Thermochimica Acta, 64 (1983) 195–205
Elsevier Science Publishers B.V., Amsterdam – Printed in The Netherlands

ITERATIVE NUMERICAL AND GRAPHICAL PROCEDURES FOR DETERMINING KINETIC PARAMETERS USING FOUR TG DATA PAIRS

J.E. HOUSE, Jr. and D.K. TCHENG

Department of Chemistry, Illinois State University, Normal, IL 61761 (U.S.A.)

(Received 4 November 1982)

ABSTRACT

Using four (α, T) data pairs, an equation has been obtained from which E/R is eliminated. One side of the equation is a function of α_1, α_2, T_1, T_2, and n, and the other is a function of α_3, α_4, T_3, T_4, and n. The two functions have a singular point at the correct value of n. An iterative numerical method with step refinement has been developed for determining n using a programmable calculator. A graphical method has been developed that employs a segment expansion procedure. In this procedure, implemented using a pocket computer, the region near the correct n is successively expanded until the point of intersection is determined to the desired accuracy.

INTRODUCTION

Numerical methods for identifying the reaction order, n, and the activation energy, E, for reactions following the rate law

$$\frac{\mathrm{d}\alpha}{\mathrm{d}T}=\frac{A}{\beta}(1-\alpha)^n\,\mathrm{e}^{-E/RT} \tag{1}$$

are efficiently carried out using microcomputers and programmable calculators [1–9]. Most conveniently, the two-point integrated form

$$\ln\left[\frac{1-(1-\alpha_i)^{1-n}}{1-(1-\alpha_{i+1})^{1-n}}\left(\frac{T_{i+1}}{T_i}\right)^2\right]=\frac{E}{R}\left(\frac{1}{T_{i+1}}-\frac{1}{T_i}\right) \tag{2}$$

is used with a series of (α, T) data [2,8]. Recent iterative methods involve different approaches to approximating the temperature integral [2–6] or methods of iterating to determine the value of n that meets some conditional requirement. First, using the left-hand side of eqn. (2) as f(α, T, n) and the right-hand side as f(T), the requirement can be used that the intercept must be zero for the correct value of n when f(α, T, n) is plotted against f(T) using various values for n [2]. Second, an approximately correct value of E/R is established based on an assumed value of n using small values of α.

0040-6031/83/$03.00

Then, using large values of α, the value of n is iterated to find the value that yields the correct value of E/R [9]. We describe here two new methods of finding n and E using programmable calculators and pocket computers.

METHODS

General procedure

We will assume that four values α_1, α_2, α_3, and α_4 are known at four temperatures T_1, T_2, T_3, and T_4, respectively. Further, we will assume that $\alpha_1 < \alpha_2 < \alpha_3 < \alpha_4$. The present method makes use of the two-point equation, eqn. (2), which can be written as

$$\frac{\ln\left[\dfrac{1-(1-\alpha_1)^{1-n}}{1-(1-\alpha_2)^{1-n}}\left(\dfrac{T_2}{T_1}\right)^2\right]}{\left(\dfrac{1}{T_2}-\dfrac{1}{T_1}\right)}=\frac{E}{R} \tag{3}$$

By considering an analogous equation for the points (α_3, T_3) and (α_4, T_4), elimination of E/R yields

$$\frac{\ln\left[\dfrac{1-(1-\alpha_1)^{1-n}}{1-(1-\alpha_2)^{1-n}}\left(\dfrac{T_2}{T_1}\right)^2\right]}{\left(\dfrac{1}{T_2}-\dfrac{1}{T_1}\right)}=\frac{\ln\left[\dfrac{1-(1-\alpha_3)^{1-n}}{1-(1-\alpha_4)^{1-n}}\left(\dfrac{T_4}{T_3}\right)^2\right]}{\left(\dfrac{1}{T_4}-\dfrac{1}{T_3}\right)} \tag{4}$$

Since both sides of eqn. (4) are equal to E/R, the two sides will be equal to each other only when n has the correct value. It is a characteristic of the E/R values that they are too small for values of n that are less than the correct value [1,9]. Further, the E/R values are smaller for larger values of α for a given value of n. Thus, it is possible to start with $n = 0$ and find that the left-hand side of eqn. (4) is greater than the right-hand side (unless, of course, $n = 0$ is the correct value). This will always be true if the condition $\alpha_1 < \alpha_2 < \alpha_3 < \alpha_4$ is met. The basic problem is to determine the value of n for which eqn. (4) is correct. We have developed an iterative numerical approach and a graphical method to determine the value of n meeting this condition. Both methods are rapid and are easily adapted to machines other than those described here.

