Note

BASIC LANGUAGE PROGRAMS FOR AUTOMATIC PROCESSING NON-ISOTHERMAL KINETIC DATA

EUGENIA EFTIMIE and E. SEGAL

Chair of Physical Chemistry and Electrochemical Technology, Polytechnic Institute of Bucharest, Bd. Republicii 13, Bucharest (Romania)

(Received 13 June 1986)

In a previous note [1] we reported a BASIC language program for working the non-isothermal kinetic data by use of the Coats–Redfern method [2]. This communication deals with two other BASIC programs for the modified Coats–Redfern method [2] and an iterative method [4] recently worked out to evaluate the non-isothermal kinetic parameters.

The modified Coats–Redfern method uses the equation:

$$\log\int_0^\alpha \frac{\mathrm{d}\alpha}{\mathrm{f}(\alpha)T^2} = \log\frac{AR}{\beta E} - \frac{E}{2.303R}\frac{1}{T} \tag{1}$$

where all the parameters have their usual meanings, whose variant for $\mathrm{f}(\alpha) = (1-\alpha)^n$ is:

$$\log\int_0^\alpha \frac{\mathrm{d}\alpha}{(1-\alpha)^n T^2} = \log\frac{AR}{\beta E} - \frac{E}{2.303R}\frac{1}{T} \tag{2}$$

For the correct value of n the plot of $\log\int_0^\alpha \frac{\mathrm{d}\alpha}{(1-\alpha)^n T^2}$ values against $1/T$ values, should give a straight line whose slope and intercept allow the calculation of the activation energy, E, and the pre-exponential factor A.

To calculate the integral from eqn. (2) the trapezoidal method for unequal distances is used [5]. By introducing the notation:

$$\int_0^\alpha \frac{\mathrm{d}\alpha}{(1-\alpha)^n T^2} = I_\alpha \tag{3}$$

with $I_{\alpha_0} = 0$, one obtains:

$$I_{\alpha_{i+1}} = I_{\alpha_i} + (\alpha_{i+1} - \alpha_i)\left[\frac{\mathrm{g}(\alpha_{i+1}) + \mathrm{g}(\alpha_i)}{2}\right] \tag{4}$$

where

$$\mathrm{g}(\alpha) = \left[(1-\alpha)^n T^2\right]^{-1} \tag{5}$$

0040-6031/87/$03.50

The program for the modified Coats–Redfern method allows the selection of the value of the "reaction order", n, corresponding to the best linearity $\left[\log\frac{\mathrm{d}\alpha}{(1-\alpha)^n T^2}\ \frac{1}{T}\right]$.The program calculates the slope and the intercept of the straight line as well as the value of the correlation coefficient corresponding to the linear regression.

Input data: N—number of experimental points; W_i—sample weight decrease at temperature T_i; W_g—total sample weight decrease; T_i—temperature (°C); U—heating rate.

The maximum number of experimental points which can be used is 20.

In the vectors $X(I)$ and $Y(I)$ the values $T(I)^{-1}$ and, respectively, $\log\int_0^{\alpha}\frac{\mathrm{d}\alpha}{(1-\alpha)^n T^2}$ are calculated. The two vectors are indexed for a given reaction order in each experimental point. To evaluate the elements of $Y(I)$

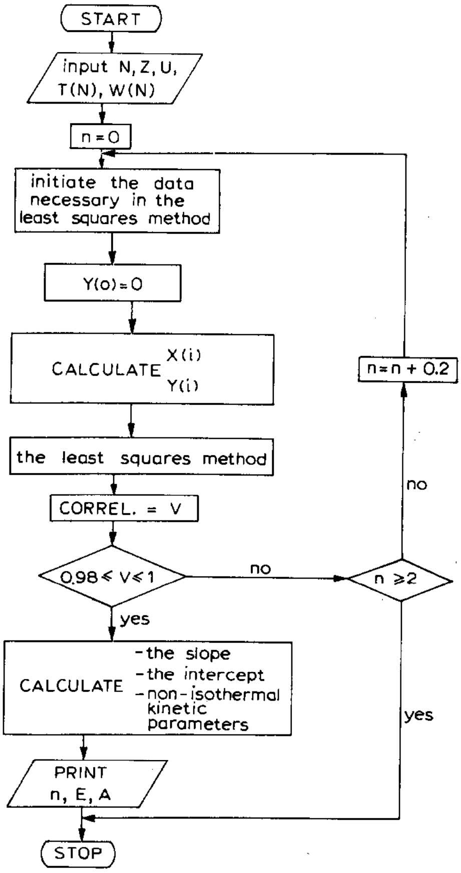
START
input N, Z, U, T(N), W(N)
n = 0
initiate the data necessary in the least squares method
Y(o) = 0
CALCULATE X(i) Y(i)
the least squares method
CORREL. = V
0.98 ≤ V ≤ 1
no
yes
n ≥ 2
no
yes
n = n + 0.2
CALCULATE -the slope -the intercept -non-isothermal kinetic parameters
PRINT n, E, A
STOP

