理すると文番号670になる。これらについて再び * MARQ によるあてはめを行うと図5のような最終解になり、非常に良い精度であてはめられている。

1-2) 選点 Lagrange 補間法によるあてはめ

“APPROX” のメニューで番号2 (* LAGSP) を選択し5次から8次までの次数について重ね書きしたのが図6である。この方法は必ずデータ点を満足するので、全体のあてはめの良否はグラフによる視的判断が欠かせない。低次では不安定で、5次から8次でほぼ平滑な曲線になり、それ以上では再び不安定になる。6次のとき、図中 $X = 0.87$ 付近でズレている。

1-3) 従来の多項式近似法によるあてはめ

表2はメニュー番号4 (* MLSN) を選択し、3~30次までのあてはめを行った誤差(残差の平方和)である。図7は精度の良い23~25次を計算した結果である。前の二方法に較べるとその差は一目瞭然であろう。

[処理例-2]

表3は大気圧(760mm Hg)と平衡にある水中

の溶存空気 (ml/l) と温度 (°C) の関係である⁴⁾。APPROX のいずれかの方法によりあてはめてみよう。

2-1) 選点 Lagrange 補間法によるあてはめ

図8は次数2であてはめた結果で、殆ど完璧に近く一致している。

2-2) 他の方法によるあてはめ

非線形最小二乗法および従来の多項式近似法によるあてはめは3次方程式が最良で、かなり良い精度であったが、選点 Lagrange 補間法には及ばなかった。

[処理例-3]

プラスチック円筒にアルミナ球の入った充填塔がある。この塔の下方から水を供給し、上方より流出する流量とその間の差圧をマンメータにて測定した。この測定値を流速 u (m/s) と差圧 P (kg/m²) に換算したところ表4のようになった。Ergun⁵⁾の実験式 $P = A_1 \cdot u + A_2 \cdot u^2$ の係数 A_1 , A_2 を決定してみよう⁶⁾。

3-1) 項数選択多項式近似法によるあてはめ

図9と表5は入力した項数と次数による計算結果で、非常に良くあてはめられている。

3-2) 非線形最小二乗法によるあてはめ

表6はあてはめによる係数と計算結果を示したもので、3-1)の結果とほぼ完全に一致した。

5. ま と め

前報では数値データを処理する一方法としてモデル式へのあてはめ(相関式の作成)を取りあげ、

表4

No.	流速	差圧
1	0.0028	42.41
2	0.0054	81.78
3	0.0076	119.49
4	0.0091	145.97
5	0.0108	186.91
6	0.0118	199.99
7	0.0145	271.62
8	0.0153	271.10
9	0.0176	345.97
10	0.0191	347.90
11	0.0199	425.79
12	0.0216	421.68
13	0.0223	473.13
14	0.0236	481.06
15	0.0241	520.92
16	0.0248	576.31
17	0.0270	636.58
18	0.0270	547.37
19	0.0273	625.62
20	0.0290	614.58
21	0.0298	644.46

