Thermochimica Acta, 20 (1977) 279–296
© Elsevier Scientific Publishing Company, Amsterdam – Printed in The Netherlands

NON-ADIABATIC TREATMENT OF DIATHERMIC-CHEMICAL-REACTION-DATA BY THE INTEGRATIONAL MINIMIZATION METHOD

W. FRANKVOORT
Laboratorium voor Chemische Technologie, University of Amsterdam (The Netherlands)

(Received 23 September 1976)

ABSTRACT

The present paper is concerned with the analysis of temperature time data from diathermic-chemical liquid reactions. In addition, mathematical methods are described for calculating the activation energy and the frequency factor from the observed temperature–time relation.

1. INTRODUCTION

In the determination of rate constants in a kinetic model on the basis of experimental data, the complexity of the model indicates the approach. The problem is relatively easy to solve when the kinetic model is sufficiently simple as shown by Dammers and Frankvoort[1, 2].

Apart from the just mentioned differential method, the analytical integration of the differential equation can lead to simple objective functions in the regression analysis.

However, when the model is complex and the number of differential equations increases, only numerical methods can handle the difficulties.

As already shown[2] determination of kinetic constants in semi-adiabatic systems gives two alternatives of solving the problems. The one in which a perfectly adiabatic reaction calorimeter is used, is not considered here. The other deals with the numerical treatment which considers the small corrections resulting from the deviations from adiabaticity[2].

The objective of this paper is to develop a computational program for the estimation of parameters in ordinary differential equations.

Considering the fact, that temperature (measured at equidistant times) is the only detected variable, the problem can be formulated by requiring that:

$$S(\text{par}) = \sum_{i=1}^{N} (T_{i,\text{exp}} - T_{i,\text{cal}})^2 \qquad (1)$$

should be minimized with respect to the parameters. The entire problem can be separated into two major parts: the computation of the temperatures $T_{i,\text{cal}}$ and the method used to minimize S(par).

Because of the large number of data-points, both phases of the problem require considerable computer facilities. Programs have been written in Algol-60 for the CDC-73-28 (see section 2). The program is illustrated with a few examples from the literature in section 3. In section 4 we introduce the measuring method and the reaction used. Finally we discuss the results and method in section 5.

2. THEORETICAL PART

2.1. *Basic equations*

The system under consideration can mathematically be stated by[3–7]

$$\frac{dy}{dt} = f(t, y, \text{par}) \qquad 0 \leqslant t \leqslant \tau \tag{2}$$

$$y(0, \text{par}) = y_0 \tag{3}$$

where $y(t,\text{par})$ is an n-vector, par is an m-vector of constant but unknown parameters and t denotes time.

We assume that f, $\partial f/\partial y$, $\partial f/\partial\text{par}$, y and $\partial y/\partial\text{par}$ are continuous in y and par and piece-wise continuous with respect to t. The experimental values are denoted by $\hat{y}(t)$, an n-vector related to the state by

$$\hat{y}(t) = y(t, \text{par}) + \varepsilon \tag{4}$$

where ε is an n-vector of errors.

Our aim is to find a set of parameters minimizing the quadratic function $S(\text{par})$ defined by:

$$S(\text{par}) = \int_0^\tau \|\hat{y}(t) - y(t, \text{par})\|_2^2 \, dt \tag{5}$$

where $\|v\|_2$ denotes the euclidian-norm given by:

$$\|v\|_2 \overset{\text{def}}{=} (v^* v)^{\frac{1}{2}} \tag{6}$$

The asterisk stands for the transposition. Using an integration procedure to solve $y(t,\text{par})$ we can minimize $S(\text{par})$ using standard techniques.

Because the observations are made only at discrete times we must replace (5) by:

$$S(\text{par}) = \sum_{i=1}^{N} (\hat{y}(t_i) - y(t_i, \text{par}))^2 \tag{7}$$

To find the final least squares estimate $\overline{\text{par}}$ we must satisfy the normal equations (m-equations)[10]:

$$\sum_{i=1}^{N} (\hat{y}(t_i) - y(t_i, \overline{\text{par}})) \cdot \left\{ \frac{\partial y(t_i, \text{par})}{\partial \text{par}_j} \right\}_{\text{par}=\overline{\text{par}}} = 0 \tag{8}$$

for $j = 1, 2 \ldots m$.

The $n \times m$ matrix $\partial y(t,\text{par})/\partial\text{par}$ can be evaluated by numerical differentiation[8,9] or by solving the set of linearized differential equations[3–6]

$$\frac{\mathrm{d}\,YP}{\mathrm{d}t} = FP + FY \cdot YP; \qquad YP(0) = 0 \tag{9}$$

where

$$YP = \frac{\partial y(t, \text{par})}{\partial \text{par}}$$

$$FY = \frac{\partial f(t, y, \text{par})}{\partial y} \tag{10}$$

$$FP = \frac{\partial f(t, y, \text{par})}{\partial \text{par}}$$

As shown in the next section we prefer the last method because we can integrate simultaneously eqns (2) and (9) owing to the similarity of the jacobians of both systems[3, 6].

2.2. *The integration method*

We tried to combine several features of the predictor–corrector-linear multistep integration methods developed by Adams et al.[7]. Our goals are to realize the following demands:

(a) variable step size
(b) realising a robust procedure if possible
(c) handling normal and stiff systems of differential equations
(d) simultaneous integration of eqns (2) and (9).

