DNP WBScott

2783

Two – Compartment Model of Uptake and Excretion of Chemicals

V. Zitko

Biological Station,
St. Andrews, N.B., E0G 2XO

January 1982

Canadian Technical Report of Fisheries and Aquatic Sciences No. 1076

Government of Canada Fisheries and Oceans
Gouvernement du Canada Pêches et Océans

Canadian Technical Report of
Fisheries and Aquatic Sciences 1076

January 1982

TWO-COMPARTMENT MODEL OF UPTAKE AND EXCRETION OF CHEMICALS

by

V. Zitko

Fisheries and Environmental Sciences
Department of Fisheries and Oceans
Biological Station
St. Andrews, New Brunswick EOG 2X0
Canada

This is the one hundred and forty-sixth Technical Report from
the Biological Station, St. Andrews, N.B.

Cat. No. Fs 97-6/1076 ISSN 0706-6457

Correct citation for this publication:

Zitko, V. 1982. Two-compartment model of uptake and excretion of chemicals. Can. Tech. Rep. Fish. Aquat. Sci. 1076: iii +27 p.

ABSTRACT

Zitko, V. 1982. Two-compartment model of uptake and excretion of chemicals. Can. Tech. Rep. Fish. Aquat. Sci. 1076, iii + 27 p.

Determination of uptake and excretion rate constants in the two-compartment model (TCM) is described. The Nelder-Mead simplex method is used to minimize the sum of squared deviations (SUSQ) by fitting rate constants to the differential equations. SUSQ surfaces of TCM and of other models used in aquatic toxicology are discussed.

Key words: Compartmental analysis, bioconcentration, programs BASIC

RÉSUMÉ

Zitko, V. 1982. Two-compartment model of uptake and excretion of chemicals. Can. Tech. Rep. Fish. Aquat. Sci. 1076, iii + 27 p.

On trouvera dans le rapport qui suit la description d'une méthode de détermination des constantes de taux d'absorption et d'excrétion dans un modèle ã deux compartiments (TCM). Le modèle simplex de Nelder-Mead est utilisé pour minimiser la somme des carrés des déviations (SUSQ) en adaptant les constantes de taux aux équations différentielles. On y analyse les surfaces des SUSQ du TCM et autres modèles utilisés en toxicologie aquatique.

INTRODUCTION

The majority of data on uptake and excretion of chemicals by aquatic fauna are fitted to the one-compartment model (OCM) (Fig. 1) (see for example Zitko 1980).

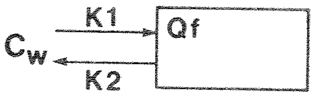


Fig. 1. One-compartment model: K1, K2 = uptake and excretion rate constants, respectively.

OCM is a considerable oversimplification, but it usually describes the data quite well. The rate constants are interpretable easily in terms of the bioconcentration factor (BCF = K1/K2) and excretion half-life ($t_{\frac{1}{2}}$ = ln 2/K2). High variance caused by biological variability, small number of tissues analyzed, and a small number of experimental points during uptake and excretion often do not warrant a more detailed examination. The differential equation describing the OCM is

$$dQ_f/dt = K1C_w - K2Q_f \qquad (1)$$

where Q_f = quantity (or concentration) of chemical in fish
C_w = concentration of chemical in water
t = time
K1,K2 = uptake and excretion rate constant, respectively

Integration of (1) for C_w = constant and for no chemical present in the fish initially ($Q_f = 0$ at $t = 0$) yields

$$Q_f = C_w(K1/K2)(1 - Exp(-K2t)). \qquad (2)$$

The equilibrium bioconcentration factor (BCF) is

$$BCF = Q_f/C_w = K1/K2 \qquad (3)$$

This relationship can be obtained from (1) as well since, at equilibrium, $dQ_f/dt = 0$. It should be noted that in (3) Q_f stands for concentration, not amount of chemical in the fish. Several techniques for fitting (2) to experimental data have been described (see for example Zitko 1980). During the excretion phase $C_w = 0$ and integration of (1) for $Q_f = Q$ at $t = 0$ gives

$$Q_f = Q\ Exp(-K2t). \qquad (4)$$

Equation (4) contains a single exponential term and becomes linear when plotted in semilogarithmic coordinates ($\log Q_f$ vs t).

There are cases in which the OCM cannot provide an adequate description of the data. The inadequacy of the OCM becomes particularly obvious for the excretion phase of the experiment with the data deviating systematically from a straight line in semilogarithmic coordinates. In such situations, the two-compartment model (TCM) should be used.

The differential equations describing the TCM (Fig. 2) are

$$\begin{aligned} dQ1/dt &= K1C_w + K4Q2 - (K2 + K3)Q1 \\ dQ2/dt &= K3Q1 - K4Q2 \end{aligned} \qquad (5)$$

where Q1,Q2 = quantity of chemical in compartments 1 and 2, respectively;
C_w = concentration of chemical in water;
K1,K3 = uptake rate constants for compartments 1 and 2, respectively;
K2,K4 = excretion rate constants for compartments 1 and 2, respectively;
t = time.

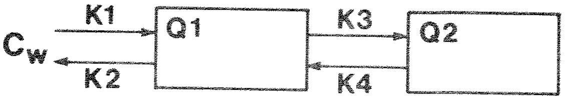


Fig. 2. Two-compartment model:
W1,Q1,C1,W2,Q2,C2 = weight of compartment, quantity, and concentration of chemical for compartments 1 and 2, respectively;
K1,K3 = uptake rate constants for compartments 1 and 2, respectively;
K2,K4 = excretion rate constants for compartments 1 and 2, respectively;
C_w = concentration of chemical in water.

The integration of (5) is straightforward but quite cumbersome (see for example Könemann and van Leeuwen 1980). Another method is given later in this report. Some relationships between the rate constants and quantities of chemical in the compartments can be derived readily from (5). At equilibrium (dQ1/dt=0, dQ2/dt=0)

$$Q2/Q1 = K3/K4 \qquad (6)$$

$$Q1 = (K1/K2)C_w \qquad (7)$$

$$Q2 = (K1/K2)C_w(K3/K4) \qquad (8)$$

In other words, the equilibrium BCF in compartment 1 is the same as for the OCM, and the equilibrium BCF in compartment 2 is the BCF in compartment 1 multiplied by K3/K4. Even more generally, the equilibrium BCF of a compartment is a ratio of its uptake and excretion rate constants.

It should be noted that in (5)-(8), Q's mean quantities, not concentrations. To obtain concentrations, the quantities must be divided by the respective compartment weights. In addition, data for the two compartments taken individually may not be available. In such cases the TCM can be used to fit the total concentration CT.

$$CT = (Q1 + Q2)/(W1 + W2) \qquad (9)$$

The BCF then is

$$BCF = CT/C_W = (Q1 + Q2)/(W1 + W2)(1/C_W) = (K1/K2)(1 + K3/K4)/(W1 + W2) \qquad (10)$$

For excretion, TCM does not have a simple relationship between the rate constants and the half-life of the chemical, in contrast to the OCM. According to the TCM, excretion is described by an equation consisting of *the sum of two exponential* terms, and the half-life increases with decreasing concentration of the chemical.