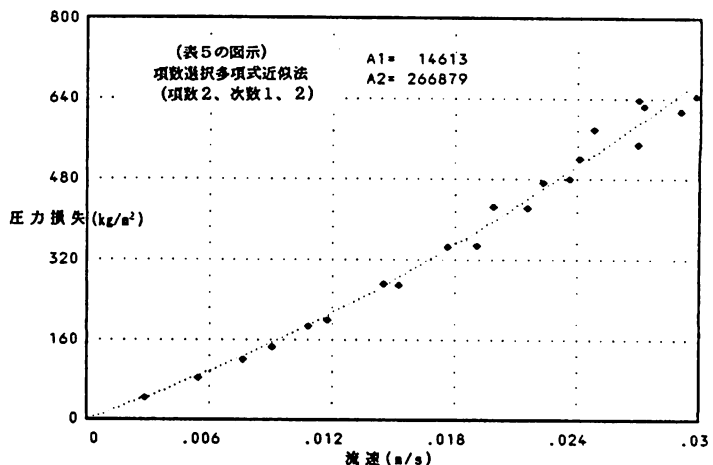


図9 充填塔の圧力損失(固液系): アルミナー水

表5 項数選択多項式近似法の計算値と誤差

項数選択多項式近似法／多項式の項数を入力して下さい。？2
項の次数を入力して下さい。

PM(1)=? 1

PM(2)=? 2

8.59486E-03 2.05253E-04 180.375

2.05253E-04 5.13203E-06 4.36901

Constant (1JI)= 14613

Constant (2JI)= 266879

No.	u	Pcal.	Pdef.	No.	u	Pcal.	Pdef.	No.	u	Pcal.	Pdef.
1	0.0028	43.17	0.76	8	0.0153	285.60	14.50	15	0.0241	507.46	-13.46
2	0.0054	87.39	5.61	9	0.0176	340.10	-5.87	16	0.0248	527.10	-49.21
3	0.0076	127.22	7.73	10	0.0191	375.97	28.07	17	0.0270	588.82	-47.76
4	0.0091	154.11	8.14	11	0.0199	395.48	-30.31	18	0.0270	588.82	41.45
5	0.0108	189.36	2.45	12	0.0216	439.37	17.69	19	0.0273	596.38	-29.24
6	0.0118	208.97	8.98	13	0.0223	458.85	-14.28	20	0.0290	649.43	34.85
7	0.0145	268.45	-3.17	14	0.0236	493.78	12.72	21	0.0298	672.47	28.01

表6 非線形最小二乗法の計算値と誤差

No.	u	Pcal.	Pdef.	No.	u	Pcal.	Pdef.	No.	u	Pcal.	Pdef.
1	0.0028	43.17	0.76	8	0.0153	285.60	14.50	15	0.0241	507.46	-13.46
2	0.0054	87.40	5.62	9	0.0176	340.10	-5.87	16	0.0248	527.10	-49.21
3	0.0076	127.22	7.73	10	0.0191	375.98	28.08	17	0.0270	588.82	-47.77
4	0.0091	154.11	8.14	11	0.0199	395.48	-30.31	18	0.0270	588.82	41.44
5	0.0108	189.36	2.45	12	0.0216	439.37	17.69	19	0.0273	596.38	-29.24
6	0.0118	208.97	8.98	13	0.0223	458.85	-14.28	20	0.0290	649.42	34.84
7	0.0145	268.45	-3.17	14	0.0236	493.78	12.72	21	0.0298	672.46	28.00

繰り返し演算の回数 = 5

関数を呼び出して誤差を計算した回数 = 24

残差の平方和 = 12340.7

PM(1)= 14613.6

PM(2)= 266854

そのためのツールとして Marquardt の非線形最小二乗法のプログラミングを行った。更に、簡単な例題を処理し、このプログラムによって敏速で確実なあてはめができることを示した。勿論、この方法は〔処理例-1〕のようになかなか複雑な処理も可能であるが、データ処理は多岐にわたり、総てがこの方法で解決する訳ではない。

本報はモデル式が選定できないか、或はモデル式へのあてはめの精度が上がらないときの対処方法について検討したものである。検討された方法の主要部分は従来から使用されてきた Lagrange 補間法と多項式の線形最小二乗法である。しかし、これらの方法には高次方程式になると振動・発散する短所があるため、この短所を改良して低次数であてはめが可能な選点 Lagrange 補間法と項数選択多項式近似法を開発した。これに前報の非線形最小二乗法と従来の多項式近似法を加えて新しいソフトパック“APPROX”とした。

開発された方法により文献および実測データを処理したところ、非常に良い精度で確実なあてはめができることが分かった。

しかし、先にも述べたようにデータを図示して見ると色々な形状（データ数、データ点の上下左

右の間隔（バラツキの度合）、平滑さなど）があり、それに応じた近似法を何度も試行する必要がある。

以上のことを総合すると、今回開発された“APPROX”は広範な数値データ解析に対処できるツールとして、手軽にしかも確実に使用できる筈である。

参考文献

- 1) 宇野克志, 苫小牧工業高等専門学校紀要第28号, 79頁 (1993)
- 2) Perry, H. R., Chilton, H. C., 3-162, Tab. 3-219, Chemical engineer's hand book 5th ed., McGraw-Hill Kogakusha, Ltd. (1973)
- 3) 曲線・グラフ総覧, 聖文社 (1986)
- 4) 日本化学会編, 化学便覧 基礎編Ⅱ (改訂2版), 770頁, 表7.75, 丸善 (1975)
- 5) Ergun, S., Chem. Eng. Prog., 48, 89 (1952)
- 6) 平沼充安, 木村規矩也, 宇野克志, 本間清士, 平野博人, 遠藤俊二, 化学工学実験書 (補遺), 苫小牧工業高等専門学校 工業化学科 (平成5年度版)