Using Adams–Moulton–Bashford methods for integrating $\mathrm{d}y/\mathrm{d}t = f(t,y,\text{par})$ the corrector-predictor procedure can be written as[3, 7, 12]:

$$_{(m+1)}y = {}_{m}y + c\,h\,(I - h\beta J)^{-1}\,(f(t, {}_{m}y, \text{par}) - {}_{(m)}y') \tag{11}$$

and

$$_{(m+1)}y' = {}_{m}y' + (I - h\beta J)^{-1}\,(f(t, {}_{m}y, \text{par}) - {}_{(m)}y') \tag{12}$$

where h is the stepsize used, c and β are functions of t and J ($= \partial f(t,y,\text{par})/\partial y$) is the jacobian of the system. $_{(m+1)}y$ and $_{(m+1)}y'$ are improvements of $_{m}y$ and $_{m}y'$, respectively. For the integration of eqn (9) we find a similar set of equations as written in (11) and (12). In each step we solve first eqns (11) and (12) iteratively and then eqn (9) by substitution of c, h and $(I - h\beta J)^{-1}$, the same factors used in (11) and (12). The Adams–Moulton–Bashford methods and the Curtis–Hirschfelder methods can be implemented in such a way that one can switch during the integration from normal to stiff differential equations[7, 12]. All these features are not implementable in a norma

Runge–Kutta integration method. Finally, we will state that the matching in time with the data points is realized by Newton interpolation of the temperature with the same order of the stability polynomial in the numerical integration rather then by stepsize matching during the integration.

2.3. *The minimization procedure*

To find the least squares approximation of par in $S(\text{par})$ we have to fulfill the condition $\partial S(\text{par})/\partial\text{par} = 0$. It is not possible to solve the parameters directly because of the non-linearity of the equations. So we have to iterate from a given starting point to the minimum of the function $S(\text{par})$.

Using the method suggested by Marquardt[4, 8, 10, 13–20] we can write for normal equation:

$$(J^*J + \lambda I)\Delta = J^*(\hat{y}(t) - y(t, \text{par})) \tag{13}$$

where λ is a real constant chosen in each iteration in accordance with several rules, I is the identity matrix and Δ is the correction on the parameters

$$\text{par}^{j+1} = \text{par}^j + \Delta \tag{14}$$

When $\lambda = 0$ eqn (13) gives a pure Gauss–Newton correction. For $\lambda \to \infty$ the algorithm tends to a steepest descend correction. It is shown[15, 19, 20] that there exists a $\lambda < \infty$ for which

$$S(\text{par}^j + \Delta^j) < S(\text{par}^j) \tag{15}$$

and there exists also a $\lambda > \sigma^2_{\min}$ such that

$$S(\text{par}^j + \Delta^j) > S(\text{par}^j) \tag{16}$$

where $\sigma^2_{\min}$ is the smallest singular value of J^*J. This means there exists at least one λ, viz. $\sigma^2_{\min} < \lambda < \infty$, at which the inequality (15) is fulfilled. There are several procedures for calculating λ[17–20, 14, 15]. The one we use is the following. We take the condition number μ of the matrix J^*J as an initial guess for λ, given by

$$\lambda^0 = \text{alfa} \cdot \sigma^0_{\max}/\sigma^0_{\min}. \tag{17}$$

where alfa is a user definable correction factor. If the correction Δ^j on the parameters is accepted we replace λ^j by:

$$\lambda^{j+1} = \text{beta} \cdot \lambda^j \tag{18}$$

where beta is a user definable reducing factor. If the correction Δ^j is not accepted we replace λ^j by:

$$\lambda^{j+1} = 10 \cdot \lambda^j \tag{19}$$

On approaching the minimum, the minimization method of Marquardt leads to the Gauss–Newton method and λ^j forms a decreasing sequence in subsequent iterations.

Starting with an initial guess λ^0 we find the first correction on the parameters by

$$\Delta^0 = (J^*J + \lambda^0 I)^{-1} J^*b \tag{20}$$

where $b = \hat{y}(t) - y(t,\text{par})$. This equation might be solved by inversing $(J^*J + \lambda^0 I)$. We can, however, avoid these difficulties by using the singular-value decomposition method[13].

If J is a real $m \times n$ matrix we can write:

$$J = U\Sigma V^* \tag{21a}$$

where $UU^* = I$ and $VV^* = I$ and (21b)

$$\Sigma = \begin{vmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \end{vmatrix} \tag{21c}$$

and σ_j are the singular values. Substitution in eqn (20) gives for the j-th iteration by dropping the indices:

$$\Delta = V(\Sigma^2 + \lambda I)^{-1}\, \Sigma\, U^*b \tag{22}$$

The iterative algorithm based on these principles is the following:

```
(a) guess parameter alfa: $\lambda^0 =$ alfa $\cdot \sigma^0_{\max}/\sigma^0_{\min}$ and par$^0$
(b) given par$^j$, determine par$^{j+1}$ by performing the following steps:
    (1) solve the singular-value decomposition for $J$
    (2) solve eqn (22) for $\Delta^j$
    (3) solve $S(\text{par}^j + \Delta^j)$
    (4) if $S(\text{par}^j + \Delta^j) < S(\text{par}^j)$ then par$^{j+1}$ = par$^j + \Delta^j$;
            $\lambda^{j+1} =$ beta $\cdot \lambda^j$; go to (1)
        else $\lambda^{j+1} = 10 \cdot \lambda^j$; go to (2).
```

2.4. The complete program

As we can see in Fig. 1 we always perform a complete integration of eqns (2) and (9) except when condition (15) is not fulfilled. Then eqn (19) is used to change λ in order to meet condition (15). Subsequently we have to make one complete integration of (2) and (9) together for the new singular-value decomposition. During the whole calculation we avoid λ to decrease the value of $\sigma^2_{\min}$.

