Journal of Thermal Analysis, Vol. 19 (1980) 333–345

KINETIC ANALYSIS OF THERMOGRAVIMETRIC DATA

XIV. THREE INTEGRAL METHODS AND THEIR COMPUTER PROGRAMS

J. ZSAKÓ and J. ZSAKÓ jr.

Department of Chemical Technology, "Babes-Bolyai" University, str. Arany János 11, 3400 Cluj-Napoca, Rumania

(Received January 29, 1980)

Computer programs are given in Fortran language for three integral methods of deriving kinetic parameters from TG curves. Method 1 is a computerized variant of Doyle's curve-fitting method and performs the calculation of the exponential integral $p(x)$ by means of author's empirical formula. Methods 2 and 3 are variants of the Coats–Redfern linearization method. Testing of the methods on both theoretical and experimental TG curves shows them to be almost equivalent as far as the results obtained are concerned, but Method 1 needs a ten-fold higher computer time.

In a solid-state reaction leading to the evolution of a gaseous product, the conversion can be difined as

$$\alpha = \frac{G_0 - G}{G_0 - G_1} \tag{1}$$

where G_0, G and G_1 stand for the initial, actual and final sample weights, respectively. The rate of the reaction depends upon the conversion, according to the relation

$$\frac{d\alpha}{dt} = kf(\alpha) \tag{2}$$

where the rate constant k is frequently presumed to obey the Arrhenius law

$$k = Ze^{-E/RT} \tag{3}$$

Z will be referred to as the pre-exponential factor, E stands for the apparent activation energy, R is the gas constant and T the absolute temperature.

Integral methods use the equation of the thermogravimetric curves (2) in its integrated form [1]:

$$g(\alpha) = \frac{ZE}{Rq} p(x) \tag{4}$$

where $g(\alpha)$ stands for the conversion integral

$$g(\alpha) = \int_0^{\alpha'} \frac{d\alpha}{f(\alpha)} \tag{5}$$

q for the heating rate dT/dt and $p(x)$ for the exponential integral

$$p(x) = \int_x^\infty \frac{e^{-u}}{u^2} du \quad \text{with} \quad u = E/RT \tag{6}$$

The conversion integral $g(\alpha)$ can easily be calculated if one presumes [2] the conversion function $f(\alpha)$ to be of the following form:

$$f(\alpha) = (1 - \alpha)^n \tag{7}$$

where n stands for the apparent reaction order.

For the calculation of the exponential integral, a rather good [3] empirical formula has been proposed in our earlier paper [4]:

$$p(x) = \frac{e^{-x}}{(x + 2)(x - d)} \quad \text{with} \quad d = \frac{16}{x^2 - 4x + 84} \tag{8}$$

In the present paper three methods are suggested with the aim of deriving the kinetic parameters n, E and Z from TG curves.

Method 1

The basic idea of the method is the constancy of the magnitude

$$B = \log \frac{ZE}{Rq} = \log g(\alpha) - \log p(x) \tag{9}$$

resulting from (4). The integrals $g(\alpha)$ and $p(x)$ have different values for each point of the TG curves, but their ratio is constant. In our trial-and-error method proposed earlier [5], tabulated values of $p(x)$ are used, and by the systematic variation of the presumed E value the minimization of the standard deviation

$$\delta = \sqrt{\frac{\Sigma (B_i - \bar{B})^2}{M}} \tag{10}$$

is performed for a given n value. B_i stands for the B value calculated from a single point of the TG curve by means of formula (9), $\bar{B}$ represents the arithmetical mean of the individual B_i values, and M stands for the number of experimental points used. In order to determine the most probable n value, the above minimization is performed for the following n values: 0, 1/3, 1/2, 2/3, 1 and 2. The minimum of the δ_{min} values indicates the best n value.

A computer program of this method has been proposed by Simon *et al.* [6].