(平成5年11月30日受理)

```

100 'ファイル名 "APPROX"
110 '苫小牧工業高等専門学校 工業化学科 宇野 克志
120 'Programed by Katsushi UNO : Up to date (Nov.25 1993)
130 LPRINT "Date =";DATES;:LPRINT "Time =";TIMES
140 CONSOLE 0,25,0,1 : WIDTH 80,25 : SCREEN 3 : CLS 3
150 DIM PM(11),PMX(11),XM(11),DXM(11),SAM(11),AM(11,11),P(5,2),Q(5,2)
160 DIM AMM(11,12),FM(51),FMX(51),AJ(51,11),XD(51),YD(51),YC(51)
170 DIM XL(101),YL(101),XJ(51),YJ(51),R(31,31),S(31),KP(31),PN(31)
270 DEF FNY(X)=1/((PM(1)*X+PM(2))^PM(3)-PM(4)/(PM(5)*X+PM(6))+PM(7))
300 GOSUB *TITLE : RESTORE 500
310 NL=41 : ' GRAPHICSのグラフに推算値を表示するための独立変数のデータ数
330 CLS 3 : READ THEMES$,AXISX$,AXISY$,ND
340 READ NP : FOR J=1 TO NP : READ PM(J) : NEXT J
350 FOR J=1 TO ND : READ XD(J) : NEXT J : FOR J=1 TO ND : READ YD(J) : NEXT J
360 READ XMIN,XMAX,YMIN,YMAX
370 CLS 3 : FOR J=1 TO ND : PRINT USING "###";J;:PRINT ;SPC(5);
380 PRINT USING "###.#####";XD(J);YD(J) : NEXT J
500 '----- data -----
900 GOSUB *JUST : GOSUB *SENTAKU
980 LOCATE 0,0 : END
1000 *TITLE
1010 CONSOLE 0,25,0,1:WIDTH 40,25:SCREEN 3:CLS 3
1020 LINE (100,70)-(540,170),7,BF,7 : COLOR 2
1030 LOCATE 10, 6 : PRINT "データ解析プログラム" : COLOR 4
1040 LOCATE 12, 9 : PRINT "( APPROX )" : COLOR 6
1050 LOCATE 13,15 : PRINT "実験式の作成と"
1060 LOCATE 15,17 : PRINT "グラフ表示" : COLOR 5
1070 LOCATE 10,23 : PRINT "スペースキーを押して下さい。";
1080 WHILE INKEY$<>" ":WEND: CLS 3 : WIDTH 80,25 : COLOR 7 : RETURN
1100 *SENTAKU
1190 CLS 3 : LINE (150,100)-(500,233),7,BF
1200 LOCATE 20,4 : COLOR 4 : PRINT "---- モデル式のあてはめ ----"
1210 LOCATE 20, 7 : COLOR 2 : PRINT "1. Marquardtの非線形最小二乗法"
1220 LOCATE 20, 9 : PRINT "2. 選点Lagrange補間法"
1230 LOCATE 20,11 : PRINT "3. 項数選択多項式近似法"
1240 LOCATE 20,13 : PRINT "4. 従来の多項式近似法"
1250 LOCATE 20,18 : COLOR 6
1260 PRINT "番号を入力して下さい。";: NO$=INPUT$(1) : PRINT NO$ : COLOR 7
1270 CLS 3 : ON VAL(NO$) GOSUB *MARQ,*LAGSP,*MMANT,*MLSN
1280 GOSUB *SUISAN : LOCATE 30,0 : PRINT "別の方法で行いますか。(Y/N) ";
1290 SG$=INPUT$(1) : PRINT SG$ : IF SG$="Y" OR SG$="y" THEN CLS 3 : GOSUB 1190
1300 RETURN
1320 *KEKKA
1330 NK=2 : NN=INT((ND*NK-1)/NK) : PRINT THEMES$
1340 FOR J=1 TO NK : PRINT SPC(3);"No. X Y Ycal Ydef";: NEXT J
1350 FOR J=1 TO NN : PRINT : FOR K=1 TO NK : L=(K-1)*NN+J
1360 IF L>ND THEN PRINT " " : GOTO 1390
1370 PRINT USING "###";L;: PRINT USING "###.#####";XD(L);
1380 PRINT USING "###.#####";YD(L);YC(L);: PRINT USING "###.#####";FM(L);
1390 NEXT K,J : PRINT : RETURN
1450 *GRAPH2
1460 CLS 3 : COLOR 7 : GOSUB *FRAME
1470 CR=2 : LG=3 : GOSUB *XYPLOT : GOSUB *JUST : GOSUB *XYLINE : RETURN
1500 *GRAPH5 : CLS 3 : COLOR 7 : GOSUB *MAXMIN : PRINT "MAX,MIN=";MIN,MAX
1510 INPUT "YMIN,YMAX="; YMIN,YMAX : PRINT "X(1),X(ND)=";XD(1),XD(ND)
1520 INPUT "XMIN,XMAX="; XMIN,XMAX : CLS 3 : GOSUB *FRAME : LG=3 : CR=1
1530 FOR J=1 TO ND : XP=XD(J) : YP=YD(J) : GOSUB *CROSS : NEXT J : RETURN
1600 *SUISAN
1610 GOSUB *JUST : CLS 3 : GOSUB *GRAPH2
1620 LOCATE 15,18 : PRINT "作成した式による推算を行いますか。(Y/N) ";
1630 SG$=INPUT$(1) : PRINT SG$ : IF SG$="N" OR SG$="n" THEN RETURN
1640 LOCATE 15,18 : INPUT "Xの数値を入力して下さい。";XP
1650 ON VAL(NO$) GOSUB *MAQ,*LSP,*MNT,*MLS
1660 COLOR CR : LOCATE 50,3 : PRINT "X = ";XP : LOCATE 50,5:PRINT "Y = ";YP
1670 LG=3 : GOSUB *CROSS : COLOR 7
1680 LOCATE 15,18 : PRINT " ";
1690 LOCATE 15,18 : PRINT "入力をつづけますか。(Y/N) ";
1700 SG$=INPUT$(1) : PRINT SG$ : IF SG$="Y" OR SG$="y" THEN 1640
1710 LOCATE 15,18 : PRINT " ";
1720 RETURN
1740 *JUST
1750 LOCATE 53,0 :PRINT "スペースキーを打って下さい";:WHILE INKEY$<>" ":WEND
1760 LOCATE 53,0 :PRINT " ";: RETURN
1800 *MAXMIN
1810 MAX=YD(1) : MIN=YD(1)
1820 FOR L=2 TO ND : IF YD(L)<MIN THEN MIN=YD(L)