Iterative numerical procedure

The computation makes use of four (α, T) data pairs and computes the left-hand side (L) and the right-hand side (R) of eqn. (4). A comparison of L and R is made to determine if L > R, which indicates that n is less than the correct value. If it is, then the value of n is incremented by Δn and the process is repeated until L < R. At that point, n is decremented by Δn and the step refinement of Δn takes place ($\Delta n \rightarrow \Delta n/2$). In that way, n is reduced to its value in the previous cycle, but the next increment is by $\Delta n/2$.

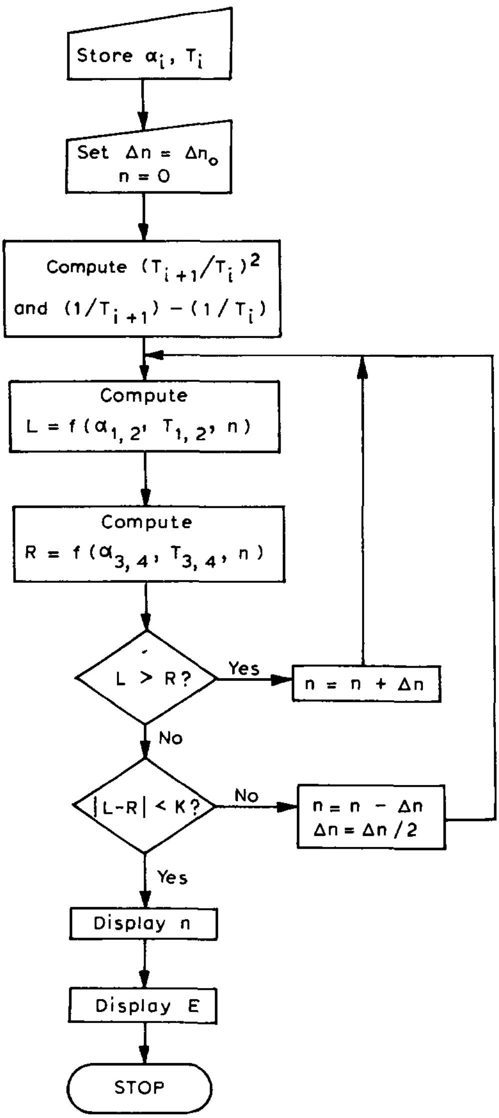

Fig. 1. Flow chart for the iterative numerical method.

When a trial value of n is too large, indicated by $L < R$, it is reduced first and then the refinement of Δn occurs so that n is approached from below in increments of decreasing size. After n is determined to the desired accuracy, the value of E can be calculated from the value of L or R since they are essentially equal.

In carrying out the computation, the initial value for Δn must be specified. Also, some criterion for the $L = R$ condition must be set to indicate that L is sufficiently close to R. While it is possible to integerize L and R are iterate until $L = R$, this is hardly necessary or practical. It is necessary only to find a value of n such that L is approximately equal to R. In this case, the condition $|L - R| < K$ has been used. As will be shown later, a value of $K = 5$ is perfectly adequate to assure that n is sufficiently accurate. Since a value of $E = 100$ kJ mole^{-1} corresponds to a value of $E/R = 12027$, this conditional test is sufficiently precise to assure an accurate result, and it is quickly met by the iteration scheme.

The computation was carried out using a Hewlett-Packard HP-34C programmable calculator. A flow chart for the computation is shown in Fig. 1 and a listing of the program appears in Appendix 1. From the flow chart, it is possible to adapt the algorithm to other microcomputers and programmable calculators.

Graphical procedure

From the foregoing discussion and previously published E/R data [1,9], it is readily apparent that if the left-hand side of eqn. (4) is computed using two (α, T) data pairs and various n values, a curve is generated (L vs. n). Depending on how small the α values are, the line will be reasonably straight and of small slope. However, if the right-hand side of eqn. (4) is used where α_3 and α_4 are somewhat larger than α_1 and α_2, the relationship of R vs. n will deviate from linearity and will always have a slope greater than that of L vs. n. Consequently, the two curves will intersect ($L = R$) at some "correct" value of n [1].