SOLUTION OF THE TCM DIFFERENTIAL EQUATIONS

Equation (3) can be solved readily by the procedure described for systems of linear differential equations with constant coefficients (see for example Bronson 1973). To use this procedure, a third equation, describing constant concentration in water, must be added ($dC_W/dt = 0$). *Equation (5)* then becomes

$$\begin{aligned} dQ1/dt &= -(K2 + K3)Q1 + K4Q2 + K1C_W \\ dQ2/dt &= K3Q1 - K4Q2 \\ dC_W/dt &= 0 \end{aligned} \qquad (11)$$

or, in matrix notation

$$d'Q'/dt = 'k''Q' \qquad (12)$$

where ' ' denotes a matrix.

$$'Q' = \begin{pmatrix} Q1 \\ Q2 \\ C_W \end{pmatrix}, \quad 'K' = \begin{pmatrix} -(K2+K3) & K4 & K1 \\ K3 & -K4 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

To solve (11), the eigenvalues and eigenvectors of 'K' must be calculated. The eigenvalues E are the roots of the characteristic polynomial of 'K':

$$\begin{aligned} E1,2 &= 0.5(-(K2+K3+K4) \pm SQR((K2+K3+K4)^2 - 4K2K4) \\ E3 &= 0 \end{aligned} \qquad (13)$$

Eigenvectors 'V' are obtained by solving the system of equation *(12) for each eigenvalue:*

$$\begin{pmatrix} -(K2+K3)-E & K4 & K1 \\ K3 & -K4-E & 0 \\ 0 & 0 & -E \end{pmatrix} \begin{pmatrix} V1 \\ V2 \\ V3 \end{pmatrix} = 0 \qquad (14)$$

Thus, for example, for eigenvalue E1:

$$\begin{aligned} (-(K2+K3)-E1)V1 + K4V2 + K1V3 &= 0 \\ K3V1 - (K4+E1)V2 &= 0 \\ -E1V3 &= 0 \end{aligned}$$

$$V3 = 0$$

For $V2 = 1$

$$V1 = K4/(K2+K3+E1)$$

The complete eigenvector matrix 'V' (eigenvectors as columns) is

$$'V' = \begin{pmatrix} K4/(K2+K3+E1) & K4/(K2+K3+E2) & K1/K2 \\ 1 & 1 & K1K3/(K2K4) \\ 0 & 0 & 1 \end{pmatrix}$$

The general solution of (11) then is

$$'Q' = \ 'V' \begin{pmatrix} A1ExpE1t \\ A2ExpE2t \\ A3 \end{pmatrix} \qquad (15)$$

A1, A2, and A3 are constants. For a particular solution, their values are determined from the initial (t=0) values of Q1, Q2, and C_W by solving the system of equation (16).

$$\begin{pmatrix} Q1 \\ Q2 \\ C_W \end{pmatrix}_{t=0} = \ 'Q'_{t=0} = \ 'V' \begin{pmatrix} A1 \\ A2 \\ A3 \end{pmatrix} = \ 'V' \ 'A' \qquad (16)$$

The solution is

$$'A' = \ 'V'^{-1} \ 'Q'_{t=0} \qquad (17)$$

where $'V'^{-1}$ = inverted matrix 'V'.

For the excretion phase, the differential equations of the TCM are

$$\begin{aligned} dQ1/dt &= -(K2 + K3)Q1 + K4Q2 \\ dQ2/dt &= K3Q1 - K4Q2 \end{aligned} \qquad (18)$$

and the system may be solved as described. It should be noted that the eigenvalues E1 and E2 are the same as in the uptake phase. Consequently, the exponential terms during the uptake and during the excretion are the same.

The outlined calculations are *ideally suited* for a computer and listing of a BASIC program is given in the Appendix. The disadvantage of this approach is that it is not immediately obvious how the values of the individual rate constants affect the solution. However, considering the unwieldiness of the explicit analytical solution this is a *relatively small price to pay* and the relationships (6)-(8) give a general idea about the effects of the rate constants.

A "hidden" problem lies in the inversion of the matrix 'V'. If the matrix is "ill-conditioned" in respect to inversion, small changes in the values of the elements of 'V' may have a very pronounced erratic effect on the solution. The problem is analogous to the one encountered in finding the intercept of two almost parallel straight lines. *Matrices of systems with large K1 and K3, and small* K2 and K4 tend to be "ill-conditioned."

FITTING TCM TO EXPERIMENTAL DATA

The described method for solving the TCM differential equations provides an implicit solution in terms of the rate constants. For example, for K1=10, K2=3, K3=2, and K4=1, and for initial concentrations $Q1_o=0$, $Q2_o=0$, and $C_W=1$, the solution is

$$\begin{aligned} Q1 &= -1.67Exp(-5.45t) - 1.67Exp(-0.55t) + 3.33 \\ Q2 &= \ \ 0.75Exp(-5.45t) - 7.41Exp(-0.55t) + 6.67 \\ C_W &= 1 \end{aligned} \qquad (19)$$

The solution does not indicate explicitly how the rate constants and the initial concentrations enter into the constants and exponents of (19). Consequently, the easiest approach to fit TCM to experimental data is to fit the rate constants in the differential equations. To fit the coefficients and exponents of the solved differential equations would require an explicit solution to be available.

Fitting coefficients and exponents of a general expression (20) to the data is not acceptable

$$Q = AExp(-Bt) + CExp(-Dt) + E \qquad (20)$$

since it neglects the relationships between the values A, B, C, D, and E existing in the solution of the TCM differential equations. The fit would be very good because seven adjustable parameters can fit an elephant (actually, 30 parameters are needed (Wei 1975)).

The easiest to adopt fitting criterion is the least squares, minimizing the sum of squared deviations (SUSQ) between the data and values calculated from TCM. Several methods are available for the SUSQ minimization (see for example Metzler 1980). The simplex method described below was selected because of its simplicity and easy adaptability to a wide variety of other problems.

SIMPLEX MINIMIZATION

The simplex minimization can be used to fit equations containing any number of parameters to experimental data (for example OCM contains two parameters, K1, K2; TCM is based on four parameters, K1-K4). A good description of the simplex technique is available (Deming and Morgan 1973). From the point of mathematics, the technique is extremely simple and is best illustrated by considering the fit of two parameters, K1 and K2 (Fig. 3).

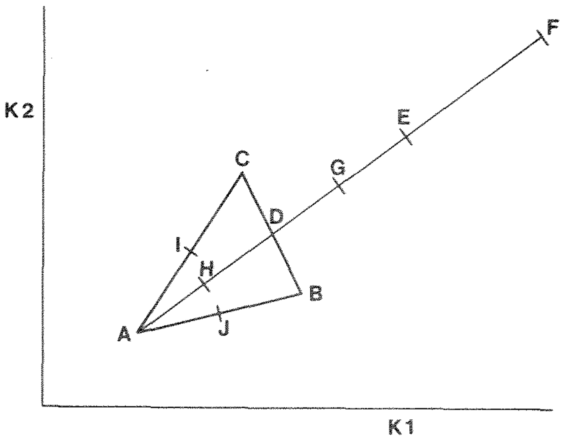


Fig. 3. Two-dimensional simplex (see text for explanation).

For two parameters (two-dimensional simplex), the "response" is determined at three points (A,B,C, Fig. 3). The "response" is the SUSQ between experimental data and values calculated from the fitted function for the given pair of parameters K1 and K2. Since the objective is to minimize the SUSQ, the larger the SUSQ value, the worse the response. If the worst response among A, B, and C was obtained at point A, the next point to look at is E. This point is the reflection of A across the line BC. If the response at E is better than those at B and C, the next point to look at is F (we are going in the right direction!). Point F is a reflection of D through E, and is called expansion in terms of the original simplex ABC. On the other hand, if the response at E is between the responses at B and at C, the new simplex is BCE. If the response at E is worse than those at B and C, but still better than that at A, the new simplex will be BCG. This move is called contraction. However, if in the last case the response is worse than that at A, the new simplex is BCH. If the response at either G or H is still worse than that at A, the new simplex will be CDI, if the response at C is better than at B; if the response is better at B, the new simplex will be DBJ.