In the present paper we propose a modification of this method in the following way:

1. To calculate the exponential integral by means of (8) instead of using the very large numerical tables.

2. To calculate $g(\alpha)$ by means of formulae

$$g(\alpha) = \frac{(1-\alpha)^{1-n} - 1}{n-1} \quad \text{if} \quad n \neq 1 \quad \text{and}$$

$$g(\alpha) = -\ln(1-\alpha) \quad \text{if} \quad n = 1 \tag{11}$$

3. To minimize the magnitude

$$D = \sqrt{\frac{\Sigma (B_i - \bar{B})^2}{M\bar{B}^2}} \tag{12}$$

by performing the systematic variation of E and n, D is preferred to δ, since it is an absolute measure of the constancy of B_i, unlike δ, which is proportional to $\bar{B}$.

In order to reduce the calculation time, the following procedure has been accepted:

For $n = 0$, D_{i0} is calculated by using successive E_{i0} values, according to the formula

$$E_{i+1.0} = E_{i0} + \Delta_i \quad \text{where} \quad \Delta_i = \begin{cases} \Delta_{i-1}, & \text{if} \quad D_{i0} < D_{i-1.0} \\ -\frac{1}{2}\Delta_{i-1}, & \text{if} \quad D_{i0} > D_{i-1.0} \end{cases}$$

and by taking $E_{00} = \Delta_0 = 10\,000$ joule/mol. If $|\Delta_k| < 100$ joule/mol, minimization is considered as achieved and D_{k0} is obtained.

The parameter n is varied according to the same principle, i.e.

$$n_{j+1} = n_j + \Delta_j' \quad \text{where} \quad \Delta_j' = \begin{cases} \Delta_{i-1}', & \text{if} \quad D_{kj} < D_{k,i-1} \\ -\frac{1}{2}\Delta_{j-1}', & \text{if} \quad D_{kj} > D_{k,j-1} \end{cases}$$

by taking $n_0 = 0$ and $\Delta_j' = 1$. If $|\Delta_m'| < 0.01$, the double minimization is considered to be achie ved. The obtained n_m and E_{km} values are the kinetic parameters which were to be determined, and the corresponding D_{km} value characterizes the agreement of the experimental data with the kinetic parameters derived.

The calculation of the pre-exponential factor can be performed by means of the formula

$$\log Z = \bar{B} + \log Rq - \log E \tag{13}$$

The following data are introduced:

– the first card contains the number of experimental points M, the initial sample weight G_0, the final sample weight G_1 and the heating rate q in $K\,s^{-1}$,

– the second card contains the M sample weight values, denoted by $C(I)$,

– the third card contains the corresponding M temperature values in °C.