In the program, the variable name Y1 is used to denote the left-hand side of eqn. (4), A is used in place of α, and the variable name X is used for n. Thus, using the variable names as they are encountered in the program, the left-hand side of eqn. (4) is written as

$$\mathrm{Y1} = \frac{\ln\left[\dfrac{1-(1-A1)^{1-X}}{1-(1-A2)^{1-X}}\left(\dfrac{T2}{T1}\right)^2\right]}{\left(\dfrac{1}{T2}-\dfrac{1}{T1}\right)} \tag{5}$$

Similarly, Y2 is written as a function of $A3$, $A4$, $T3$, $T4$, and X. Thus, the graphical method consists of finding the point where variables Y1 and Y2 intersect. The variables Y1, Y2, and *X* are used because the method can be applied to any two equations by changing program lines 190 and 240 to compute the desired functions and by making an appropriate change in the range and domain.

A graphical solution is carried out using a computer program written in BASIC for the Radio Shack TRS-80 Model PC-2 pocket computer. The program uses features of both the computer and the plotter to solve

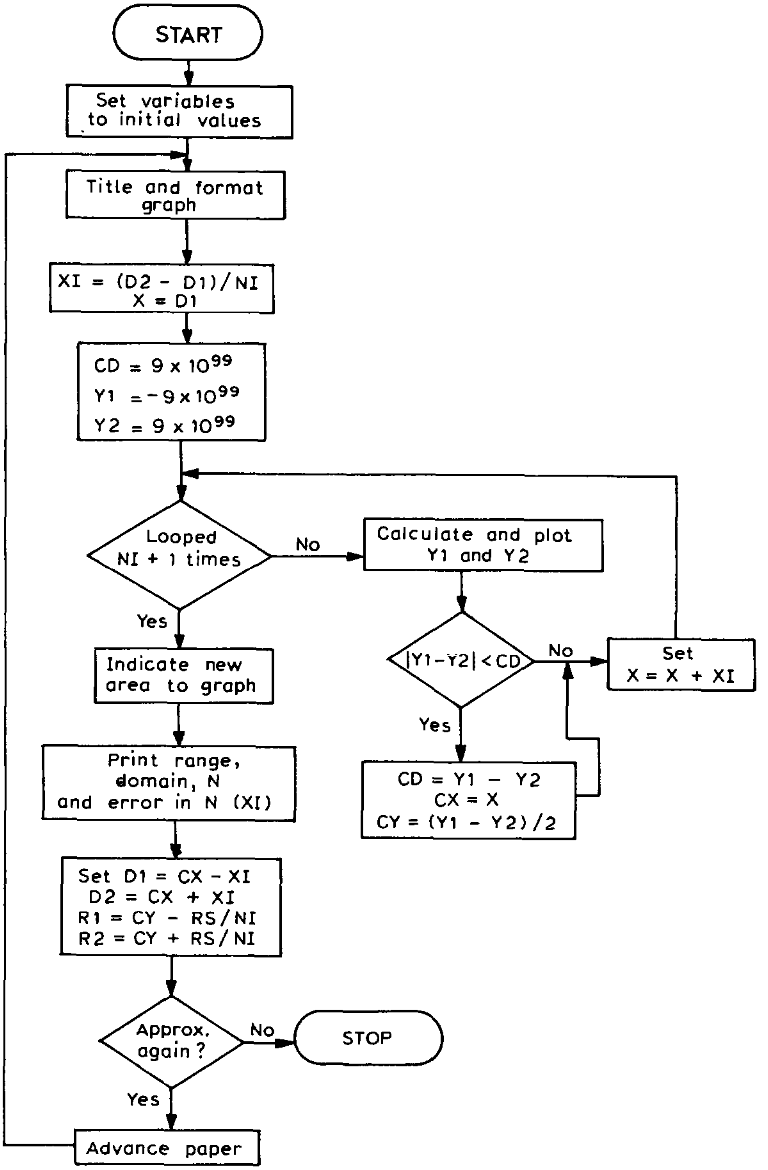

Fig. 2. Flow chart for the graphical method.

graphically for the value of X (actually the reaction order n in this case) where Y1 and Y2 are equal. Given two equations of the form

$$Y1 = f_1(X)$$

$$Y2 = f_2(X)$$

that intersect somewhere within a domain of X and a range of Y, the program will graph both functions at 10 intervals over a given domain. The domain of 0–3 is used here for n. The computer remembers the X value where the two curves were closest. Finally the program calculates a new, smaller domain and range based on the interval containing the point of intersection and produces subsequently a new full scale graph of that small region. The process is repeated until the point of intersection is located as accurately as is desired. In most cases, it appears that three successive expansions provide an accurate value for X. Figure 2 shows a flow chart of this method and a program listing is shown in Appendix 2.