It is neither necessary nor useful to start with the same tolerance on the integration we need at the end. So we change it during the calculation. We found it useful to introduce some restrictions on the values of the parameters (dictated by physical conditions), which means that the parameter values are checked against user definable boundaries. The program can produce information about boundaries constraint jumps, the values of parameters, λ and the least square sum, $S(\text{par})$, all during the iterations.

In Appendix I we give the heading of the minimization procedure including the meaning of the formal parameters. The procedure will be available on microfilm.

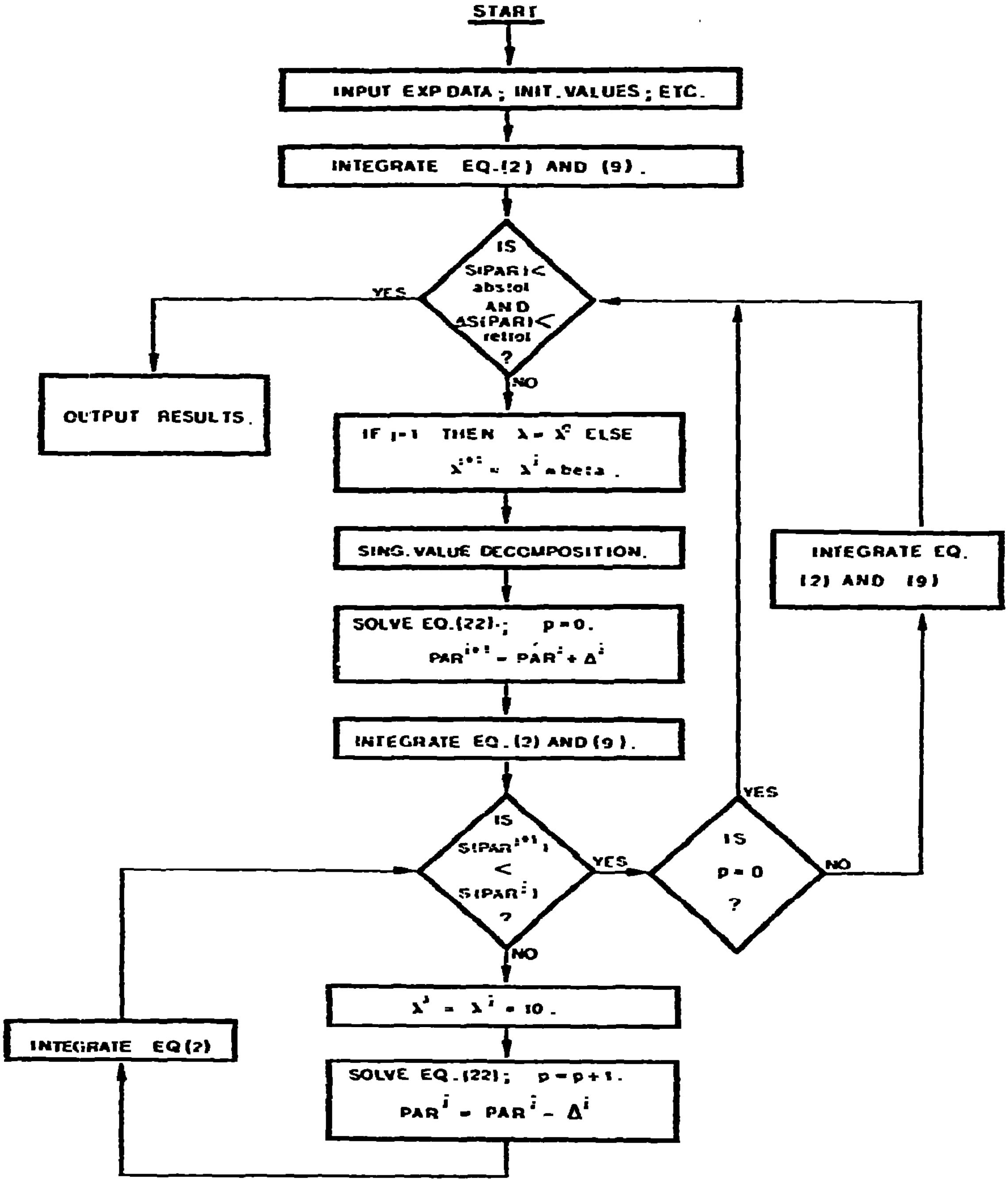

Fig. 1. Flow diagram of the algorithm.