```



```

1830 IF YD(L)>MAX THEN MAX=YD(L)
1840 NEXT L : RETURN
1860 *SHIFT
1870 MADE=ND
1880 FOR J=2 TO MADE : IF XD(J-1)<XD(J) THEN 1910
1890 W=XD(J-1) : XD(J-1)=XD(J) : XD(J)=W : W=YD(J-1) : YD(J-1)=YD(J) : YD(J)=W
1900 W=YC(J-1) : YC(J-1)=YC(J) : YC(J)=W : W=FM(J-1) : FM(J-1)=FM(J) : FM(J)=W
1910 NEXT J : MADE=MADE-1 : IF MADE>=2 THEN 1880
1920 RETURN
2000 *MARQ : NOS="1" : PRINT "-----"
2010 PRINT "初期条件"
2020 EPS=.000001 : TOL=EPS : AINC=10 : DEC=.1 : FLAM=.001 : DX=.001
2030 PRINT "収束判定基準=" ; EPS
2040 FOR J=1 TO NP : PRINT "PM(";J;")=" ; PM(J) : NEXT J
2050 PRINT "-----" : CR=2
2060 NP1=NP+1 : NI=0 : NF=0 : GOSUB *FUNC
2070 NI=NI+1 : SS=SS : FLAM=FLAM*DEC : IF FLAM<TOL THEN FLAM=TOL
2080 PRINT "繰り返し演算の回数=" ; NI
2090 FOR J=1 TO NP : PRINT "PM(";J;")=" ; PM(J) : NEXT J
2100 PRINT "残差の平方和=" ; COLOR CR : PRINT SS : CR=CR+1 : COLOR 7
2110 PRINT "-----" : IF CR>=7 THEN CR=2
2120 FOR J=1 TO ND : FMX(J)=FM(J) : NEXT J
2130 FOR J=1 TO NP : PMX(J)=PM(J) : DM=(ABS(PM(J))+DX)*DX : PM(J)=PM(J)+DM
2140 GOSUB *FUNC : FOR I=1 TO ND : AJ(I,J)=(FM(I)-FMX(I))/DM : NEXT I
2150 PM(J)=PM(J)-DM : NEXT J
2160 FOR I=1 TO NP : XM(I)=0 : FOR J=1 TO ND : XM(I)=XM(I)+AJ(J,I)*FMX(J) : NEXT J
2170 FOR J=1 TO NP : AM(I,J)=0
2180 FOR K=1 TO ND : AM(I,J)=AM(I,J)+AJ(K,I)*AJ(K,J) : NEXT K,J,I
2190 FOR I=1 TO NP : SAM(I)=SQR(AM(I,I)) : NEXT I
2200 FOR I=1 TO NP : XM(I)=XM(I)/SAM(I)
2210 FOR J=1 TO NP : AM(I,J)=AM(I,J)/(SAM(I)*SAM(J)) : NEXT J,I
2220 FOR I=1 TO NP : AMM(I,NP1)=-XM(I)
2230 FOR J=1 TO NP : AMM(I,J)=AM(I,J) : NEXT J : AMM(I,I)=AMM(I,I)+FLAM : NEXT I
2240 FOR I=1 TO NP : I1=I+1 : FOR J=I1 TO NP1 : AMM(I,J)=AMM(I,J)/AMM(I,I) : NEXT J
2250 FOR J=1 TO NP : IF I=J THEN 2270
2260 FOR K=I1 TO NP1 : AMM(J,K)=AMM(J,K)-AMM(I,K)*AMM(J,I) : NEXT K
2270 NEXT J,I