TESTING THE METHODS

In order to test the two methods of determining n, the (α, T) data obtained by numerical solutions of rate equations for various values of n have been used [10]. The actual data used are shown in Table 1. However, selecting other data pairs from those published did not materially change the results. Very small values of α (< 0.03) should be avoided because such data have larger relative errors owing to the initial boundary conditions used in the Runge–Kutta method [10].

The program for the numerical method is designed to display only the final n that meets the $|L - R| < K$ condition and E/R. Intermediate L and R values can be printed or displayed to show the convergence of L and R values as n is iterated. In the present case, $K = 5$ was used although this

TABLE 1

The (α, T) data used to test the computational methods [a]

n	α_1	T_1	α_2	T_2	α_3	T_3	α_4	T_4
0	0.03263	400	0.07142	410	0.30693	430	0.60670	440
1/3	0.03246	400	0.07056	410	0.53996	440	0.89313	450
1/2	0.03237	400	0.07015	410	0.51426	440	0.82452	450
2/3	0.06974	410	0.14317	420	0.77010	450	0.97868	460
1	0.06893	410	0.13981	420	0.88562	460	0.98060	470
4/3	0.06818	410	0.13666	420	0.80441	460	0.91931	470
5/3	0.06740	410	0.13369	420	0.73851	460	0.85531	470

[a] $E = 100$ kJ mole^{-1} and $A/\beta = 3 \times 10^{10}$ min^{-1} used in the Runge–Kutta solutions [10].

TABLE 2

Computed results using the present methods [a]

Actual n	Iterative method		Graphical method	
	n	E (kJ mole^{-1})	n	E/R range
0	0.005	100.115	0.000	11629–12429
1/3	0.337	100.050	0.336	11752–12552
1/2	0.505	100.078	0.504	11534–12334
2/3	0.668	99.913	0.672	11546–12346
1	1.001	99.926	0.996	11807–12607
4/3	1.335	99.950	1.332	11524–12324
5/3	1.670	99.959	1.668	11668–12468
2	2.003	99.932	2.004	11516–12316

[a] $E = 100$ kJ mole^{-1} and $A/\beta = 3 \times 10^{10}$ min^{-1} used in the Runge-Kutta solutions [10].

parameter could be assigned other values if desired. To perform the computation, the α_i values are all entered (registers R_0 through R_3) followed by the T_i values (in registers R_4 through R_7). Next, the trial n is set equal to zero ($R_{.2}$) and Δn is given its initial value (in $R_{.3}$). These parameters are later shifted to R_6 and R_7 after the T_i are used to compute the $(T_{i+1}/T_i)^2$ and $((1/T_{i+1}) - (1/T_i))$. This is done so that changes in n and Δn can be made using register arithmetic which is performed only on registers R_0 to R_9 on the HP-34C machine. A value of $\Delta n = 0.2501$ was used in the present work so that n cannot be exactly $1.00 \cdots$.

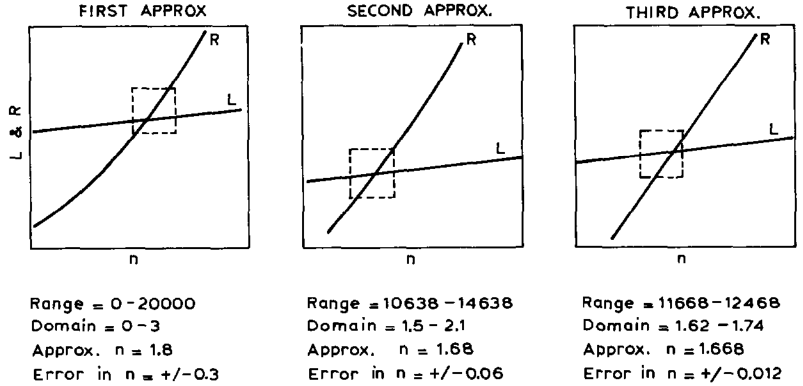

Fig. 3. Output from the graphical method applied to the case where $n = 5/3$. The entire graph shown in the Second Approximation represents an expansion of the small square (dashed lines) surrounding the intersection point in the First Approximation graph. The Third Approximation graph represents an expansion of the small square in the Second Approximation, etc.

Table 2 shows the results obtained using the four-point iterative method. It is readily apparent that the method accurately determines n and E. The computation time is short when $K = 5$, and the results are virtually identical regardless of the starting value for Δn.