It will become obvious after several readings that the algorithm is very simple. The coordinates of the points (values of K1 and K2) are easy to calculate. If $K1_B$, $K2_B$ and $K1_C$ and $K2_C$ are the coordinates of B and C, respectively, then the coordinates of D are

$$\begin{aligned} K1_D &= 0.5(K1_B + K1_C) \\ K2_D &= 0.5(K2_B + K2_C) \end{aligned} \qquad (21)$$

The formula may be simplified further by introducing vector notation. If B* and C* are the coordinate vectors of points B and C, respectively, that is $B^* = (K1_B, K2_B)$ and $C^* = (K1_C, K2_C)$, then

$$D^* = 0.5(B^* + C^*)$$

(In vector addition the corresponding components are added, that is, K1's are added to form the K1 of the sum.) The following formulae give the coordinates of the remaining points in Fig. 3:

$$\begin{aligned} E^* &= D^* + (D^* - A^*) \\ F^* &= D^* + 2(D^* - A^*) \\ G^* &= D^* + 0.5(D^* - A^*) \\ H^* &= D^* - 0.5(D^* - A^*) \\ I^* &= 0.5(A^* + C^*) \\ J^* &= 0.5(A^* + B^*) \end{aligned} \qquad (22)$$

To illustrate the simplex technique further, let us fit the OCM excretion function

$$C_f = C_o Exp(-K2t)$$

to "experimental data" C_f = 1, 0.606, and 0.368 at t=0, 1, and 2, respectively. The fitted parameters are C_o and K2. The first step is to choose the initial simplex consisting of three (C_o, K2) pairs, for example (1.5, 1.5), (1.4, 1.7), and (1.2, 1.5). The next step is to calculate the SUSQ's for these three points. The procedure is outlined below for the point (1.5, 1.5).

t	Experimental data C_f	Values in (1.5,1.5) C_f	Difference	Difference^2
0	1	1.5	-0.5	0.25
1	0.606	0.335	0.271	0.073
2	0.368	0.075	0.293	0.086
			SUSQ =	0.409

Values of C_f in (1.5,1.5) are obtained from the fitted equation by substituting 1.5 for C_o and 1.5 for K2. The SUSQ values for the remaining two points of the initial simplex are calculated in the same way. It turns out that the worst response is in the point (1.5,1.5). New simplex is formed applying the rules described above and the process is continued until a reasonable fit is obtained. The progress of the fit is outlined in Fig. 4. It can be seen that the simplex procedure leads fairly rapidly to the "best fit" (1.0,0.5) used to generate the "experimental data."

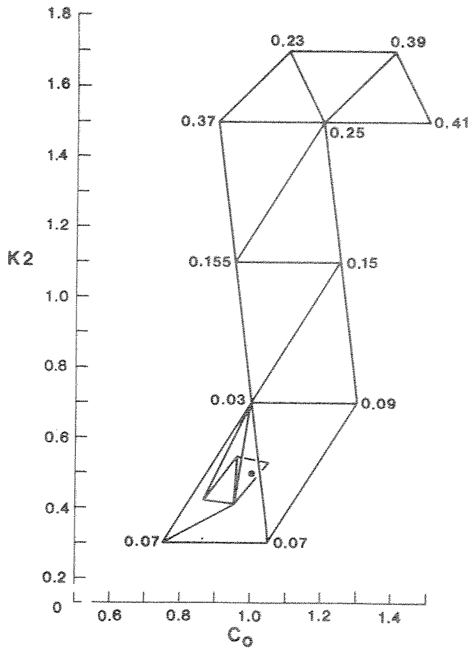


Fig. 4. Progress of simplex fitting the function $C_f = C_o\text{Exp}(-K2t)$; see text.

The procedure is usually terminated when the variance of the three SUSQ values of a simplex is less than a specified value. The procedure is by no means foolproof and fail-safe. It may lock on and converge to a local minimum or it may attempt to construct a simplex outside the field of definition of the fitted function (for example negative x in y = logx).

It is good practice to have some idea about the shape of the SUSQ surface in (as the mathematicians call it) the fitted parameter space (three-dimensional Fig. 4 with the SUSQ's perpendicular).

The simplex algorithm is ideally suited for automation and the listing of a BASIC program is given in the Appendix.

SUSQ SURFACES

SUSQ surface is a function of parameters of the fitted equation and, implicitly, of the values (range) of the independent variable. It is useful, particularly for simplex fits, to visualize the shape of the SUSQ surface.

Taking a straight line Y = K1X + K2 as an example, the SUSQ surface is a function of K1 and K2 and describes the SUSQ values obtained when the values K1 and K2 deviate from the "true" or "best fit" values, for example K1=5 and K2=30 (Fig. 5). The SUSQ values are calculated as described under Simplex Minimization. The surface has, as one would expect, a minimum for K1=5 and K2=30, located in the origin of the coordinates in Fig. 5. SUSQ values increase in all directions from the origin, but in some directions the "slope" is much steeper than in others (see Fig. 6 for a "spike" presentation of the same surface). The slope is very steep from the origin to the upper left corner and to the lower right corner. A "valley" extends from the lower left to the upper right corner.

Simplex seeks the steepest slope and, in this case, would be approaching the origin (finding the best fit) from the upper left or the lower right, depending on the selected initial position. The approach along the "valley" will be slower and more sensitive to experimental errors.

The direction of the "valley" depends on the relative contributions of the terms containing K1 and K2 to the value of Y. This is where the range of values of the independent variable X has an effect (the values used in Fig. 5 and 6 were X=0-10). In this case the values of K1X and K2 are roughly comparable. If, on the other hand, the "true" value of K2 was 900, the "valley" will extend in the direction of the K1 axis (Fig. 7), since deviations in K1 have relatively little effect on the value of Y if the range of X values is 0-10. Similarly, for K2=3, the "valley" will extend along the K2 axis (Fig. 8).

For a single-term exponential function Y = K1 Exp(-K2X) with K1 $>$ K2 $>$ 0, as in the case during excretion, the "valley" tends to extend in the upper left to lower right direction (Fig. 9). For the uptake phase of the corresponding OCM, Y = K1(1-Exp(-K2X))/K2, K1 $>$ K2 $>$ 0 and, depending on the ratio of K1 and K2, the "valley" shifts from the diagonal direction until it parallels the K2 axis for very low values of K2 relative to K1 (Fig. 10-12).

The simplex tends to perform well on these surfaces, provided the starting values of K1 and K2 are chosen reasonably well. The performance is generally better when the starting values are on a steeper slope rather than in the "valley." Consequently, according to Fig. 9-11, one constant should be underestimated and the other overestimated when selecting the starting values. Uptake with a very small K2 (Fig. 12) is difficult to fit to the OCM uptake equation. In this case the concentration (Y) increases practically linearly and the SUSQ is almost independent of the value of K2.

TCM is described by four parameters (K1-K4) and the corresponding SUSQ is a surface in five dimensions. Consequently, only a three-dimensional section can be visualized (Fig. 13-18).

In the first example (Fig. 13, 14) a wide "valley" lies from lower left to upper right, when the remaining two parameters have their "true values." The shape of the section changes considerably when the remaining parameters have different values (Fig. 15, 16). Even in this case there are relatively flat areas on the subsurface. The shape of a subsurface obtained by varying the second pair of parameters while maintaining the first pair at "true" values is similar (Fig. 17, 18).

"Valleys" and flat subsurface areas cause problems in fitting models to experimental data. In these areas, experimental and computational errors (whichever come first, usually the experimental ones) will have a very pronounced effect on the fit.

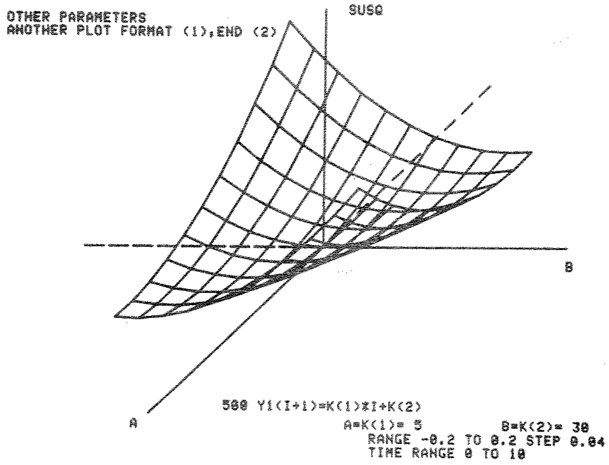


Fig. 5. SUSQ surface of straight line:

Y = K1X + K2;

range is deviation in fractions of the given values (± 20% in this case). Time range is the range of X values.