The computer program in Fortran language is the following:

```
C METHOD 1                                                              1
*DEFINE FILE *1 = 105                                                   2
*DEFINE FILE *2 = 108                                                   3
      DIMENSION C(20),T(20),B(20)                                       4
      REAL N                                                            5
      COMMON T,C,N,SSB,SD,M,A                                           6
  100 READ (105,20,END = 50) M,GO,G1,Q                                  7
   20 FORMAT (I2,1X,F4.2,1X,F4.2,1X,F4.4)                               8
  160 WRITE (108,150) M,GO,G1,Q                                         9
  150 FORMAT (1H, 10X,2HM = ,I2,10X,3HGO = ,F5.2,10X,3HG1 = ,           10
     *F5.2,10X,2HQ = ,F5.4)                                             11
   72 IF (M.LE.O.OR.G1.LE.O.OR.GO.LE.G1.OR.Q.LEO.) GO TO 70             12
      READ (105,21,END = 51) (C(I),I = 1,M)                             13
   21 FORMAT (20F4.2)                                                   14
      WRITE (108,151) (I,C(I),I = 1,M)                                  15
  151 FORMAT (1H,10X,20(2HC(,I1,2H) = ,F5.2,5X))                        16
      DO 77 I = 1,M                                                     17
      IF (C(I).LE.G1.OR.C(I).GE.GO) GO TO 74                            18
   77 CONTINUE                                                          19
      READ (105,22,END = 51) (T(I),I = 1,M)                             20
   22 FORMAT (20F4.1)                                                   21
      WRITE (108,152) (I,T(I),I = 1,M)                                  22
  152 FORMAT (1H ,10X,20(2HT(,I1,2H) = ,F5.1,5X))                       23
      GO TO 41                                                          24
   50 STOP                                                              25
   70 WRITE (108,71)                                                    26
   71 FORMAT (1H ,51HN,G1,OR Q LESS OR EQUAL ZERO OR GO                 27
     *LESS OR EQUAL G1)                                                 28
      READ (105,73,END = 50) M,GO,G1,Q                                  29
   73 FORMAT (//I2,2(1X,F4.2),1X,F4.4)                                  30
      GO TO 160                                                         31
   74 WRITE (108,75)                                                    32
   75 FORMAT (1H ,5X,20HDATA ARE NOT CORRECT)                           33
      READ (105,76,END = 50) M,GO,G1,Q                                  34
   76 FORMAT (/I2,2(1X,F4.2),1X,F4.4)                                   35
      GO TO 160                                                         36
   51 WRITE (108,52)                                                    37
   52 FORMAT (1H ,10X,21HDATA ARE NOT COMPLETE)                         38
      GO TO 50                                                          39
   41 N = O                                                             40
      SS = 1.                                                           41
```

```
      DO 23 I = 1,M                                                    42
      IF (T(I).LE.O) GO TO 99                                          43
      C(I) = (C(I)-G1)/(GO-G1)                                         44
   23 T(I) = 1/(273.1 + T(I))                                          45
   30 CALL ALFA                                                        46
      IF (N) 24,25,24                                                  47
   25 SSSB = SSB                                                       48
      SSD = SD                                                         49
      SA = A                                                           50
      GO TO 26                                                         51
   24 IF (SD.LT.SSD) GO TO 25                                          52
      SS = -0.5*SS                                                     53
      IF (ABS(SS)-0.01) 28,25,25                                       54
   26 N = N + SS                                                       55
      GO TO 30                                                         56
   28 SA = 8.314*SA                                                    57
      Z = SSSB-ALOG10(SA) + ALOG10(8.314*Q)                            58
      WRITE(108,35)N,SA,Z,SSD                                          59
   35 FORMAT (1H ,10X,2HN = ,F5.2,10X,2HE = ,F8.0,10X,                 60
     *7HLOG(Z) = ,F5.1,10X,2HD = ,F6.5)                                61
      GO TO 100                                                        62
   99 WRITE (108,98) (T(I),I = 1,M)                                    63
   98 FORMAT (1H ,20F5.1)                                              64
      GO TO 100                                                        65
      END                                                              66

      SUBROUTINE ALFA                                                   1
      DIMENSION T(20),C(20,B(20),R(20)                                  2
      REAL N                                                            3
      COMMON T,C,N,SSB,SD,M,A                                           4
      P(X) = 0.43429*X + ALOG10((X + 2)*(X - 16/(X**2-4*X +             5
     *+ 84)))                                                           6
      G(X) = (X**(1.-N)-1.)/(N-1.)                                      7
      A = 10000                                                         8
      S = 10000
      IF (N.EQ.1) GO TO 2                                               9
      DO 3 I = 1,M                                                     10
    3 R(I) = G(C(I))                                                   11
      GO TO 12                                                         12
    2 DO 18 I = 1,M                                                    13
   18 R(I) = -ALOG(C(I))                                               14
   12 SB = 0                                                           15
      DO 17 I = 1,M                                                    16
      X = P(T(I)*A)                                                    17
```

```
17 B(I) = ALOG10(R(I)) + X                                    18
   DO 6 I = 1,M                                               19
 6 SB = SB + B(I)                                             20
   BM = SB/M                                                  21
   SBB = 0                                                    22
   DO 7 I = 1,M                                               23
 7 SBB = SBB + (B(I)-BM)**2                                   24
   D = SBB/M/BM**2                                            25
   IF (A-10000.) 8,9,8                                        26
 9 SSB = BM                                                   27
   SD = D                                                     28
   GO TO 10                                                   29
 8 IF (D.LT.SD) GO TO 9                                       30
70 S = -0.5*S                                                 31
   IF (ABS(S),LT.100) GO TO 11                                32
   GO TO 9                                                    33
10 A = A + S                                                  34
   IF (A.EQ.O) GO TO 70                                       35
11 RETURN                                                     36
   END                                                        37
```

E values printed are given in joule/mol.