3. TEST EXAMPLES

We used three examples to test our program. The first and the last are of our own[2], the other is the one given by Bellman et al.[21]. In diathermic reaction calorimetry the system is described by the two differential equations:

$$\frac{dT}{dt} = -\frac{Q}{c_p}\frac{dm}{dt} - \frac{hA}{c_w}(T - T_c) + \frac{W_s}{c_w} \qquad -\frac{dm}{dt} = Z_n m^n \exp\left(\frac{-T_a}{T}\right) \quad (23)$$

For a detailed description see refs. 1 and 2. In here Q/c_p, Z_n and T_a are unknown parameters, to determine in the minimization and n, hA/c_w, T_c and W_s/c_w are known constants. Using the values mentioned in Table 1 we calculated an exact T,t-data set.

TABLE 1

VALUES OF VARIABLES USED IN CALCULATING T, t-DATA SET

$T(t = 0)$	= 278.000 K	Q/c_p	= 17.5 K kg mol^{-1}
T_c	= 278.000 K	hA/c_w	= 1.6 · 10^{-5} s^{-1}
T_a	= 9000.00 K	W_s/c_w	= 4.3 · 10^{-4} K s^{-1}
$m(t = 0)$	= 2.0 mol kg^{-1}	Z_1	= 5 · 10^{11} s^{-1}

With this data set and some starting values for the parameters we minimized $S(T_a, Z_n, Q/c_p)$ with the results tabulated in Table 2.

TABLE 2

RESULTS OF THE MINIMIZATION OF EQN (23) WITH THE DATA SET BASED ON THE VALUES FROM TABLE 1

Parameter	*Starting values*	*Final values*
T_a	9.200000 · 10^3	9.0000002 · 10^3 K
Z_1	4.500000 · 10^{11}	5.0000038 · 10^{11} s^{-1}
Q/c_p	1.800000 · 10^1	1.7500000 · 10^1 K kg mol^{-1}
S(par)	2.788 · 10^4	2.403 · 10^{-12} K^2

We used for this minimization 17 integrations of eqns (2) and (9) together and zero integrations of eqn (2) alone. The calculations started with $\lambda = 1.205 \cdot 10^3$ and finished, monotoneously descending, with $\lambda = 3.677 \cdot 10^{-2}$.

Bellman et al.[21] give an example of the reversible homogeneous gas phase reaction

$$2\,NO + O_2 \underset{k_2}{\overset{k_1}{\rightleftarrows}} 2\,NO_2$$

with data of Bodenstein and Lindner[22].

The descriptive differential equation has the form

$$\frac{dy}{dt} = k_1 (a - y)(b - y)^2 - k_2 y^2 \tag{24}$$

where $a = 126.2$ and $b = 91.9$ and k_1 and k_2 the rate constants to be estimated. The results are tabulated in Table 3 including the results of Bellman[21] and Hemker[3].

TABLE 3

RESULTS OF THE MINIMIZATION OF EQN (24) WITH THE DATA SET OF BODENSTEIN AND LINDNER

Parameter	*Starting values*	*Final values*		
		Ours	*Bellman*	*Hemker*
k_1	$1.000 \cdot 10^{-6}$	$4.57 \cdot 10^{-6}$	$4.577 \cdot 10^{-6}$	$4.5 \cdot 10^{-6}$
k_2	$1.000 \cdot 10^{-4}$	$2.78 \cdot 10^{-4}$	$2.797 \cdot 10^{-4}$	$2.7 \cdot 10^{-4}$
$S(\text{par})$	$4.084 \cdot 10^{2}$	21.9	21.0	22.0

We started the calculations with a λ of $2.83 \cdot 10^8$ and finished after five integrations of eqns (2) and (9) together and zero integrations of eqn (2) alone with a λ of $3.53 \cdot 10^7$.

The third example deals with the equation appearing in perfectly adiabatic reaction systems. Frankvoort and Dammers[2] showed that for the n-th order adiabatic reaction the rate of temperature change is given by

$$\frac{dT}{dt} = Z_n (T_m - T)^n \exp\left(-\frac{T_s}{T}\right) \tag{25}$$

where

$$Z_n = Z \left[\frac{m_0}{T_m - T_0}\right]^{n-1} \tag{26}$$

The importance of this equation lies in the simultaneous determination of all the kinetic parameters. Using the values mentioned in Table 4 we calculated an exact T,t-data set.

TABLE 4

VALUES OF VARIABLES USED IN CALCULATING AN EXACT T, t-DATA SET

$T(t = 0) = 2.780000 \cdot 10^3$ K
$T_s = 9.000000 \cdot 10^3$ K
$Z_n = 5.000000 \cdot 10^{11}$ mol kg^{-1} s^{-1}
$n = 1.000000$ —
$T_m = 3.130000 \cdot 10^3$ K
$m(t = 0) = 2.000000$ mol kg^{-1}

With this data set and some starting values for the parameters we minimized $S(T_a, Z_a, n)$ with the results tabulated in Table 5.

TABLE 5

RESULTS OF THE MINIMIZATION OF EQN (25) WITH THE DATA SET BASED ON THE VALUES FROM TABLE 4

Parameter	*Starting values*	*Final values*	
T_a	$9.200000 \cdot 10^3$	$9.000004 \cdot 10^3$	K
Z_a	$4.500000 \cdot 10^{11}$	$5.000064 \cdot 10^{11}$	s^{-1}
n	1.000000	1.000001	—
S(par)	$1.224 \cdot 10^4$	$2.529 \cdot 10^{-11}$	K^2

We used for this minimization 27 integrations of eqns (2) and (9) together and zero integrations of eqn (2) alone. The calculations started with a λ of $3.359 \cdot 10^6$ and finished with a λ of $9.454 \cdot 10^{-4}$.