A sample of the output from the graphical method is shown in Fig. 3 using the (α, T) data for which $n = 1.66 \cdots$. The approximate value for n is determined as 1.8 ± 0.3, 1.68 ± 0.06, and 1.668 ± 0.012, respectively, in the first three cycles. Similar results were obtained using the other (α, T) data for other values of n [10]. Table 2 shows the values of n obtained after three expansion cycles for cases where n varies from 0 to 2. As presented, the program does not compute E directly, but rather computes the range of E/R represented in the interval considered. Consequently, the E/R value is very nearly the mean value for the range shown in the interval.

The results shown in Table 2 indicate that the methods described here accurately determine n and E. These methods are rapid and are easily adapted to other calculators and computers. However, as in other methods, the accuracy of experimental (α, T) data will be of great importance in determining the computed values for n and E [9]. Considering the variation from sample to sample [11], such methods based on limited data may achieve as much accuracy as can be justified by numerical procedures unless the data from a great many runs are considered.

REFERENCES

1 L. Reich and S.S. Stivala, Thermochim. Acta, 24 (1978) 9.
2 L. Reich and S.S. Stivala, Thermochim. Acta, 36 (1980) 103.
3 L. Reich and S.S. Stivala, Thermochim. Acta, 41 (1980) 391.
4 L. Reich and S.S. Stivala, Thermochim. Acta, 52 (1982) 337.
5 L. Reich and S.S. Stivala, Thermochim. Acta, 53 (1982) 121.
6 L. Reich and S.S. Stivala, Thermochim. Acta, 55 (1982) 385.
7 J.E. House, Jr., Thermochim. Acta, 57 (1982) 47.
8 J.E. House, Jr., Comput. Chem., 6 (1982) 27.
9 J.E. House, Jr., and J.D. House, Thermochim. Acta, 61 (1983) 277.
10 J.E. House, Jr., Thermochim. Acta, 55 (1982) 241.
11 J.E. House, Jr., Thermochim. Acta, 47 (1981) 379.

APPENDIX 1

HP-34C Program listing for the four-point iterative method

000–		
001–	25 13 11	h LBL A
002–	24 5	RCL 5
003–	24 4	RCL 4
004–	71	÷
005–	15 3	g X^2
006–	23 8	STO 8
007–	24 7	RCL 7
008–	24 6	RCL 6
009–	71	÷
010–	15 3	g X^2
011–	23 9	STO 9
012–	24 5	RCL 5
013–	25 2	h 1/X
014–	24 4	RCl 4
015–	25 2	h 1/X
016–	41	–
017–	23 .0	STO .0
018–	24 7	RCL 7
019–	25 2	h 1/X
020–	24 6	RCL 6
021–	25 2	h 1/X
022–	41	–
023–	23 .1	STO .1
024–	24 .3	RCL .3
025–	23 7	STO 7
026–	24 .2	RCL .2
027–	23 6	STO 6
028–	25 13 12	h LBL B
029–	1	1
030–	24 0	RCL 0
031–	41	–
032–	1	1
033–	24 6	RCL 6
034–	41	–
035–	25 3	h Y^X
036–	32	CHS
037–	1	1
038–	51	+
039–	1	1
040–	24 1	RCL 1
041–	41	–
042–	1	1
043–	24 6	RCL 6
044–	41	–
045–	25 3	h Y^X
046–	32	CHS
047–	1	1
048–	51	+
049–	71	÷
050–	24 8	RCL 8
051–	61	x
052–	14 1	f LN
053–	24 .0	RCL .0
054–	71	÷
055–	23 4	STO 4
056–	1	1
057–	24 2	RCL 2
058–	41	–
059–	1	1
060–	24 6	RCL 6
061–	41	–
062–	25 3	h Y^X
063–	32	CHS
064–	1	1
065–	51	+
066–	1	1
067–	24 3	RCL 3
068–	41	–
069–	1	1
070–	24 6	RCL 6
071–	41	–
072–	25 3	h Y^X
073–	32	CHS
074–	1	1
075–	51	+
076–	71	÷
077–	24 9	RCL 9
078–	61	x
079–	14 .1	f LN
080–	24 1	RCL .1
081–	71	÷
082–	23 5	STO 5
083–	24 4	RCL 4
084–	24 5	RCL 5
085	14 51	f X > Y
086–	22 0	GTO 0
087–	22 1	GTO 1
088–	25 13 0	h LBL 0
089–	24 7	RCL 7
090–	23 41 6	STO–6
091–	2	2
092–	23 71 7	STO÷7
093–	22 12	GTO B
094–	25 13 1	h LBL 1
095–	5	5
096–	24 4	RCL 4
097–	24 5	RCL 5
098–	41	–
099–	25 34	h ABS
100–	14 41	f X ⩽ Y
101–	22 2	GTO 2
102–	24 7	RCL 7
103–	23 51 6	STO+6
104–	22 12	GTO B
105–	25 13 2	h LBL 2
106–	24 6	RCL 6
107–	74	R/S
108–	24 4	RCL 4
109–	74	R/S