Method 2

This method is a computerized variant of the Coats–Redfern method [7], based on the following approximation of the exponential integral:

$$p(x) \approx e^{-x} \frac{x-2}{x^3}$$

By using this expression, Eq. (4) can be written in the following form:

$$\log \frac{g(\alpha)}{T^2} = \log \frac{ZR}{qE}\left(1 - \frac{2RT}{E}\right) - \frac{E}{2.3RT}$$

Coats and Redfern propose a graphical trial-and-error method, consisting in the plot of $\log g(\alpha)/T$ *vs.* $1/T$, by presuming different n values in the calculation of $g(\alpha)$. Thus, the best linearity indicates the correct n value, and from the slope of the straight line E can easily be derived.

In the present paper we propose the application of the least square method to calculate the slope of the straight line corresponding to the above graphical plot,

and the systematic variation of n, until for Jaffé's correlation coefficient R [8] the maximum value is obtained.

The procedure involves the calculation of R_i by using

$$n_{i+1} = n_i + \Delta_i$$

$$\text{with} \quad \Delta_i = \begin{cases} \Delta_{i-1}, & \text{if} \quad R_i > R_{i-1} \\ -\frac{1}{2}\Delta_{i-1}, & \text{if} \quad R_i < R_{i-1} \end{cases}$$

and by taking $n_0 = 0$, $\Delta_0 = 1$. The true n value is considered to be found if $|\Delta_n| < 0.01$. The corresponding E value is calculated by multiplying the slope of the straight line by $2.3R$.

In order to obtain the pre-exponential factor, the

$$\log Z = \log g(\alpha) + \log Rq - \log E - \log p(x)$$

value is derived for each experimental point and their arithmetical mean is calculated.

The same cards can be used as with Method 1 and the computer program is the following:

```
C METHOD 2                                                          1

*DEFINE FILE *1 = 105                                               2
*DEFINE FILE *2 = 108                                               3
      DIMENSION C(20),G(20),T(20),X(20),Y(20)                       4
      REAL N,NA                                                     5
      P(T) = 0.43429*T + ALOG10((T + 2)*(T-16/(T**2-4*T + 84)))     6
  lines 7 to 40 are identical with the same lines in Method 1
      DO 30 I = 1,M                                                 41
      C(I) = (GO-C(I))/(GO-G1)                                      42
   30 X(I) = 1000/(273.1 + T(I))                                    43
      S = 1.                                                        44
      SX = 0                                                        45
      SXX = 0                                                       46
      DO 2 I = 1,M                                                  47
      IF (X(I),LE,O) GO TO 25                                       48
      D(I) = 2*ALOG10(X(I))                                         49
      SX = SX + X(I)                                                50
    2 SXX = SXX + X(I)**2                                           51
      S1 = SXX-SX**2/M                                              52
   17 IF (N-1.) 5,6,5                                               53
    5 DO 7 I = 1,M                                                  54
```

```
      7 G(I) = ((1.-C(I))**(1.-N)-1.)/(N-1.)                                55
        GO TO 8                                                             56
      6 DO 9 I = 1,M                                                        57
      9 G(I) = -ALOG(1.-C(I))                                               58
      8 SY = 0                                                              59
        SYY = 0                                                             60
        SXY = 0                                                             61
        DO 10 I = 1,M                                                       62
        Y(I) = ALOG10(G(I)) + D(I)                                          63
        SY = SY + Y(I)                                                      64
        SYY = SYY + Y(I)**2                                                 65
     10 SXY = SXY + X(I)*Y(I)                                               66
        S2 = SYY-SY**2/M                                                    67
        S3 = SXY-SX*SY/M                                                    68
        SL = S3/S1                                                          69
        R = SQRT(S3**2/S1/S2)                                               70
        IF (N) 14,15,14                                                     71
     15 SLA = SL                                                            72
        RA = R                                                              73
        GO TO 18                                                            74
     14 IF (R,GT,RA) GO TO 15                                               75
        S = -S*0.5                                                          76
        IF (ABS(S)-0.01) 16,15,15                                           77
     16 SLA = -19.122*SLA                                                   78
        ZB = 0                                                              79
        MA = M                                                              80
        DO 90 M = 1,MA                                                      81
        Z = ALOG10(G(M)) + P(X(M)*SLA/8.314-ALOG10(SLA/8.314/Q)             82
     90 ZB = ZB + Z                                                         83
        WRITE (108,19) N,SLA,Z,RA                                           84
     19 FORMAT (1H ,10X,2HN = ,F5.2,10X,2HE = ,F8.0,10x,                    85
        *7HLOG(Z) = ,F5.1,10X, 2HR = ,F6.4)                                 86
        GO TO 100                                                           87
     18 N = N + S                                                           88
        GO TO 17                                                            89
     25 WRITE (108,26) (X(I),I = 1,M)                                       90
     26 FORMAT (1H ,20F4.1)                                                 91
        GO TO 100                                                           92
        END                                                                 93
```