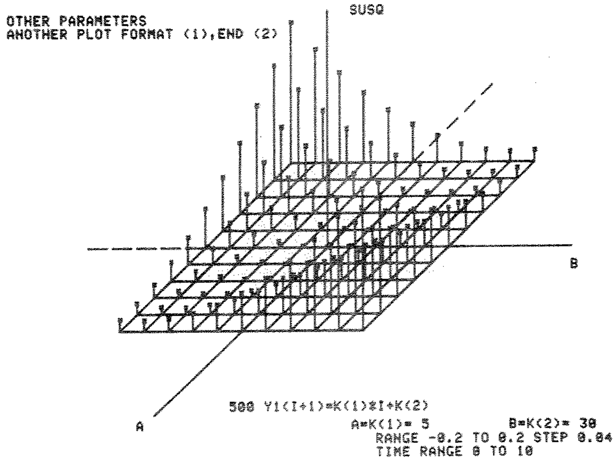


Fig. 6. SUSQ surface of straight line:

Y = K1X + K2;

"spike" presentation of the surface in Fig. 5.

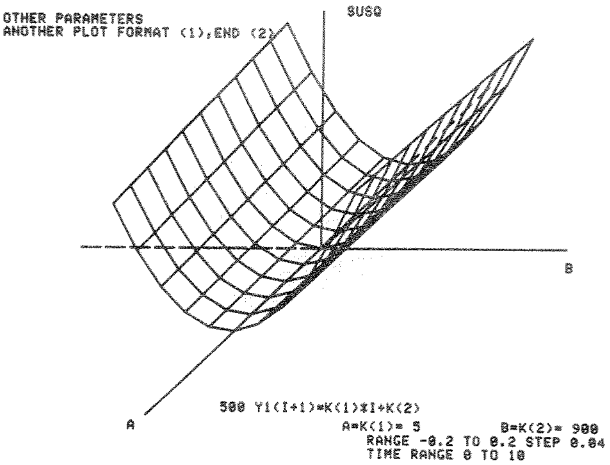


Fig. 7. SUSQ surface of straight line:

Y = K1X + K2 when K2 > K1.

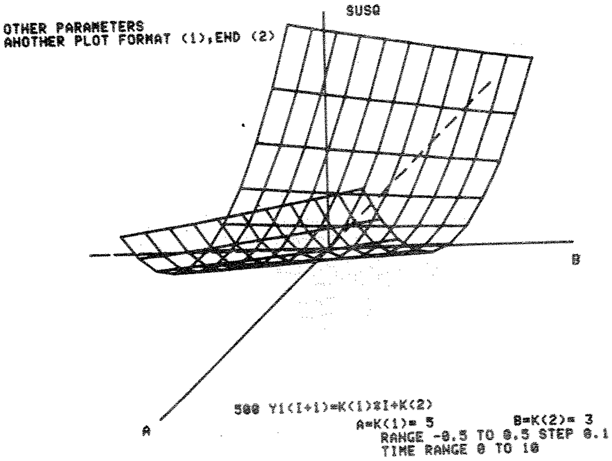
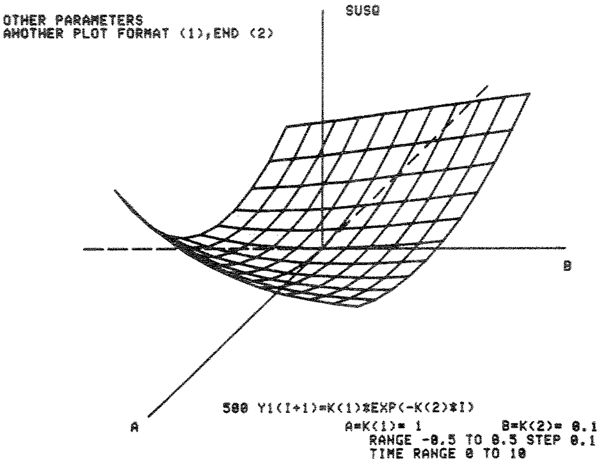


Fig. 8. SUSQ surface of straight line:

Y = K1X + K2 when K1 > K2.

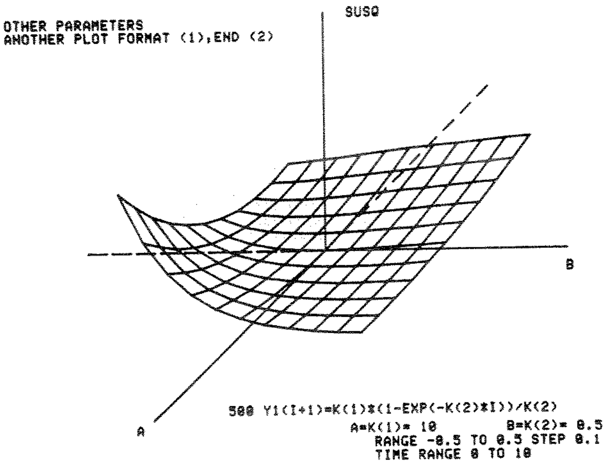


```
OTHER PARAMETERS
ANOTHER PLOT FORMAT (1),END (2)

500 Y1(I+1)=K(1)*EXP(-K(2)*I)
                    A=K(1)= 1           B=K(2)= 0.1
                      RANGE -0.5 TO 0.5 STEP 0.1
                      TIME RANGE 0 TO 10
```

Fig. 9. SUSQ surface of exponential function:

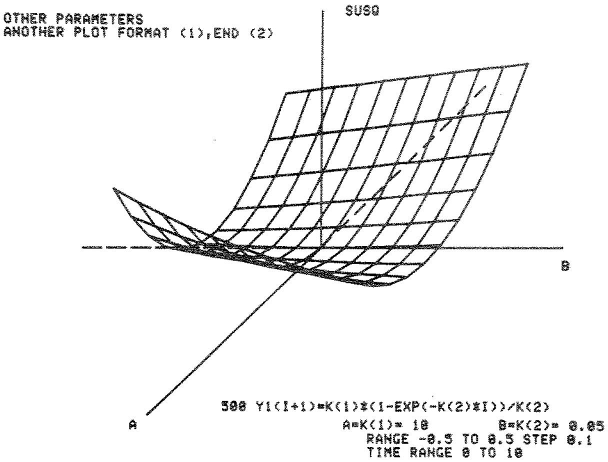
Y = K1Exp(-K2X)



```
OTHER PARAMETERS
ANOTHER PLOT FORMAT (1),END (2)

500 Y1(I+1)=K(1)*(1-EXP(-K(2)*I))/K(2)
                    A=K(1)= 10          B=K(2)= 0.5
                      RANGE -0.5 TO 0.5 STEP 0.1
                      TIME RANGE 0 TO 10
```

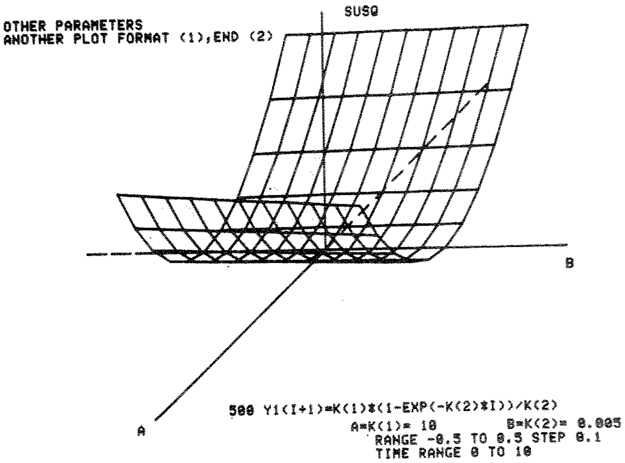
Fig. 10. SUSQ surface of OCM uptake phase.



```
OTHER PARAMETERS
ANOTHER PLOT FORMAT (1),END (2)

500 Y1(I+1)=K(1)*(1-EXP(-K(2)*I))/K(2)
                    A=K(1)= 10          B=K(2)= 0.05
                      RANGE -0.5 TO 0.5 STEP 0.1
                      TIME RANGE 0 TO 10
```

Fig. 11. SUSQ surface of OCM uptake phase.



```
OTHER PARAMETERS
ANOTHER PLOT FORMAT (1),END (2)

500 Y1(I+1)=K(1)*(1-EXP(-K(2)*I))/K(2)
                    A=K(1)= 10          B=K(2)= 0.005
                      RANGE -0.5 TO 0.5 STEP 0.1
                      TIME RANGE 0 TO 10
```

Fig. 12. SUSQ surface of OCM uptake phase.