2280 FOR I=1 TO NP : DXM(I)=AMM(I,NP1)/SAM(I) : PM(I)=PMX(I)+DXM(I) : NEXT I
2290 GOSUB *FUNC : IF NF >= 2000 THEN 2350
2300 IF SS<SS0 THEN 2330
2310 FLAM=FLAM*AINC : IF FLAM > 1E+10 THEN PRINT " OVER " : GOTO 2350
2320 GOTO 2220
2330 MQ=0 : FOR J=1 TO NP : IF ABS(DXM(J))/(ABS(PM(J))+DX)>EPS THEN MQ=MQ+1 : NEXT J
2340 IF MQ=0 THEN 2350 ELSE 2070
2350 PRINT SPC(20);"***** 最適解 (D.W.Marquardt) *****"
2370 FOR J=1 TO NL : XL(J)=XMIN+(J-1)*(XMAX-XMIN)/(NL-1) : YL(J)=FNY(XL(J)) : NEXT J
2380 GOSUB *KEKKA : PRINT "繰り返し演算の回数=" ; NI
2390 PRINT "関数を呼び出して誤差を計算した回数=" ; NF
2400 PRINT "残差の平方和=" ; SS
2410 FOR J=1 TO NP : PRINT "PM(";J;")=" ; PM(J) : NEXT J
2420 RETURN
2440 *MAQ : YP=FNY(XP) : RETURN
2500 *FUNC : SS=0!
2510 FOR K=1 TO ND : YC(K)=FNY(XD(K))
2520 FM(K)=YC(K)-YD(K) : SS=SS+FM(K)^2 : NEXT K : NF=NF+1 : RETURN
2600 *LAGSP
2610 NOS="2" : PRINT "選点 Lagrange 補間法/最高次数を入力して下さい。";
2620 INPUT JS : JS=JS+1
2640 FOR K=1 TO NL : XL(K)=XD(1)+(K-1)*(XD(ND)-XD(1))/(NL-1) : XP=XL(K)
2720 GOSUB *LSP : YL(K)=YP : NEXT K : GOSUB *KEKKA : RETURN
2770 *LSP
2780 FOR K0=1 TO ND : IF XP<XD(K0) THEN 2790 ELSE NEXT K0
2790 K1=K0-2 : IF K0>(ND-JS+2) THEN K1=ND-JS
2800 FOR K2=1 TO JS : K3=K1+K2 : XJ(K2)=XD(K3) : YJ(K2)=YD(K3) : NEXT K2
2810 PH=0!
2820 FOR K1=1 TO JS : B=1!
2830 FOR K2=1 TO JS : IF K1=K2 THEN 2850
2840 IF XJ(K1)-XJ(K2)>0 THEN B=B*(XP-XJ(K2))/(XJ(K1)-XJ(K2))
2850 NEXT K2 : PH=PH+YJ(K1)*B : NEXT K1 : YP=PH : RETURN
2900 *MMANT : NOS="3"
2910 PRINT "項数選択多項式近似法/多項式の項数を入力して下さい。";
2920 INPUT KSU : PRINT "各項の次数を入力して下さい。"
2930 FOR J=1 TO KSU : PRINT "PN(";J;")=" ; INPUT PN(J) : NEXT J
2940 FOR J=1 TO KSU : S(J)=0 : FOR K=1 TO KSU : R(J,K)=0 : NEXT K,J
2950 FOR J=1 TO KSU : FOR K=1 TO ND : S(J)=S(J)+YD(K)*XD(K)^PN(J)
2960 FOR L=1 TO KSU : R(J,L)=R(J,L)+XD(K)^PN(J)+PN(L) : NEXT L,K,J