E values printed are given in joule/mol.

Method 3

This method, proposed in our previous paper [9] and computerized in the present one, is based upon the same Coats-Redfern method as Method 2. The difference consists in the following. Experimental sample weight data are not introduced directly into the computer, but first a graphical plot of $(1 - \alpha)$ *vs.* T is performed. By graphical means, nine temperature values are derived, corresponding to the following α values: 0.1, 0.2, 0.3, . . . 0.9.

As in the case of Method 2, $\log g(\alpha)/T^2$ values are calculated by presuming different n values. The parameters of the straight line given by the nine $\log g(\alpha)/T^2 vs\ 1/T$ pairs are derived by means of the least square method. The maximum of the correlation coefficient indicates the true n value. E and $\log Z$ values are calculated as before.

Data are introduced by means of a single card, containing the heating rate in $K \cdot s^{-1}$, and the nine $10^3\ T^{-1}$ values.

The computer program is the following:

```
C METHOD 3                                                          1
*DEFINE FILE*1 = 105                                                2
*DEFINE FILE*2 = 108                                                3
      DIMENSION A(9),C(9),D(9),X(9),1(9)                            4
      REAL N                                                        5
      DATA A /0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9/                  6
      P(T) = 0.43429*T + ALOG10((T + 2)*(T-16/(T**2-4*T + 84)))     7
    3 READ (105,1,END = 20) Q,X                                     8
    1 FORMAT (F4.4,3X,9F5.4)                                        9
      N = 0                                                         10
      S = 1.                                                        11
      SX = 0                                                        12
      SXX = 0                                                       13
      DO 2 I = 1,9                                                  14
      IF (X(I).LE.O) GO TO 25                                       15
      D(I) = 2*ALOG10(X(I))                                         16
      SX = SX + X(I)                                                17
    2 SXX = SXX + X(I)**2                                           18
      S1 = SXX-SX**2/9                                              19
   17 IF (N-1.) 5,6,5                                               20
    5 DO 7 I = 1,9                                                  21
    7 C(I) = ((1.-A(I))**(1.-N)-1.)/(N-1.)                          22
      GO TO 8                                                       23
    6 DO 9 I = 1,9                                                  24
    9 C(I) = -ALOG(1.-A(I))                                         25
```

```
 8 SY = 0                                                              26
   SYY = 0                                                             27
   SXY = 0                                                             28
   DO 10 I = 1,9                                                       29
   IF (C(I).LE.O) GO TO 25                                             30
   Y(I) = ALOG10(C(I)) + D(I)                                          31
   SY = SY + Y(I)                                                      32
   SYY = SYY + Y(I)**2                                                 33
10 SXY = SXY + X(I)*Y(I)                                               34
   S2 = SYY-SY**2/9                                                    35
   S3 = SXY-SX*SY/9                                                    36
   SL = S3/S1                                                          37
   R = SQRT(S3**2/S1/S2)                                               38
   IF (N) 14,15,14                                                     39