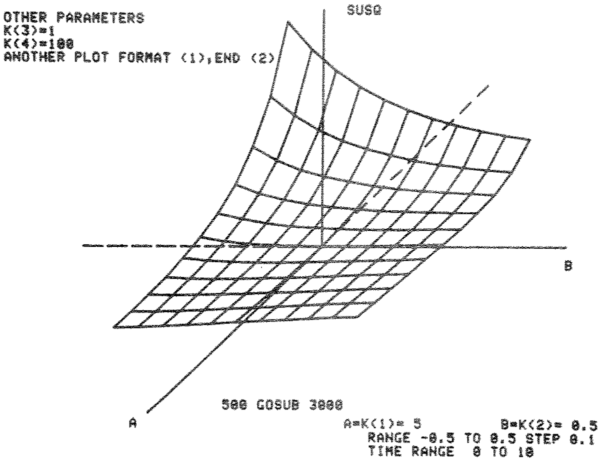


Fig. 13. SUSQ surface of TCM uptake phase. In this and in Fig. 14-18, the subroutine 3000 solves the OCM differential equations. The rate constants are renumbered for programming convenience: K(1)=K2, K(2)=K4, K(3)=K3, K(4)=K1 (see Appendix, TCMP).

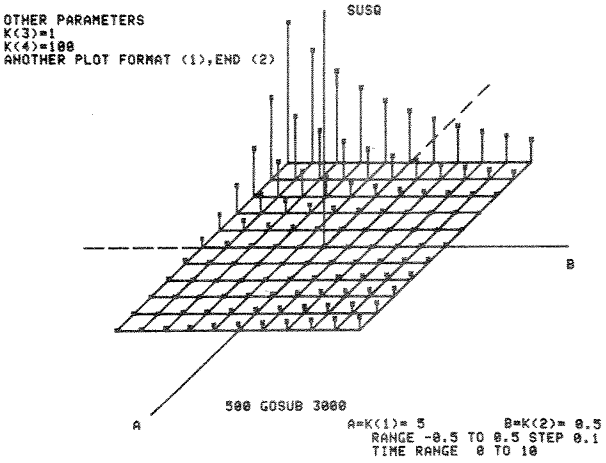


Fig. 14. SUSQ surface of TCM uptake phase; same parameters as in Fig. 13.

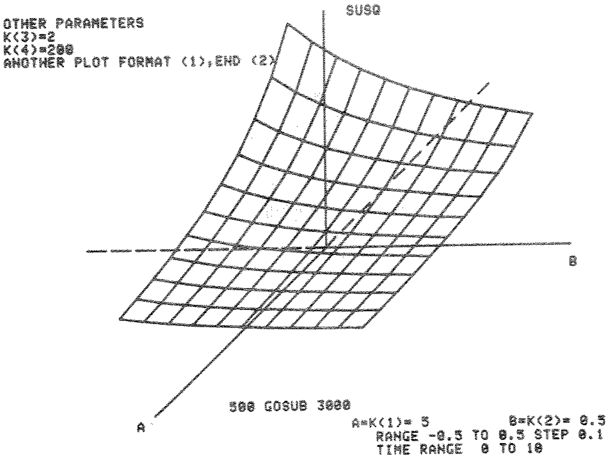


Fig. 15. SUSQ surface for TCM uptake phase; K(3) and K(4) at twice the true values.

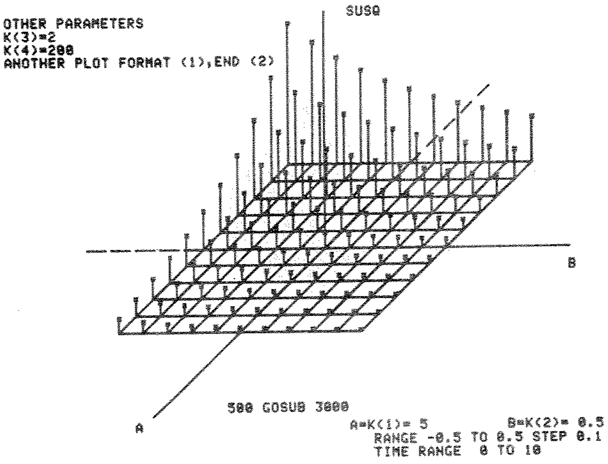


Fig. 16. SUSQ surface for TCM uptake phase; "spike" presentation of Fig. 15.

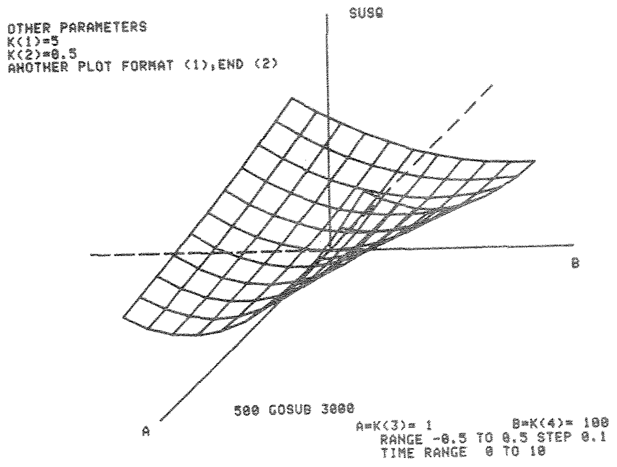


Fig. 17. SUSQ surface for TCM uptake phase.

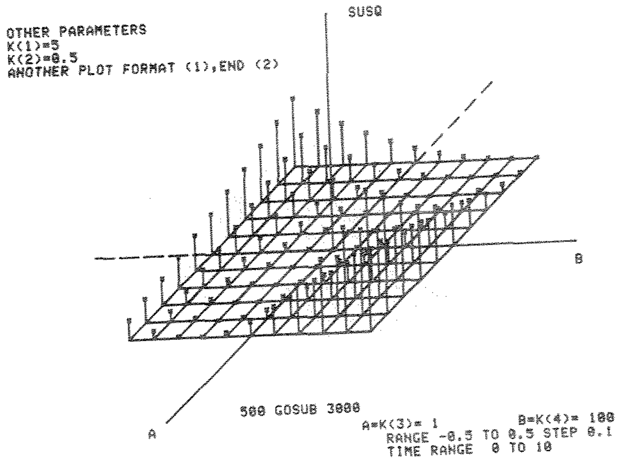


Fig. 18. SUSQ surface for TCM uptake phase; "spike" presentation of Fig. 17.

Since these areas are generally more extensive for the TCM than for the OCM, it is much more difficult to fit the former to experimental data.

Listing of a program for the examination of the SUSQ surfaces is given in the Appendix.

CONCLUDING REMARKS

The first requirement for a successful application of TCM is good quality data. TCM fits are more difficult to perform than OCM fits and attempts to fit poor quality data by the TCM are frustrating and may lead to meaningless results.

Experimenting with the program TCMP is advisable to give the user a feeling for the effects of the different rate constants on the solution. The program may be expanded easily to give a graphic presentation of the solution.

Having made the crucial decision to use the TCM, *the next step* should be the attempt to fit the excretion data (there is one less rate constant to worry about). Only after obtaining a reasonable fit of the excretion data one should proceed to the uptake data.

The better the data, the more flexibility the user has to select the initial rate constants. But even for the best quality data the program TCMS will go astray when the *initial rate constant estimates* are "way out". In any case, the program should be run several times to make sure that even after starting from different directions, the program is finding the same solution. It is very important to realize that the fit is not an iterative process. The solution is not necessarily improved by using the results of one attempt as the starting values in another attempt to fit the data.

It is possible to use the TCM even if data for the two compartments taken individually are not available. However, rate constants obtained in this case *may not reflect accurately the real situation* since the distinction between the kinetics of the two compartments has been lost by their addition.

It is also important to keep in mind that the TCM deals with amounts, not with concentrations (except for water). Concentrations may be used instead of amounts only when the compartments are of equal size. Otherwise, size of the compartments must be taken into consideration when converting from amounts to concentrations.