Fig. 1. Flow diagram of the modified Coats–Redfern method.

a new vector $L(I)$ in which one calculates the integral I_a, taking into account relationships (3), (4) and (5) is introduced. The vector $L(I)$ is also indexed for each experimental point.

In the initialized quantities A, B, C, D and E which are the same as in the previous program for the Coats–Redfern method [1], the data necessary for the least squares method are calculated. In P and Q the slope and the intercept of the straight line are calculated; K and M calculate the standard deviation for $X(I)$ and $Y(I)$, respectively, and V calculates the correlation coefficient.

To calculate the kinetic parameters E, and A the following formulae are used:

$$E = 4.573P \tag{6}$$

$$A = (10^{Q} \cdot UE)/(60R) \tag{7}$$

The calculation of E (cal mol^{-1}) and A (s) according to eqns. (6) and (7) are performed only after the fulfilment of the condition:

$$|V| \in (0.98 - 1) \tag{8}$$

If the condition (8) is not fulfilled for a given reaction order, denoted in the program by H, the reaction order is changed with a given step (in our case 0.2) imposed by the program (see instruction 50) in the interval of values 0–2 (see instruction 49).

The program is initiated using the instruction RUN e_1 where e_1 is the starting label of the program.

The logical diagram of the program given in Fig. 1 is followed by the instructions of the program.

Instructions of the program for the modified Coats–Redfern method

```
 1. IMPUT "NR. OBS"; N
 2. DIM T(N), W(N), X(N), Y(N)
 3. WAIT 48
 4. PAUSE "ENTER"; N; "PAIRS"
 5. FOR I = 0 TO N
 6. PRINT "PAIR"; I
 7. IMPUT "T?"; T(I)
 8. IMPUT "W?"; W(I)
 9. NEXT I
10. WAIT 128
11. INPUT "HEATING RATE?"; U
12. INPUT "TOTAL DECREASE?"; Z
13. H = 0
14. PRINT "ORDER"; H
15. A = 0, B = 0, C = 0, D = 0, E = 0
16. L(0) = 0, R = N - 1
17.    FOR I = 0 TO R
```

```
18.  K = I + 1
19.  L(K) = L(I) + ((W(I) − W(K))/(2 * Z) * (Z ∧ H) *
     (1/(W(K) ∧ H * (T(K) + 273) ∧ 2) + 1/(W(I) ∧ H * (T(I) + 273) ∧
     2))
20.  X(K) = 1/T(K) : Y(K) = LOG (L(K))
21.  A = A + X(K)
22.  B = B + X(K) ∧ 2
23.  C = C + Y(K)
24.  D = D + Y(K) ∧ 2
25.  E = E + (X(K) + Y(K))
26.  NEXT I
27.  WAIT
28.  F = A/N
29.  G = C/N
30.  J = B − ((A ∧ 2)/N)
31.  K = SQR (J/(N − 1))
32.  L = D − ((C ∧ 2)/N)
33.  M = SQR (L/(N − 1))
34.  O = E − ((A * C)/N)
35.  R = O/SQR (J * L)
36.  PRINT "ORDER = "; H
37.  PRINT "CORRELATION = "; R
38.  IF (ABS(R) < .98) GOTO 49
39.  P = O/J
40.  Q = G − (P * F)
41.  WAIT 128
42.  PRINT "EQ.IS"
43.  WAIT
44.  PRINT "Y = "; Q; "+"; P; "X"
45.  E = 4.575 * P
46.  A = (10 ∧ Q) * E * U/120
47.  PRINT "ENERGY = "; E
48.  PRINT "FACTOR = "; A
49.  IF (H = 2) GOTO 52
50.  H = H + 0.2
51.  GOTO 14
52.  PRINT = PRINT
53.  END
```