APPENDIX 2

Program listing for the graphical method

```
  1:  REM
      GRAPHICAL
      METHOD
  2:  T1 = 410: T2 = 420:
      T3 = 460: T4 = 470
  3:  A1 = .06740: A2 = .
      13369: A3 = .7385
      1: A4 = .85531
  4:  R1 = 0: R2 = 20000:
      D1 = 0: D2 = 3: NI = 1
      0: K = 200
100:  NS = NS + 1: GT$ = "#
      " + STR$ NS + " AP
      PROX. OF N":
      GRAPH
110:  GLCURSOR (16, 0
      ): SORGN
120:  CSIZE 2: ROTATE
      0: GLCURSOR (10
      , 204): LPRINT G
      T$
130:  CSIZE 1: ROTATE
      3: GLCURSOR (-2
      , 58): LPRINT "L
      &R in Degrees"
140:  ROTATE 0:
      GLCURSOR (97, -
      8): LPRINT "N"
150:  LINE (0, 200)-(
      200, 0), 0,, B
160:  XI = (D2-D1)/(NI
      ): X = D1
170:  CD = 9E99: Y1 = -9E
      99: Y2 = 9E99
180:  FOR L = 1TO NI+1
190:  Y = LN ((1-(1-A1
      )∧(1-X))/(1-(1
      -A2)∧(1-X))*(T
      2/T1)∧2)/(1/T2
      -1/T1)
200:  IF Y > R2OR Y < R1
      OR Y1 > R2OR Y1 <
      R1 THEN 230
210:  GLCURSOR ((X1-
      D1)/(D2-D1)*K,
      (Y1-R1)/(R2-R1
      )*K)
220:  LINE -((X-D1)/
      (D2-D1)*K, (Y-R
      1)/(R2-R1)*K),
      0
230:  X1 = X: Y1 = Y
240:  Y = LN ((1-(1-A3
      )∧(1-X))/(1-(1
      -A4)∧(1-X))*(T
      4/T3)∧2)/(1/T4
      -1/T3)
250:  IF Y > R2OR Y < R1
      OR Y2 > R2OR Y2 <
      R1 THEN 280
260:  GLCURSOR ((X2-
      D1)/(D2-D1)*K,
      (Y2-R1)/(R2-R1
      )*K)
270:  LINE -((X-D1)/
      (D2-D1)*K, (Y-R
      1)/(R2-R1)*K),
      0
280:  X2 = X: Y2 = Y
290:  IF ABS (Y1-Y2)
      < CDTHEN LET CD
      = ABS (Y1-Y2): C
      X = X: CY = (Y1+Y2)
      /2
300:  X = X + XI
310:  NEXT L
320:  RS = R2-R1: DS = D2
      -D1
330:  GLCURSOR ((CX-
      XI-D1)/(D2-D1)
      *K, (CY-RS/NI-R
      1)/(R2-R1)*K)
340:  LINE -((CX+XI-
      D1)/(D2-D1)*K,
      (CY+RS/NI-R1)/
      (R2-R1)*K), 3,,
      B
350:  GLCURSOR (-16,
      -20): TEXT:
      CSIZE 1: LPRINT
      "  RANGE = ";
      INT R1; "J TO"
      ;INT R2; "J"
360:  LPRINT"  DO
      MAIN = "; D1;" TO
      "; D2
```

APPENDIX 2 (continued)

```
370:  LPRINT"   AP
      PROX. N = "; CX:
      LPRINT ER
      ROR IN N = + - "
      ; XI
380:  D1 = CX - XI: D2 = CX
      + XI
390:  R1 = CY - RS/NI: R2
      = CY + RS/NI
400:  INPUT "ANOTHER
      APPROX.?"; X$
410:  IF LEFT$ (X$, 1
      ) = "Y" THEN
      GRAPH:
      GLCURSOR (0, -3
      00): GOTO 100
9999: END
```