Both programs TCMP and SUSQS (see Appendix) may be used to estimate the error of the determined rate constants from the estimated error of the measured amount of chemical in the compartments. The procedure consists of varying the rate constants and observing calculated amounts. TCMS also gives an indication of the error of the rate constants ("next best answer").

Good luck.

ACKNOWLEDGMENTS

I thank Ms. R. Garnett for editorial assistance, Messrs. W. G. McMullon and F. Cunningham for preparation of figures, Mrs. B. Fawkes for typing and arranging the report, Dr. D. W. McLeese and Mr. A. Sreedharan for comments, and Mrs. M. Irwin for proofreading.

REFERENCES

Bronson, R. 1973. Modern introductory differential equations. Schaum's Outline Series in Mathematics. McGraw-Hill, New York.

Deming, S. N., and S. L. Morgan. 1973. Simplex optimization of variables in analytical chemistry. Anal. Chem. 40: 278A-283A.

Könemann, H., and K. van Leeuwen. 1980. Toxicokinetics in fish: Accumulation and elimination of six chlorobenzenes by guppies. Chemosphere 9: 3-19.

Metzler, C. M. 1980. Factors affecting pharmacokinetic modeling, p. 15-29. In K. S. Albert (Ed.) Drug absorption and disposition. American Pharmaceutical Association, Academy of Pharmaceutical Sciences.

Wei, J. 1975. Least square fitting of an elephant. Chemtech: 128-129.

Zitko, V. 1980. Relationships governing the behaviour of pollutants in aquatic ecosystems and their use in risk assessment, p. 243-265. In Proceedings of the Sixth Annual Aquatic Toxicity Workshop, November 6 & 7, 1979, Winnipeg, Manitoba. Can. Tech. Rep. Fish. Aquat. Sci. 975.

APPENDIX

PROGRAM TCMP (two-compartment model prediction)

The program written in Tektronix Plot 50 BASIC solves the TCM differential equations for given rate constants and initial concentrations, prints the solution, and calculates Q1 and Q2 (quantity in compartments 1 and 2, respectively) for a specified time period.

The program is useful for armchair experiments and is incorporated as a "driver" in the TCMS (two-compartment model simplex) and SUSQS (SUSQ surface) given later in this Appendix.

TCMP follows exactly the procedure for solving the TCM differential equations outlined earlier in this report. First, the eigenvalues F() are calculated, followed by the eigenvectors D() and constants F() corresponding to the given initial concentrations C(). Note that for programming convenience, the rate constants X() are internally renumbered. The renumbering does not affect the user but simplifies considerably the switch from differential equations for uptake to differential equations for excretion.

The program is readily adaptable to other BASIC interpreters. The only Tektronix-specific feature is the handling of array variables. In Tektronix BASIC these can be inputted or printed by simple statements INPUT C or PRINT C. Other interpreters may require a loop for both operations. The TIME, COMP1 COMP2 printout has not been formatted by a PRINT USING statement since most BASIC interpreters differ considerably in this aspect. The user should provide his own formatting.

```
100 PRINT "************* PROGRAM TCMP  ******************"
105 REM STANDARD 2 COMP MODEL BUT RENUMBERED CONSTANTS
110 PRINT "UPTAKE (3), EXCRETION (2)"
120 INPUT Q1
130 N=Q1+1
140 DIM X(N),E(Q1),D(Q1,Q1),D1(Q1,Q1),C(Q1),F(Q1)
150 PRINT "INITIAL AMOUNT: COMP.1,2,(WATER)"
160 INPUT C
170 PRINT "STARTING VALUES FOR RATE CONSTANTS "
180 PRINT "K2"
190 INPUT X(1)
200 PRINT "K4"
210 INPUT X(2)
220 PRINT "K3"
230 INPUT X(3)
240 IF Q1=2 THEN 270
250 PRINT "K1"
260 INPUT X(4)
270 A=-X(1)-X(2)-X(3)
280 A1=SQR(A↑2-4*X(1)*X(2))
290 E(1)=0.5*(A-A1)
300 E(2)=0.5*(A+A1)
310 IF Q1=2 THEN 330
320 E(3)=0
330 D(1,1)=X(2)/(X(1)+X(3)+E(1))
340 D(2,1)=1
350 IF Q1=2 THEN 370
360 D(3,1)=0
370 D(1,2)=X(2)/(X(1)+X(3)+E(2))
380 D(2,2)=1
390 IF Q1=2 THEN 440
400 D(3,2)=0
410 D(1,3)=X(4)/X(1)
420 D(2,3)=X(3)*X(4)/(X(1)*X(2))
430 D(3,3)=1
440 D3=D(1,1)-D(1,2)
450 IF Q1=2 THEN 500
460 F(1)=(C(1)+C(3)*D(1,2)*D(2,3)-C(3)*D(1,3)-C(2)*D(1,2))/D3
470 F(2)=(C(2)*D(1,1)+C(3)*D(1,3)-C(3)*D(2,3)*D(1,1)-C(1))/D3
480 F(3)=(C(3)*D(1,1)-C(3)*D(1,2))/D3
490 GO TO 520
500 F(1)=(C(1)-C(2)*D(1,2))/D3
510 F(2)=(C(2)*D(1,1)-C(1))/D3
520 FOR I=1 TO Q1
530 FOR J=1 TO Q1
540 D1(I,J)=F(J)*D(I,J)
550 NEXT J
560 NEXT I
570 PRINT "TIME RANGE (0---- > )"
580 INPUT T9
590 DIM Y3(T9+1),Y4(T9+1)
600 Y3=0
610 Y4=0
620 FOR J1=0 TO T9
630 FOR J2=1 TO Q1
640 Y3(J1+1)=Y3(J1+1)+D1(1,J2)*EXP(E(J2)*J1)
650 Y4(J1+1)=Y4(J1+1)+D1(2,J2)*EXP(E(J2)*J1)
660 NEXT J2
670 NEXT J1
```

PROGRAM TCMP (cont'd)

```
675 PRINT @37,26:1
676 PRINT @40:"*******  TCMP   **************"
680 PRINT @40:"RATE CONSTANTS"
690 PRINT @40:"K2=";X(1),"K4=";X(2),"K3=";X(3)
700 IF Q1=2 THEN 730
710 PRINT @40:"K1=";X(4)
720 GO TO 740
730 PRINT @40:" "
740 PRINT @40:"INITIAL AMOUNTS"
750 PRINT @40:C
760 PRINT @40:"TIME                    COMP.1                   COMP.2          1+2"
770 PRINT @40:"-------------------------------------------------------------------"
780 FOR J1=Ø TO T9
785 Y=Y3(J1+1)+Y4(J1+1)
790 PRINT @40:J1,Y3(J1+1),Y4(J1+1),Y
800 NEXT J1
810 PRINT @40:"-------------------------------------------------------------------"
820 PRINT @40:"SOLUTION:"
830 FOR I=1 TO Q1
840 PRINT @40:"Q(";I;")="
850 FOR J2=1 TO Q1
860 IF D1(I,J2)=Ø THEN 880
870 PRINT @40:D1(I,J2);"*EXP(";E(J2);"*T) "
880 NEXT J2
890 PRINT @40:" "
900 NEXT I
905 PRINT @37,26:Ø
910 END
```

```
*******  TCMP   **************
RATE CONSTANTS
K2=5                K4=Ø.5              K3=1
K1=1ØØ
INITIAL AMOUNTS


 Ø                   Ø                   Ø.1


TIME                 COMP.1              COMP.2          1+2
----------------------------------------------------------------
 Ø                  -7.1Ø5427358E-15  Ø                  -7.1Ø5427358E-15
 1                   1.741843107Ø6     1.15569842258      2.89754152964
 2                   1.83118933Ø38     2.11297531311      3.94416464349
 3                   1.888Ø35379Ø8     2.74835797855      4.63639335763
 4                   1.92573521Ø57     3.1698ØØ72611      5.Ø9553593668
 5                   1.95Ø741Ø4816     3.44933869212      5.4ØØØ7974Ø28
 6                   1.9673271229      3.6347529Ø147      5.6Ø2Ø8ØØ2437
 7                   1.978328469Ø8     3.757735942Ø8      5.736Ø6441116
 8                   1.9856255312      3.8393Ø913Ø74      5.82493466193
 9                   1.99Ø46558575     3.89341565692      5.88388124267
 1Ø                  1.99367593638     3.9293Ø387245      5.9229798Ø883
----------------------------------------------------------------
SOLUTION:
Q(1)=
 -1.61631563443*EXP(-6.Ø894541729*T)
 -Ø.383684365572*EXP(-Ø.41Ø5458271*T)
 2*EXP(Ø*T)

Q(2)=
 Ø.289172356447*EXP(-6.Ø894541729*T)
 -4.28917235645*EXP(-Ø.41Ø5458271*T)
 4*EXP(Ø*T)

Q(3)=
 Ø.1*EXP(Ø*T)
```