As far as the program for the iterative method is concerned, for a given iteration j, one has to consider a system of three equations with three unknowns (A, E and n). The equations are of the general form [4]:

$$\log A^j + n^j \log\left(1 - \alpha_{ik}^j\right) - \frac{E^j}{2.303\, RT_{ik}^j} = \log\frac{\alpha_k - \alpha_i}{t_k - t_i} \tag{9}$$

with α_{ik}^{j} and T_{ik}^{j} given by:

$$\frac{\alpha_k - \alpha_i}{\left(1 - \alpha_{ik}^{j+1}\right)^{n^j}} = \frac{(1 - \alpha_i)^{1-n^j}(1 - \alpha_k)^{1-n^j}}{1 - n^j} \tag{10}$$

$$T_k - T_i \, \mathrm{e} - \frac{E^j}{RT_{ik}^{j+1}} = \frac{T_k - T_i}{6}\left(\mathrm{e}^{-E/RT_i} + 4\,\mathrm{e}^{-E/RT_{jk}} + \mathrm{e}^{-E/RT_k}\right) \tag{11}$$

For the zero (initial) iteration:

$$\alpha_{ik}^{0} = \frac{\alpha_i + \alpha_k}{2} \tag{12}$$

$$T_{ik}^{0} = \frac{T_i + T_k}{2} \tag{13}$$

Using eqns. (12) and (13) the values A^0, E^0 and n^0 can be calculated.

For four sets of conversion degree and temperature values which fulfil the conditions:

$$\Delta\alpha_{ik} = \alpha_k - \alpha_i \in (0.3 - 0.4) \tag{14}$$

$$\Delta T_{ik} = T_k - T_i \in (4 - 5\ \mathrm{K}, 15\text{–}20\ \mathrm{K}) \tag{15}$$

the iterations continue until the fulfilment of the conditions [4]:

$$\left.\begin{aligned} &|n^{j+1} - n^j| \le n \\ &|E^{j+1} - E^j| \le E \\ &|\log A^{j+1} - \log A^j| \le \log A \end{aligned}\right\} \tag{16}$$

Thus for the values α_{ik}^{j} and T_{ik}^{j} the following system of three equations can be written:

$$\left.\begin{aligned} &\log A^j + n^j \log\left(1 - \alpha_{12}^{j}\right) - \frac{E^j}{2.303\ RT_{12}^{j}} = \log\frac{\alpha_2 - \alpha_1}{t_2 - t_1} \\ &\log A^j + n^j \log\left(1 - \alpha_{23}^{j}\right) - \frac{E^j}{2.303RT_{23}^{j}} = \log\frac{\alpha_3 - \alpha_2}{t_3 - t_2} \\ &\log A^j + n^j \log\left(1 - \alpha_{34}^{j}\right) - \frac{E^j}{2.303\ RT_{34}^{j}} = \log\frac{\alpha_4 - \alpha_3}{t_4 - t_3} \end{aligned}\right\} \tag{17}$$

which can be solved using determinants. If values of $\log A^j$, E^j and n^j fulfil conditions (16), the iterations stop and the values of the kinetic parameters are displayed.

Input data: $T(I)$—temperature in point $I(K)$; $C(I)$—conversion in point I; $S(I)$—time necessary to reach conversion $C(I)$; ($I = 1, 2, 3, 4$)

The input data are recorded for a given heating rate. Variable J contains the iterations counter. The vectors $Q(j)$, $T(j)$, $R(j)$ calculate the values of

α_{ik}^{j} and the vectors $F(j)$, $G(j)$, $H(j)$ calculate the values of T_{ik}^{j}. The free term values in eqns. (17) which is the same for the same equation in every new system, is calculated in $B(I)$.