15 SLA = SL                                                            40
   RA = R                                                              41
   GO TO 18                                                            42
14 IF (R.GT.RA) GO TO 15                                               43
   S = -S*0.5                                                          44
   IF (ABS(S)-0.01) 16,15,15                                           45
16 SLA = - 19.122/*SLA                                                 46
   IF (Q.LE.O) GO TO 25                                                47
   ZB = 0                                                              48
   DO 90 I = 1,9                                                       49
   Z = ALOG10(C(I)) + P(X(I)*SLA/8.314)                                50
90 ZB = ZB + Z                                                         51
   Z = ZB/9-ALOG10 (SLA/Q/8.314)-3.                                    52
   WRITE (108,19) (I,X(I),I = 1,9)                                     53
19 FORMAT (1H ,10X,9(2HX(,I1,2H) = ,F6.4,5X))                          54
   WRITE (108,21) N,SLA,Z,RA                                           55
21 FORMAT (1H ,10X,2HN = ,F5.2,10X,2HE = ,F7.2,10X,7HLOG               56
   *(Z) = ,F6.2,10X, 2HR = ,F6.4)                                      57
   GO TO 3                                                             58
18 N = N + S                                                           59
   GO TO 17                                                            60
25 WRITE (108,26)                                                      61
26 FORMAT (1H ,5X,20HDATA ARE NOT CORRECT)                             62
   GO TO 3                                                             63
20 STOP                                                                64
   END                                                                 65
```

Testing of the methods

The above methods have been tested by using both theoretical and experimental TG curves.

Theoretical TG curves have been obtained by means of Eqs (4), (7) and (8), by using the n, E and $\log Z$ values indicated in Table 1 as "theoretical" values, and by considering a heating rate of 1°/min. Kinetic parameters n, E and $\log Z$ derived from these theoretical curves by means of Methods 1–3 are also given in Table 1.

Experimental TG curves have been obtained by using a thermobalance constructed on the basis of literature data [10]. The kinetics of the dehydration of $Ca(COO)_2 \cdot 2\,H_2O$ have been studied by means of the above three methods. Sample weight and heating rate values are presented in Table 2, while the kinetic parameter values n, E and $\log Z$, derived by means of Methods 1–3, are given in Table 3.

Table 1

Testing of the methods on theoretical TG curves

Kinetic parameter	No. of the TG curve	Theoretical value	Method		
			1	2	3
n	1	1.00	1.00	1.02	1.02
	2	1.00	1.02	1.00	1.00
	3	0.00	0.03	0.03	0.03
E kJ/mol	1	41.8	42.3	41.4	41.4
	2	83.7	84.5	83.3	83.3
	3	167.4	168.2	167.8	167.8
$\log Z$	1	1.0	1.0	0.9	0.9
	2	10.0	10.1	9.9	9.9
	3	20.0	20.1	20.0	20.0

Table 2

Dehydration of $Ca(COO)_2$, $2\,H_2O$. Sample weight and heating rate values

No. of the TG curve	Sample weight, mg	Heating rate, °/min
4	25	5
5	75	5
6	100	15

Table 3

Testing of the methods on experimental TG curves

Kinetic parameter	No. of the TG curve	Method		
		1	2	3
n	4	0.72	0.72	0.75
	5	1.05	1.05	1.00
	6	0.55	0.55	0.50
E kJ/mol	4	143.9	142.7	141.8
	5	108.4	107.9	101.3
	6	65.3	64.0	63.2
$\log Z$	4	14.5	14.8	14.5
	5	10.4	10.6	9.5
	6	4.4	4.7	4.6

As seen from Tables 1 and 3, the three methods give practically the same results. Method 1 performs a double minimization, and consequently this method requires approximately a ten-fold computer time as compared to Methods 2 and 3.