PROGRAM TCMP (cont'd)

```
*******  TCMP   **************
RATE CONSTANTS
K2=5                 K4=Ø.5               K3=1

INITIAL AMOUNTS


 1.99                3.93

TIME                  COMP.1              COMP.2          1+2
------------------------------------------------------------
 Ø                   1.99                3.93                5.92
 1                   Ø.253956123738      2.79743628455       3.Ø5139240829
 2                   Ø.166Ø29862Ø92      1.85593850905       2.Ø2196837114
 3                   Ø.11Ø12Ø15684       1.231Ø22927Ø5       1.341143Ø8389
 4                   Ø.Ø73Ø413783654     Ø.816522873714      Ø.88956425207 9
 5                   Ø.Ø484474778298     Ø.541589914243      Ø.59ØØ37392Ø73
 6                   Ø.Ø321346360372     Ø.3592301 5Ø985     Ø.391364787Ø23
 7                   Ø.Ø213145220247     Ø.23827308 8149     Ø.259587610174
 8                   Ø.Ø14137669Ø44      Ø.1580437Ø6466      Ø.17218137551
 9                   Ø.ØØ937734778969    Ø.1Ø4828511468      Ø.1142Ø5859257
 1Ø                  Ø.ØØ621988329867    Ø.Ø695315053174     Ø.0757513886161
------------------------------------------------------------
SOLUTION:
Q(1)=
 1.612636310 79*EXP(-6.Ø894541729*T)
 Ø.377363689213*EXP(-Ø.41Ø5458271*T)

Q(2)=
 -Ø.28851409474*EXP(-6.Ø894541729*T)
 4.21851409474*EXP(-Ø.41Ø5458271*T)
```

PROGRAM TCMS (TWO-COMPARTMENT MODEL SIMPLEX)

The simplex algorithm of this program is a slightly modified version of the Nelder-Mead algorithm as available in the Statistics Package by Tektronix. The program is written in HP2000 BASIC in a format almost compatible with HP Terminal BASIC and, with slight modifications, compatible with many other BASIC interpreters. In HP Terminal BASIC arrays are dimensioned directly in terms of variables. For this, the "DIM" statements should be deleted and the "REDIM" statements changed to "DIM" (for example, delete 505 and change 520 to DIM....). Some interpreters, for example TRS-80, return "-1" as the "true" condition. For such interpreters, the signs in the statement 3830 must be changed.

The program requires about 15K of RAM and when used in Terminal BASIC, memory must be set to 15000 ("SET SIZE=15000"). The program runs much faster on the mainframe HP than on the terminal. The current version does not provide a hardcopy printout and the user must copy the final display on the terminal.

Statements 100-140 contain "program heading" and alert the user to enter data in statements 530-559. This somewhat inconvenient form of data entry is preferred to INPUT since the program is likely to be used more than once with the same set of data to obtain a good fit. The data format is time X1(I) and either the amount in compartment Y1(I) when data are not available on both compartments individually, or amount in compartment 1, Y1(I) and amount in compartment 2, Y2(I). The distinguishing flag is Q7 (statement 1515).

After the data entry has been completed the program is restarted by the command RUN,1500 on the mainframe, or RUN 1500 on the terminal. The statements 1000-1380 contain TCMP solving the TCM differential equations for rate constants supplied by the simplex algorithm. The statement 1390 calculates the SUSQ (Y) and returns this value to the main program. The branch in 1385 is used to print the calculated quantities in compartments 1, 2 or 1 and 2, once the best fitting rate constants have been found (Q9 is the "flag"). Statements 1010-1040 are used to eliminate negative rate constants and to drive the simplex in another direction by assigning very unfavorable response to Y (line 1040). Similarly, the statements 1063-1065 are used to eliminate complex eigenvalues.

By modifying statements 1000-1040, the program may be used to fit other functions to data. The fitted parameters must be in the array X(I) and the subroutine must return a value of Y to the main program. The printout portion of the program may have to be modified as well.

The main program starts by the statement 1500. The string variable K$(I) contains rate constant symbols since the program uses a different numbering. The string variable R$ is for remarks such as name of data set, etc. X() is the rate constants array, E() contains eigenvalues, D() eigenvectors, D1() coefficients of the solution, C() initial quantities, F() constants of the solution for given C(). B0() contains simplex coordinates. Movements of the simplex may be monitored by printing this array. Simplex coordinates are in the first N rows. The corresponding SUSQ values are in the last row. E$ is for Y/N answers in the procedural questions. S0(), S2() contain initial rate constants, and S1() contains the step size for the formation of the initial simplex. C0() contains centroid coordinates in column 1, projection coordinates in column 2, and expansion coordinates in column 3.

Statements 1590-1964 allow the input of various procedural parameters and are more or less self-explanatory. The weighting factor W (statement 1961) sometimes improves the fit when the quantities in compartments 1 and 2 are considerably different. In such a case the SUSQ may be biased in favor of the compartment with larger quantities and W tends to compensate for this effect. A9, B9, and C9 (1970-1990) are coefficients for reflection, contraction, and expansion, respectively. Z7 and Z4 are counters for iterations and for printing frequency.

The initial simplex is formed by statements 2020-2125 and the corresponding SUSQ values are determined by statements 2160-2280. Z0 is the column number in B0(). Coordinates of the simplex points become, in turn, the variable X() for use by the TCM driver (GOSUB 1000). Note that the data are read only when going through the routine for the first time (GOSUB 500). Z9 counts SUSQ evaluations (the use of GOSUB 1000).

The best and worst responses (smallest and largest SUSQ) in the initial simplex are determined by statements 2320-2460.

The statements 2470-2474 initialize the "centroid" matrix C0(). Coordinates of centroid and reflection opposite the worst response (statement 2500) are calculated in statements 2490-2600. The actual calculation of the coordinates is in the statement 2570 for the centroid (point D, Fig. 3). SUSQ at reflection is calculated (2590-2650) and becomes F1. If SUSQ at reflection is smaller than the smallest SUSQ of the original simplex, then SUSQ is calculated at expansion (point F, Fig. 3) by statements 2690-2760. If SUSQ at reflection does not give the best response, the program goes from statement 2630 to statement 2900. On the other hand, if the expansion does not provide the smallest SUSQ, it is "back to reflection" (2770). Otherwise, coordinates of expansion replace the coordinates of the worst response to form a new simplex (statements 2780-2820). In the "back to reflection" case, the reflected point forms the new simplex (statements 2840-2880).

Turning now to the case when SUSQ at reflection is equal to or larger than the best response (statement 2900, reached from statement 2680), if the SUSQ is within the range of the two better responses (between SUSQ's at B and C, Fig. 3), J1 in 2950 is smaller than N and a new simplex, including the reflected point, is formed (branch back to 2840). If, on the other hand, SUSQ at reflection is worse than the worst response, the program heads for contraction (3030 from 2960). If the SUSQ at reflection is worse than those at the two "better" points, but better than that at the worst point, the worst point is replaced by reflection (2790-3010) and contraction is calculated (3030-3060). These moves result in points H and G (Fig. 3), respectively. If the response in these points is still worse than the worst response of the original simplex, the whole simplex contracts (3170-3220).

SUSQ's of new points are calculated by statements 3230-3320. SUSQ variance is calculated by statements 3380-3510 and if less than specified value (E1, 3520), the algorithm is terminated. The algorithm is also terminated when the number of

SUSQ evaluations exceed 150 (3525). It may happen that the simplex algorithm is not able to reach the specified variance and statements 5000-5160 take care of this situation.