In A, E, U, L, M, N the terms of the determinants are calculated and in D, O, W, V, the following determinants are calculated:

$$D = \begin{vmatrix} 1 & \log(1-\alpha_{12}^{j}) & -1/(2.303\,RT_{12}^{j}) \\ 1 & \log(1-\alpha_{23}^{j}) & -1/(2.303\,RT_{23}^{j}) \\ 1 & \log(1-\alpha_{34}^{j}) & -1/(2.303\,RT_{34}^{j}) \end{vmatrix} \tag{18}$$

$$0 = \begin{vmatrix} \log\dfrac{\alpha_2-\alpha_1}{t_2-t_1} & \log(1-\alpha_{12}^{j}) & -1/(2.303\,RT_{12}^{j}) \\ \log\dfrac{\alpha_3-\alpha_2}{t_3-t_2} & \log(1-\alpha_{23}^{j}) & -1/(2.303\,RT_{23}^{j}) \\ \log\dfrac{\alpha_4-\alpha_3}{t_4-t_3} & \log(1-\alpha_{34}^{j}) & -1/(2.303\,RT_{34}^{j}) \end{vmatrix} \tag{19}$$

$$W = \begin{vmatrix} 1 & \log\dfrac{\alpha_2-\alpha_1}{t_2-t_1} & -1/(2.303\,RT_{12}^{j}) \\ 1 & \log\dfrac{\alpha_3-\alpha_2}{t_3-t_1} & -1/(2.303\,RT_{23}^{j}) \\ 1 & \log\dfrac{\alpha_4-\alpha_3}{t_4-t_3} & -1/(2.303\,RT_{34}^{j}) \end{vmatrix} \tag{20}$$

$$V = \begin{vmatrix} 1 & \log(1-\alpha_{12}^{j}) & \log\dfrac{\alpha_2-\alpha_1}{t_2-t_1} \\ 1 & \log(1-\alpha_{23}^{j}) & \log\dfrac{\alpha_3-\alpha_2}{t_3-t_2} \\ 1 & \log(1-\alpha_{34}^{j}) & \log\dfrac{\alpha_4-\alpha_3}{t_4-t_3} \end{vmatrix} \tag{21}$$

In the vectors $X(j)$, $Y(j)$, $Z(j)$ the values of log $A^{(j)}$, $n^{(j)}$, $E^{(j)}$ for the iteration member j are calculated.

The program is initiated using the instruction RUN e_2, where e_2 is the starting label of the program.

The logical diagram of the program is given in Fig. 2.

Instructions of the program for the iterative method:

```
1. PRINT "ITERATIVE METHOD"
2. DIM T(4), C(4), S(4), B(3), Q(24), P(24), R(24)
3. DIM F(24), G(24), H(24), X(24), Y(24), Z(24)
4.    FOR I = 1 TO 4
5.    INPUT "T?"; T(I)
```

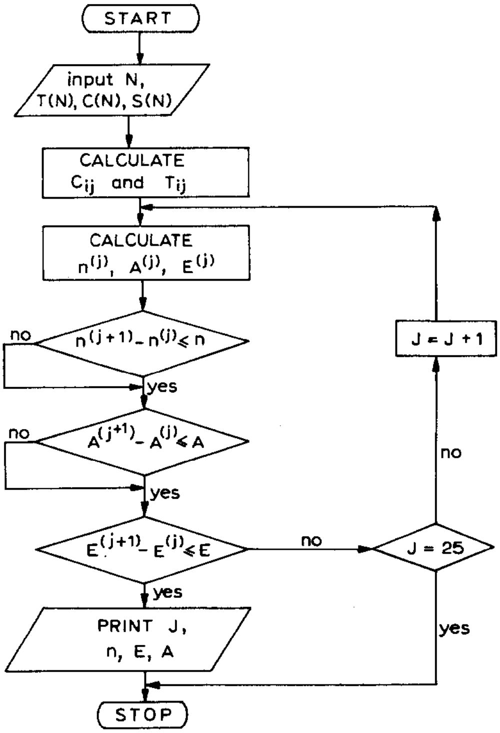
START
input N, T(N), C(N), S(N)
CALCULATE C_{ij} and T_{ij}
CALCULATE $n^{(j)}$, $A^{(j)}$, $E^{(j)}$
no
$n^{(j+1)} - n^{(j)} \leq n$
yes
no
$A^{(j+1)} - A^{(j)} \leq A$
yes
$E^{(j+1)} - E^{(j)} \leq E$
no
J = 25
no
J = J + 1
yes
yes
PRINT J, n, E, A
STOP