On the other hand, in the case of experimental TG data, Method 3 seems not to be perfectly equivalent to Methods 1 and 2. This is easy to understand, since the data introduced are not the same. Methods 1 and 2 use sample weight data, which are generally recorded at equal time intervals, i.e. more experimental points are taken from the first part of the TG curves than from the second part, where the reaction rate is much higher. In Method 3 the experimental points considered are uniformly distributed over the whole TG curve.

References

1. C. D. DOYLE, J. Appl. Polymer Sci., 15 (1961) 285
2. J. ZSAKÓ, J. Thermal Anal., 5 (1973) 239
3. G. I. SENUM and R. T. YANG, J. Thermal Anal., 11 (1977) 445
4. J. ZSAKÓ, J. Thermal Anal., 8 (1975) 593
5. J. ZSAKÓ, J. Phys. Chem., 72 (1968) 2406
6. J. SIMON, E. DEBRECZENY and É. BUZÁGH-GERE, Periodica Polytech. Chem. Eng., Budapest, 17 (1973) 203
7. A. W. COATS and J. P. REDFERN, Nature, 201 (1964) 68
8. H. H. JAFFÉ, Chem. Rev., 53 (1953) 191
9. J. ZSAKÓ, M. VÁRHELYI and Cs. VÁRHELYI, J. Thermal Anal, 17 (1979) 123.
10. L. ERDEY and F. PAULIK, Magy. Tud. Akad. Kém. Oszt. Közl., 5 (1955) 461, L. KÉKEDY, P. KRÖBL, A. SZURKOS and E. KÉKEDY, Studia Univ. Babes-Bolyai, Ser. Chem. 3 (1958) 99

RÉSUMÉ — On donne un programme d'ordinateur en langage Fortran pour trois méthodes intégrales de déduction des paramètres cinétiques des courbes TG. La première méthode est une variante pour ordinateur de la méthode d'ajustement des courbes de Doyle qui effectue

le calcul de l'intégrale exponentielle $p(x)$ à l'aide de la formule empirique de l'auteur. Les deuxième et troisième méthodes sont des variantes de la méthode de linéarisation de Coats-Redfern. L'application des trois méthodes à des courbes TG théoriques et expérimentales a montré qu'elles étaient presque équivalentes, mais la première méthode exige dix fois plus de temps d'ordinateur.

ZUSAMMENFASSUNG — Ein Komputerprogramm in Fortransprache wird für drei Integralmethoden zur Ableitung kinetischer Parameter aus TG-Kurven gegeben. Methode 1 ist eine komputerisierte Variante der Kurvenanpassungsmethode von Doyle und führt die Berechnung des exponentiellen Integrals $p(x)$ mit Hilfe der empirischen Formel des Autors durch. Die Methoden 2 und 3 sind Varianten der Linearisierungsmethode von Coats-Redfern. Die Prüfung der Methoden an theoretischen und an experimentellen TG-Kurven zeigt diese hinsichtlich des Ergebnisses als nahezu gleichwertig, jedoch benötigt Methode 1 eine zehnfache Komputerzeit.

Резюме — Представлена вычислительная программа на языке Яортран для трех интегральных методов извлечения кинетических параметров из кривых ТГ. Метод 1 является машинным вариантом метода Дойля пригонки кривой и выполняет вычисления экспоненциального интеграла $p(x)$ с помощью авторской эмпирической формулы. Методы 2 и 3 являются разновидностями метода линеаризации Коутса—Редферна. Проверка методов на теоретических и экспериментальных кривых ТГ показала их равнозначность. Орнако метод 1 требует в десять раз больше машинного времени.