```

```

2970 FOR J=1 TO KSU : FOR L=1 TO KSU : PRINT R(J,L);: NEXT L:PRINT S(J):NEXT J
2980 FOR K=1 TO KSU : A=R(K,K) : FOR J=1 TO KSU : R(K,J)=R(K,J)/A : NEXT J
2990 S(K)=S(K)/A : FOR L=1 TO KSU : IF L=K THEN 3010
3000 B=R(L,K) : FOR J=K TO KSU:R(L,J)=R(L,J)-B*R(K,J):NEXT J : S(L)=S(L)-B*S(K)
3010 NEXT L,K : SS=0
3020 FOR J=1 TO ND:YC(J)=0:FOR K=1 TO KSU:IF XD(J)=0 THEN YC(J)=S(1):GOTO 3040
3030 YC(J)=YC(J)+S(K)*XD(J)^PN(K) : NEXT K
3040 FM(J)=YC(J)-YD(J) : SS=SS+FM(J)^2 : NEXT J
3050 FOR J=1 TO KSU:PRINT USING " Constant ( ##J1 )=";PN(J);:PRINT S(J):NEXT J
3070 FOR J=1 TO NL : XL(J)=XMIN+(J-1)*(XMAX-XMIN)/(NL-1) : NEXT J
3080 FOR J=1 TO NL : YL(J)=0
3090 FOR K=1 TO KSU : IF XL(J)=0 AND PN(K)=0 THEN YL(J)=S(K):GOTO 3110
3100 YL(J)=YL(J)+S(K)*XL(J)^PN(K) : NEXT K
3110 NEXT J : GOSUB *KEKKA : RETURN
3130 *MNT
3140 YP=0
3150 FOR K=1 TO KSU : IF XP=0 AND PN(K)=0 THEN YP=S(K):GOTO 3170
3160 YP=YP+S(K)*XP^PN(K) : NEXT K
3170 RETURN
3200 *MLSN : NOS="4"
3210 LOCATE 0,0 :PRINT "従来の多項式近似法/最高次数を入力して下さい。";
3220 INPUT JSU : JSU=JSU+1
3230 FOR J=1 TO JSU : S(J)=0 : KP(J)=J-1 : FOR K=1 TO JSU : R(J,K)=0 : NEXT K,J
3240 FOR J=1 TO JSU
3250 FOR K=1 TO ND : IF XD(K)>0 THEN S(J)=S(J)+YD(K)*XD(K)^KP(J) : GOTO 3270
3260 IF KP(J)=0 THEN S(J)=S(J)+YD(K)
3270 FOR L=1 TO JSU : RJL=R(J,L)+XD(K)^KP(J)+KP(L))
3280 IF XD(K)>0 THEN R(J,L)=RJL : GOTO 3300
3290 KPP=KP(J)+KP(L) : IF KPP=0 THEN R(J,L)=R(J,L)+1
3300 NEXT L,K,J
3310 FOR K=1 TO JSU:A=R(K,K):FOR J=K TO JSU:R(K,J)=R(K,J)/A:NEXT J:S(K)=S(K)/A
3320 FOR I=1 TO JSU : IF I=K THEN GOTO 3340
3330 B=R(I,K):FOR J=K TO JSU:R(I,J)=R(I,J)-B*R(K,J):NEXT J:S(I)=S(I)-B*S(K)
3340 NEXT I,K : SS=0
3350 FOR J=1 TO ND:YC(J)=0:FOR I=1 TO JSU:IF XD(J)=0 THEN YC(J)=S(1):GOTO 3370
3360 YC(J)=YC(J)+S(I)*XD(J)^KP(I) : NEXT I
3370 FM(J)=YC(J)-YD(J) : SS=SS+FM(J)^2 : NEXT J
3380 FOR J=1 TO JSU : PRINT USING " Constant ( ##J1 )=";J-1;:PRINT S(J):NEXT J