Fig. 2. Flow diagram of the iterative method.

```
 6.    INPUT "C?"; C(I)
 7.    INPUT "S?"; S(I)
 8.    NEXT I
 9.  J = 0
10.  Q(j) = (C(1) + C(2))/2
11.  P(j) = (C(2) + C(3))/2
12.  R(j) = (C(3) + C(4))/2
13.  F(j) = ((T(1) + T(2))/2) ∧ − 1
14.  G(j) = ((T(2) + T(3))/2) ∧ − 1
15.  H(j) = ((T(3) + T(4))/2) ∧ − 1
16.    FOR I = 1 TO 3
17.    K = I + 1
18.    B(I) = LOG((C(K) − C(I))/(S(K) − S(I))
19.    NEXT I
20.    A = 1 − Q(j)
21.    E = 1 − P(j)
22.    U = 1 − R(j)
23.    L = F(j)/4.573
24.    M = G(j)/4.573
25.    N = H(j)/4.573
```

```
26.   D = E * N + U * L + A * M - L * E - U * M - A * N
27.   O = B(1) * E * N + B(2) * U * L + B(3) * A * M - B(3) * L * E -
      B(1) * U * M - B(2) * A * N
28. W = B(2) * N + B(3) * L + B(1) * M - B(2) * L - B(3) * M - B(1) * N
29. V = B(3) * E + B(2) * A + B(1) * U - B(3) * A - B(2) * U - B(1) * E
30.   X(j) = O/D
31.   Y(j) = W/D
32.   Z(j) = V/D
33.   IF(J ⩾ 2) GOTO 43
34.   J = J + 1
35.   K = J - 1
36. Q(j) = ((C(2) - C(1)) * (1 - Y(K)))/(((1 - C(1)) ∧ (Y(K) * ((1 - C(2))
    ∧ (Y(K)) * Y(J))
37.   P(j) = ((C(3) - C(2)) * (1 - Y(K)))/(((1 - C(2)) ∧ (Y(K)) * ((1 -
      C(3)) ∧ (Y(K)) * Y(J))
38. R(j) = ((C(4) - C(3)) * (1 - Y(K)))/(((1 - C(3)) ∧ (Y(K)) * ((1 - C(4))
    ∧ (Y(K)) * Y(J))
39. F(j) = (1.987/Z(J)) * LN(6/(EXP(-Z(K)/(1.987 * T(1))) +
    4 * EXP(-Z(K) * F(K)/1.987) + EXP(-Z(K)/(1.987 * T(2))))
40. G(j) = (1.987/Z(J)) * LN(6/EXP(-Z(K)/(1.987 * T(2))) +
    4 * EXP(-Z(K) * G(K)/1.987) + EXP(-Z(K)/(1.987 * T(3))))
41. H(j) = (1.987/Z(J)) * LN(6/EXP(-Z(K)/(1.987 * T(3))) +
    4 * EXP(-Z(K) * H(K)/1.987) + EXP(-Z(K)/(1.987 * T(4))))
42. GOTO 16
43. PRINT "log A = "; X(J): PRINT "ORDER = "; Y(J)
44. PRINT "ENERGY = "; Z(J)
45. K = J - 1
46. IF (ABS(X(J) - X(K)) < = LOGA) GOTO 47
47. IF (ABS(Y(J) - Y(K)) < = n) GOTO 48
48. IF (ABS(Z(J) - Z(K)) < = E) GOTO 52
49. IF (J = 7) GOTO 56
50. J = J + 1
51. GOTO 35
52. A = 10 ∧ X(J)
53. PRINT "A = "; A
54. PRINT "E = "; Z(j)
55. PRINT "N = "; Y(j)
56. WAIT
57. PRINT = PRINT
58. END
```

The two programs were run on a Sharp programmable pocket-computer, P.C.-1251. The results obtained concerning the non-isothermal kinetic parameters of the decomposition of some polynuclear coordination compounds with rare earth elements will be given in a following note.

REFERENCES

1 E. Eftimie and E. Segal, Thermochim. Acta, 105 (1986) 247.
2 A.W. Coats and J.P. Redfern, Nature (London), 201 (1964) 18.
3 E. Urbanovici and E. Segal, Thermochim. Acta, 80 (1984) 374.
4 E. Urbanovici and E. Segal, Thermochim. Acta, in press.
5 Kleine Enzyklopedie der Mathematik, Editura Tehnică, Bucharest, 1980, pp. 748–749 (in Romanian).