4. EXPERIMENTAL PART

The reaction calorimeter used for the present measurements is the one described in our previous article[2] (section 2.2). We started with two separated liquid mixtures. Sulphuric acid in the double-piston injector with a concentration of ten times the desired reaction concentration and a methyloxirane–water mixture in the covering dewar-calorimeter. After temperature equilibration, the two liquids are mixed and the temperature is written down as a function of time. The temperature is recorded digitally at equidistant times by means of a Hewlett-Packard quartz thermometer with a HP-2580-D sensor and punched on paper-tape with an TTY-SRT-33. Most measurements have been conducted with a resolution of 10^{-3}K. The temperatures are corrected for the non-linearity and the response delay of the sensors.

As mentioned in section 2.3 we need starting values for Q/c_p, Z_a and T_a. The first one is calculated using thermodynamic data[23–25]. Z_a and T_a are calculated using the methods outlined in our previous article[2] (section 4). The number of data points is between 150 and 200.

5. RESULTS AND DISCUSSION

Working with the minimization procedure we develop a feeling for the choice of the starting value of λ and its reducing factor. Our experience gives us the conviction that better selection criteria should be developed in estimating a reliable value of λ_0.

The suggestion of Davies and Whitting[17] to estimate λ from local information of the function used sounds very promising. We used therefore their estimation-function for λ in the Bellman-problem. We found a λ_{calc} of a factor of about 10^4

higher than the one we found to be optimal. Our impression is that for relatively ill-conditioned problems, like the ones of Bellman and our own, this equation does not fulfill the expectations.

In saving a lot of troubles scaling is a major job to do. Both the first problem and the last are scaled by taking the natural logarithms of the parameters. The equations in the first test problem alter in

$$\frac{dT}{dt} = -\exp(\mathrm{par}_3)\cdot\frac{dx}{dt} - \frac{hA}{c_w}(T - T_c) + \frac{W_s}{c_w} \tag{27}$$

$$-\frac{dx}{dt} = \exp(\mathrm{par}_2)\, x^n \exp(-\exp(\mathrm{par}_1)/T) \tag{28}$$

where

$\exp(\mathrm{par}_1) = T_s = E/R$

$\exp(\mathrm{par}_2) = Z_n$

$\exp(\mathrm{par}_3) = (Qm_0)/c_p$

The number of iterations given in section 3 are based on these scaling. We used these kinds of scaling because the unscaled problem converges very slow and does not reach the minimum after 100 iterations. The scaled problem converges to the known minimum as shown in 19 iterations. The same holds for the third test problem. In our previous article[2] different values are found for reactions with an initial concentration of 1 mol kg^{-1} in comparison with a concentration of 2 mol kg^{-1}. As shown[2] this discrepancy results mainly in the difference in reaction-time and thus the course of the heat transfer. In case eqn (23) is a good description of the behaviour of the reactor dynamics, there is no reason for different values for the 1 and 2 molar reaction runs. Most reaction equations are of the Arrhenius type but the Eyring type is even a useful description.

The reaction equations for the Arrhenius and Eyring type are, respectively:

$$r(m, T) = -\frac{dm}{dt} = Z_n\, m^n \exp\left(\frac{-T_s}{T}\right) \tag{29}$$

and

$$r(m, T) = -\frac{dm}{dt} = P\,T m^n \exp\left(\frac{-T_s}{T}\right)$$

The rate of change of the temperature is given for both equations by

$$\frac{dT}{dt} = \frac{Q}{c_p}\, r(m, T) - \frac{hA}{c_w}(T - T_c) + \frac{W_s}{c_w} \tag{30}$$

The reaction under consideration is the acid-catalysed hydrolysis of methyloxirane[2]. From two sets of each 15 experiments with an initial temperature of about 278 K, a pH of about 1.0 and initial reactant concentrations of about 1.0 and 2.0 mol kg^{-1}, respectively, the average values of T_a, Z_2, P, Q and k_2 are summarized in Table 6.

TABLE 6

EXPERIMENTAL KINETIC VALUES OF THE HYDROLYSIS OF METHYLOXIRANE (eqns 29 and 30)

*Parameter**	*Arrhenius equation* *1 mol kg^{-1}*	 *2 mol kg^{-1}*		*Eyring equation* *1 mol kg^{-1}*	 *2 mol kg^{-1}*
T_a	8877 ± 42	8827 ± 23	T_a	8587 ± 44	8540 ± 22
Z_2	$(7.0 \pm 1.3) \cdot 10^{11}$	$(6.9 \pm 1.2) \cdot 10^{11}$	P	$(1.04 \pm 0.17) \cdot 10^{9}$	$(9.1 \pm 1.5) \cdot 10^{8}$
Q	86.4 ± 4.0	88.8 ± 4.0	Q	86.3 ± 4.0	88.6 ± 4.0
k_2 (25°C)	0.082 ± 0.015	0.096 ± 0.017	k_2 (25°C)	0.097 ± 0.025	0.099 ± 0.016

* The dimensions of T_a, Z_2, P, Q and k_2 are K, kg mol^{-1} s^{-1}, kg mol^{-1} K^{-1} s^{-1}, kJ mol^{-1}, kg mol^{-1} s^{-1}, respectively.