3400 FOR J=1 TO NL : XL(J)=XMIN+(J-1)*(XMAX-XMIN)/(NL-1) : NEXT J
3410 FOR J=1 TO NL:YL(J)=0:FOR I=1 TO JSU:IF XL(J)=0 THEN YL(J)=S(1):GOTO 3430
3420 YL(J)=YL(J)+S(I)*XL(J)^KP(I) : NEXT I
3430 NEXT J : GOSUB *KEKKA : RETURN
3450 *MLS : YP=0
3460 FOR I=1 TO JSU : IF XP=0 THEN YP=S(1) : GOTO 3480
3470 YP=YP+S(I)*XP^KP(I) : NEXT I
3480 RETURN
4000 *FRAME
4010 WINDOW (0,0)-(639,399)
4020 LINE (87,22)-(567,342),7,B : LINE (86,21)-(568,343),7,B
4030 FOR Y1=86 TO 278 STEP 64 : DELX=(567-87)/50
4040 FOR X1=1 TO 49 : PSET (87+X1*DELX,Y1),7 : NEXT X1,Y1
4050 FOR X1=183 TO 471 STEP 96 : DELY=(342-22)/25
4060 FOR Y1=1 TO 24 : PSET (X1,22+Y1*DELY),7 : NEXT Y1,X1
4070 DY=(YMAX-YMIN)/5 : GY=YMIN
4080 FOR YJ=21 TO 1 STEP -4 : GJ=LEN(STR$(GY))
4090 IF GJ>10 THEN LOCATE 1,YJ : PRINT USING "###.##^";GY : GOTO 4110
4100 GJ=10-GJ : LOCATE GJ,YJ : PRINT GY
4110 GY=GY+DY : NEXT YJ : DX=(XMAX-XMIN)/5 : GX=XMIN
4120 FOR XJ=11 TO 71 STEP 12 : GJ=LEN(STR$(GX))
4130 IF GJ>10 THEN LOCATE XJ-4,22 : PRINT USING "###.##^";GX : GOTO 4150
4140 GJ=XJ-GJ+1 : LOCATE GJ,22 : PRINT GX
4150 GX=GX+DX : NEXT XJ
4160 LOCATE 6,11 : PRINT AXISYS : LOCATE 39,23 : PRINT AXISXS
4170 GJ=LEN(THMES) : LOCATE 40-GJ/2,24 : PRINT THMES;
4180 GX=480/(XMAX-XMIN) : GY=320/(YMAX-YMIN) : RETURN
4200 *XYLINE : ***** drawing line *****
4210 X1=(XL(1)-XMIN)*GX+87 : Y1=342-(YL(1)-YMIN)*GY : PSET (X1,Y1),CR
4220 FOR J1=2 TO NL : X1=(XL(J1)-XMIN)*GX+87 : Y1=342-(YL(J1)-YMIN)*GY
4230 LINE -(X1,Y1),7,,8H4444 : NEXT J1 : RETURN
4250 *XPLOT : ***** plotting data *****
4260 FOR J1=1 TO ND : X1=(XD(J1)-XMIN)*GX+87 : Y1=342-(YD(J1)-YMIN)*GY
4270 CIRCLE (X1,Y1),LG,CR,,,5,F,7 : NEXT J1 : RETURN
4300 *CROSS : ***** cross point *****
4310 X1=(XP-XMIN)*GX+87 : Y1=342-(YP-YMIN)*GY
4320 LINE (X1-LG,Y1-1)-(X1+LG,Y1+1),7,BF,CR
4330 LINE (X1-1,Y1-LG)-(X1+1,Y1+LG),7,BF,CR : CR=CR+1 : IF CR>6 THEN CR=1
4340 RETURN

```