The residual vectors of all calculations showed no satisfactory random pattern, but rather a periodical phenomenon. A further investigation of the experimental conditions gives the opportunity to distinguish the following facts:

(1) owing to change in chemical composition of the reaction mixture both the heat capacity and the viscosity will change;

(2) owing to the change in temperature both the heat of reaction, the heat capacity and the viscosity will change;

(3) owing to a number of varying surrounding conditions, like periodical changes in the temperature of the thermostatted bath, there are changing heat transfer conditions of the reactor.

The change in viscosity and mass density has an influence on the heat of stirring and the heat transfer. All these effects are connected to the following rate of temperature change

$$\frac{dT}{dt} = -\frac{Q}{c_p}(1 + BT)\,r(m, T) + \frac{W_s}{c_w} - \frac{hA}{c_w}(T - T_c) + \beta \sin(\omega t) \tag{31}$$

The term $(1 + BT)$ stands for the effects in Q and c_p and $\beta \sin(\omega t)$ for the effects in viscosity change, e.g., the change in the heat of stirring and the effect in heat transfer change to the fluctuating surroundings.

It turned out that the effect of the factor B is very small, its value is of $(3.35 \pm 0.01) \cdot 10^{-4}$ for all the experiments. The values for β and ω fluctuated from experiment to experiment but the values for β were of about $4 \cdot 10^{-3}$ to $7 \cdot 10^{-2}$ K s^{-1} (2 mol kg^{-1}) and of $1 \cdot 10^{-4}$ to $4 \cdot 10^{-4}$ K s^{-1} for the 1 mol kg^{-1} runs and for ω, for all experiments, between $2 \cdot 10^{-2}$ and $2 \cdot 10^{-4}$ s^{-1}.

The results of these calculations gave a fairly random pattern for the residual vectors. The values of the kinetic parameters are listed in Table 7.

TABLE 7

EXPERIMENTAL KINETIC VALUES OF THE HYDROLYSIS OF METHYLOXIRANE BASED ON EQN (31)

Parameter*	Arrhenius equation			Eyring equation	
	1 mol kg^{-1}	*2 mol kg^{-1}*		*1 mol kg^{-1}*	*2 mol kg^{-1}*
T_a	8826 ±42	8804 ±18	T_a	8556 ±29	8520 ±15
Z_2	(5.97 ±0.90) · 10^{11}	(5.6 ±1.0) · 10^{11}	P	(8.5 ±1.1) · 10^{8}	(7.3 ±1.3) · 10^{8}
Q	84.7 ±4.0	89.8 ±4.0	Q	84.9 ±4.0	90.1 ±4.0
k_2 (25°C)	0.083±0.012	0.084±0.015	k_2 (25°C)	0.087±0.011	0.085±0.015

* The dimensions are mentioned below Table 6.

The S(par)-values of the calculations in Table 7 based on eqn (31) are in comparison with those of Table 6 a factor 50 to 1000 lower. The values of S(par) of calculations on corresponding experiments of the Arrhenius and Eyring equation gave no significant difference. Putting all experiments together we find the values for the acid catalysed hydrolysis of methyloxirane mentioned in Table 8.

TABLE 8

Parameter*	Arrhenius		Eyring
T_a	8803 ± 28	T_a	8534 ± 23
Z_2	(5.9 ± 1.4) · 10^{11}	P	(8.0 ± 1.6) · 10^{8}
Q	88.4 ± 2.8	Q	88.3 ± 2.7
k_2 (25°C)	0.089 ± 0.020	k_2 (25°C)	0.088 ± 0.017

* The dimensions are mentioned below Table 6.

The advantage of using the Eyring type equation is the possibility of calculating the activation entropy of reaction by the relation

$$P = \frac{k_B}{h} \exp\left(\frac{\Delta S}{R}\right) \tag{32}$$

where $k_B = 1.380662 \cdot 10^{-23}$ J K^{-1}
$h = 6.626176 \cdot 10^{-34}$ J s^{-1}
$R = 8.31441$ J mol^{-1} K^{-1}

k_B and h are the constants of Boltzmann and Planck, respectively. Using for P the value from Table 8 we find an activation entropy of

$\Delta S = -27.1 \pm 1.5$ J mol^{-1} K^{-1}

Although the method is very computer-time consuming the results are fairly good compared with the literature. Long et al.[26] gave the following results (0.05 to 0.2 molair)

T_a (Arrhenius) = 9392 $\quad Z_2 = 2.12 \cdot 10^{12}$
T_a (Eyring) = 9100 $\quad P = 2.66 \cdot 10^{9}$
k (25°C) = 0.044 $\quad \Delta S = -17.1$

The dimensions are mentioned below Table 6.

ACKNOWLEDGEMENTS

The present investigations are conducted with financial support from the Netherlands Organization for the Advancement of Pure Research (Z.W.O., The Hague).

The author wishes to thank Drs. W. Hoffmann for the encouraging discussions during the development of the computer program.

NOMENCLATURE

A	heat transfer surface area
c_p	average specific heat capacity
c_w	heat capacity
E	energy of activation
h	overall heat transfer coefficient
k_m	pseudo reaction rate constant
m	molalility of reaction solution
m_0	initial molalility of reaction solution
n	reaction order
P	frequency factor in the Eyring equation
Q	heat of reaction (positive for exothermic reactions)
R	gas constant
$r(m,T)$	reaction rate
ΔS	reaction activation entropy
t	time
T	absolute temperature of reacting mixture
T_0	initial temperature of reacting mixture
T_a	activation energy temperature
T_c	ambient temperature
T_m	maximum adiabatic reaction temperature
W_s	heat of stirring
Z	frequency factor in the Arrhenius equation
Z_m	modified frequency factor

APPENDIX I

The procedure described in section 2 has the following heading with the meaning of the parameters given after that. The procedure has the value of $S(\text{par})$ at the end of the calculations.

```
real procedure parestinde (npar, ndata, neqn, cobsi, in, realin, obs, par, partest,
                           callystart, calldfdpar, callderiv, calljac, stiff, residu, invers, jac, out,
                           outdata);
value npar, ndata, neqn, cobsi, in, realin, partest, obs;
integer npar, ndata, neqn, cobsi;
array in, realin, partest, out, obs, par, residu, invers, jac;
boolean stiff;
boolean procedure calljac;
procedure outdata, calldfdpar, callderiv, callystart;
```

The meaning of the formal parameters is

npar	⟨arithmetic expression⟩;	the number of unknown parameters;
ndata	⟨arithmetic expression⟩;	the number of observations; ndata > npar;
neqn	⟨arithmetic expression⟩;	the number of differential equations;
cobsi	⟨arithmetic expression⟩;	the component of y observed ($1 \leq \text{cobsi} \leq \text{neqn}$);
in	⟨array identifier⟩	; array in [0:7]; entry: in this array data should be given to control the process: in[0]: machine precision; in[1]: the relative tolerance for the euclidian norm of the residual vector; in[2]: the absolute tolerance of the euclidian norm of the residual vector; in[3]: the maximum number of jacobian evaluations; in[4]: the factor alfa to calculate the starting value of lambda, a suitable value is 10^{-4}; in[5]: the reducing factor beta to reduce lambda, a suitable value is 0.5; in[6]: on the way change of integration precision if $\lVert\text{residue}\rVert < \text{in}[6]$;
realin	⟨array identifier⟩	; array realin [1:6]; entry: in this array control data for the integration procedure should be given; realin[1]: starting value of the time; realin[2]: final value of the time; realin[3]: minimal integration step, a suitable value is $\text{realin}[4] * 10^{-4}$;

realin[4]: maximal integration step, a suitable value is the time step between two observations;
realin[5]: the initial relative local error bound for the integration, a suitable value is in[1];
realin[6]: if $\|residue\| < in[6]$ then the relative local error bound is equal to realin[6], a suitable value is $in[1]*10^{-2}$;

obs <array identifier>; array obs[1:ndata, 1:neqn + 1];
entry: contains the experimental observations;
obs[i,j]: is equivalent with y[j];
obs[i,neqn + 1]: is equivalent with time;

par <array identifier>; array par[1:npar];
entry: starting values of the parameters;
exit: the calculated parameters;

partest <array identifier>; array partest[1:npar, 1:3];
entry: information about boundaries for the parameters should be given;
partest[i,1]: lower bound of parameter i;
partest[i,2]: new value of parameter i when boundary is crossed;
partest[i,3]: upper bound of parameter i;

callystart <procedure identifier>; procedure callystart (y, ynp, ymax);
array y, ynp, ymax;
exit: $y[1:6\times neqn]$, y[1] to y[neqn] contains the starting values of y;
$ynp[0:7, 1:neqn\times npar]$, ynp[0, 1] to $ynp[0, neqn\times npar]$ contains the starting values for the integration of dyp/dx;
ymax[1:neqn], ymax[i] should contain an estimate of the absolute maximal value of y[i] over the integration interval;

calldfdpar <procedure identifier>; procedure calldfdpar (dfdpar, par y, x);
array dfdpar, par, y; real x;
entry: par[1:npar] contains the current values of the parameters;
y[1:neqn] contains the solution of the differential equation at time x;
exit: array dfdpar[1:neqn, 1:npar], this array contains the values dfdpar[i,j] := d(dy[i]/dx)/d(par[j]);

callderiv	⟨procedure identifier⟩	; procedure callderiv (df, par, y, x); array df, par, y; real x; entry: see calldfdpar; exit: array df[1:neqn]; df[i]:= dy[i]/dx;
calljac	⟨procedure identifier⟩	; boolean procedure calljac (jacobian, par, y, x); array jacobian, par, y; real x; entry: see calldfdpar; exit: array jacobian [1:neqn, 1:neqn]; jacobian[i,j]:= d(dy[i]/dx)/dy[j]; the boolean procedure should deliver the value true after a successful call of calljac;
stiff	⟨boolean identifier⟩	; when we are dealing with stiff differential equation stiff should be true;
residu	⟨array identifier⟩	; array residue[1:ndata]; exit: residu[1:ndata] contains the residue vector at the calculated minimum. residu[i]:= observed y[i] − calculated y[i];
invers	⟨array identifier⟩	; array invers[1:npar, 1:npar]; exit: the invers of J^*J is stored in invers, this matrix is necessary to calculate the covariance and correlation matrices;
jac	⟨array identifier⟩	; array jac[1:ndata, 1:npar]; exit: the matrix jac contains the jacobian in the calculated minimum, jac[i,j]:= d(rv[i])/dpar[j]; should be used by calculating the value of the gradient in the minimum;
out	⟨array identifier⟩	array out [1:9]: divers by-products of the calculations are stored in the array out; out[1]: the condition number of J^*J; out[2]: the euclidian norm of the residual vector calculated in the minimum; out[3]: the last improvement of the residual vector; out[4]: the euclidian norm with the starting values of the parameters; out[5]: the number of calls of procedure funct (eqn 2 alone); out[6]: the number of calls of the procedure jacobian (eqn 2 and 9); out[7] = 0 normal termination;

= 1 number of iterations to perform exceed in[3];

= 2 the call of jacobian delivers the value false;

= 3 the first call of jacobian delivers the value false;

= 4 the precision asked for cannot be attained, reduce in[2];

out[8] = 1 a call of funct delivers the value false;

out[9] = 1 a call of the integration procedure delivers the value false;

outdata ⟨procedure identifier⟩ : procedure outdata (pointer, integer, real 1, real 2, array);
integer pointer, integer; real real 1, real 2;
array array;

exit: this procedure can be used to obtain information during the integration. There are four places where output is activated;

pointer = 1 after a successful call of funct integer gives the number of calls. real 1 gives the used lambda, real 2 gives the value of S(par), array gives the used parameter array;

pointer = 2 after a successful call of jacobian integer, real 1, real 2, array: see pointer = 1;

pointer = 3 after a parameter bound jumps the rejected value and its substitute integer is the number i of par[i] which is rejected, real 1 is the rejected value, real 2 is its substitute, array is the parameter array;

pointer = 4 after a failure in the integration procedure the following information is available, integer, real 1 and real 2 are zero, array contains the following;

array[− 1] = 0 no error messages available;

array[− 1] = 1 minimal step length (realin[3]) too great to handle the non-linearity;

array[− 2]: number of local error bound exceedings;

array[− 3]: if array[− 2] ≠ 0 this gives an estimate of the maximal local error;

REFERENCES

1 W. R. Dammers, W. Frankvoort and M. Tels, *Thermochim. Acta*, 3 (1971) 133.
2 W. Frankvoort and W. R. Dammers, *Thermochim. Acta*, 11 (1975) 5.
3 P. W. Hemker, *Parameter estimation in non-linear differential equations*, Mathematical Centre report, MR 134/72, Amsterdam, 1972.
4 R. Szuniewicz and A. Manitius, *Chem. Eng. Sci.*, 29 (1974) 1701.
5 M. Hwang and J. H. Seinfeld, *A.I.Ch.E.J.*, 18 (1972) 90.
6 G. M. Willems, ch. 5 in *Colloquium stijve differentiaal vergelijkingen*, MCS-15-2, in Dutch, Mathematisch Centrum, Amsterdam, 1973.
7 P. W. Hemker, ch. 4 in *Colloquium stijve differentiaal vergelijkingen*, MCS-15-1, in Dutch, Mathematisch Centrum, Amsterdam, 1972.
8 W. E. Ball and L. C. D. Groenewegne, *Ind. Engg. Chem. Fund.*, 5 (1966) 181.
9 F. O. Gerdes, P. J. Hoftijzer, J. F. Kemkes, M. van Loon and C. Schweigman, *Chem. Eng.*, 241 (1970) 267.
10 N. Draper and H. Smith, *Applied Regression Analysis* Wiley, New York, 1966.
11 P. W. Hemker (Ed.), Numal, *A library of numerical procedures in algol-60, index and kwic index*, Mathematisch Centrum, Amsterdam, 1974.
12 P. W. Hemker, *An algol-60 procedure for the solution of stiff differential equations*, Mathematical Centre report, MR 128/71, Amsterdam, 1971.
13 G. H. Golub and M. A. Saunders, in J. Abadie (Ed.), *Linear least squares and quadratic programming, Integer and Nonlinear Programming*, North-Holland, Amsterdam, 1970.
14 D. W. Marquardt, *J. Soc. Ind. Appl. Math.*, 11 (1963) 431.
15 R. R. Meyer and P. M. Roth, *J. Inst. Maths. Appl.*, 9 (1972) 218.
16 W. Murray, *Numerical Methods for Unconstrained Optimization*, Academic Press, 1972.
17 M. Davies and I. J. Whitting, *A modified form of Levenberg's correction*, in *Numerical methods for non-linear optimization*, Academic Press, 1972.
18 A. W. Matz, *Appl. Stat. J.R. Stat. Soc.*, 13 (1964) 118.
19 F. B. Smith and D. F. Shanno, *Technometrics*, 13 (1971) 63.
20 K. Levenberg, *Quart. Appl. Math.*, 2 (1944) 164.
21 R. Bellman, J. Jacquez, R. Kabala and S. Schwimmer, *Math. Biosci.*, 1 (1967) 71.
22 M. Bodenstein, *Z. Phys. Chem.*, 100 (1922) 68.
23 Landolt-Börnstein, *Physikalisch-Chemische Tabellen*, Springer, Heidelberg, 1936, HW 1586, Eg I 866.
24 P. W. Zubow and W. Swietoslowski, *Bull. Soc. Chim.*, 37 (1925) 271.
25 *Selected Values of Chemical Thermodynamic Properties*, circ. 500, Nat. Bur. Stand., Washington, D.C., 1952.
26 F. A. Long, J. G. Pritchard and F. E. Stafford, *J. Am. Chem. Soc.*, 79 (1957) 2362.