Intermediate results are printed by the subroutine in statements 3820-4030. When the specified value of SUSQ variance is reached, entered data, calculated data, and the fitted rate constants are printed by statements 4000-4160 and 3630-3810.

```
 list 1ØØ-123Ø
FCMS
  1ØØ PRINT "TWO COMPARTMENT RATE CONSTANTS BY SIMPLEX"
  11Ø PRINT "VERSION 2;12.11.1981"
  125 PRINT "DATA IN 5ØØ-6ØØ"
  13Ø PRINT "CONTINUE BY 'RUN,15ØØ'"
  14Ø STOP
  5ØØ REM XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
  5Ø5 DIM X1[5Ø],Y1[5Ø],Y2[5Ø],Y3[5Ø],Y4[5Ø]
  51Ø INPUT "NUMBER OF DATA POINTS ",N2
  52Ø REDIM X1[N2],Y1[N2],Y2[N2],Y3[N2],Y4[N2]
  53Ø DATA 1,1.74,1.15,2,1.83,2.11,4,1.93,3.17,8,1.99,3.84
  531 DATA 4,7.22,21.2,5,7.44,27.4,6,7.64,33.4
  532 DATA 7,7.84,39.3,8,8.Ø4,45.1,9,8.23,5Ø.8
  533 DATA 1Ø,8.42,56.5
  56Ø FOR I=1 TO N2
  565   IF Q7=2 THEN 575
  57Ø   READ X1[I],Y1[I],Y2[I]
  572   GOTO 58Ø
  575   READ X1[I],Y1[I]
  58Ø NEXT I
  59Ø RETURN
  999 REM XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
 1ØØØ REM STANDARD 2COMP MODEL BUT WATCH FOR RENUMBERED CONSTANTS
 1Ø1Ø FOR V9=1 TO N
 1Ø2Ø   IF X[V9]<Ø THEN 1Ø3Ø
 1Ø25 NEXT V9
 1Ø26 GOTO 1Ø6Ø
 1Ø3Ø PRINT "NEGATIVE RATE CONSTANT ";V9
 1Ø4Ø Y=1E7*V9
 1Ø5Ø GOTO 141Ø
 1Ø6Ø A=-X[1]-X[3]-X[2]
 1Ø61 A2=A**2-4*X[1]*X[2]
 1Ø62 IF A2>Ø THEN 1Ø66
 1Ø63 Y=1E7
 1Ø64 PRINT "COMPLEX EIGENVALUES"
 1Ø65 GOTO 141Ø
 1Ø66 A1=SQR(A2)
 1Ø7Ø E[1]=(A-A1)/2
 1Ø75 E[2]=(A+A1)/2
 1Ø8Ø IF Q1=2 THEN 1Ø9Ø
 1Ø85 E[3]=Ø
 1Ø9Ø D[1,1]=X[2]/(X[1]+X[3]+E[1])
 1Ø95 D[2,1]=1
 11ØØ IF Q1=2 THEN 111Ø
 11Ø5 D[3,1]=Ø
 111Ø D[1,2]=X[2]/(X[1]+X[3]+E[2])
 1115 D[2,2]=1
 112Ø IF Q1=2 THEN 1165
 1125 D[3,2]=Ø
 113Ø D[1,3]=X[4]/X[1]
 1135 D[2,3]=X[4]*X[3]/(X[1]*X[2])
 114Ø D[3,3]=1
 1165 D3=D[1,1]-D[1,2]
 12ØØ IF Q1=2 THEN 1225
 12Ø5 F[1]=(C[1]+C[3]*D[1,2]*D[2,3]-C[3]*D[1,3]-C[2]*D[1,2])/D3
 121Ø F[2]=(C[2]*D[1,1]+C[3]*D[1,3]-C[3]*D[2,3]*D[1,1]-C[1])/D3
 1215 F[3]=(C[3]*D[1,1]-C[3]*D[1,2])/D3
 122Ø GOTO 1235
 1225 F[1]=(C[1]-C[2]*D[1,2])/D3
 123Ø F[2]=(C[2]*D[1,1]-C[1])/D3
>
```

PROGRAM TCMS (cont'd)

```
 list 1235-175Ø
TCMS
 1235 FOR I=1 TO Q1
 1237   FOR J=1 TO Q1
 124Ø     D1[I,J]=F[J]*D[I,J]
 1245   NEXT J
 1247 NEXT I
 13ØØ REM XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
 131Ø REM SUSQ CALCULATION
 132Ø Y=Ø
 1321 FOR J1=1 TO N2
 1322   Y3[J1]=Ø
 1323   Y4[J1]=Ø
 1325 NEXT J1
 133Ø FOR J1=1 TO N2
 135Ø   FOR J2=1 TO Q1
 136Ø     Y3[J1]=Y3[J1]+D1[1,J2]*EXP(E[J2]*X1[J1])
 137Ø     Y4[J1]=Y4[J1]+D1[2,J2]*EXP(E[J2]*X1[J1])
 1371     IF Q7=1 THEN 138Ø
 1375     Y3[J1]=Y3[J1]+Y4[J1]
 138Ø   NEXT J2
 1385   IF Q9=Ø THEN 14ØØ
 1386   IF Q7=2 THEN 1395
 139Ø   Y=Y+(Y1[J1]-Y3[J1])**2/Y1[J1]**2+(Y2[J1]-Y4[J1])**2/Y2[J1]**2
 1391   GOTO 14ØØ
 1395   Y=Y+(Y1[J1]-Y3[J1])**2/Y1[J1]**2
 14ØØ NEXT J1
 141Ø RETURN
 15ØØ REM STARTING VALUES AND STEP SIZE
 15Ø1 Q9=1
 15Ø2 DIM K$[4,2],R$[1,6Ø]
 15Ø4 INPUT "REMARKS ",R$[1]
 15Ø5 INPUT "UPTAKE (3), EXCRETION (2) ",Q1
 151Ø N=Q1+1
 1515 INPUT "TWO COMPARTMENTS INDIVIDUALLY (1) OR TOGETHER (2) ",Q7
 1518 DIM X[4],E[3],D[3,3],D1[3,3],C[3],F[3]
 152Ø REDIM X[N],E[Q1],D[Q1,Q1],D1[Q1,Q1],C[Q1],F[Q1]
 1525 PRINT "INITIAL AMOUNT: COMP.1,COMP.2, (WATER)"
 153Ø FOR I=1 TO Q1
 1532   INPUT C[I]
 1534 NEXT I
 156Ø N1=N+1
 1571 K$[1]="K2"
 1572 K$[2]="K4"
 1573 K$[3]="K3"
 1574 K$[4]="K1"
 1575 DIM BØ[5,5],SØ[4],E$[1,1],CØ[5,3],S1[4],S2[4]
 158Ø REDIM BØ[N1,N1],SØ[N],CØ[N,3],S1[N],S2[N]
 16ØØ PRINT "AUTOMATIC STEP SIZE IS Ø.1 OF STARTING VALUE"
 161Ø PRINT "DO YOU WISH TO ENTER STEP SIZE? "
 162Ø INPUT E$[1]
 163Ø IF E$[1]="Y" THEN 171Ø
 164Ø PRINT "ENTER ESTIMATES OF RATE CONSTANTS IN ORDER K2,K4,K3,K1"
 165Ø FOR I1=1 TO N
 166Ø   PRINT "STARTING VALUE FOR ";K$[I1]
 167Ø   INPUT SØ[I1]
 1675   S2[I1]=SØ[I1]
 168Ø   S1[I1]=.1*ABS(SØ[I1])
 169Ø NEXT I1
 17ØØ GOTO 176Ø
 171Ø PRINT "ENTER STARTING VALUES AND STEP SIZE"
 172Ø FOR I1=1 TO N
 173Ø   PRINT "STARTING VALUE AND STEP SIZE FOR VARIABLE";I1
 174Ø   INPUT SØ[I1],S1[I1]
```

PROGRAM TCMS (cont'd)

```
 list 1745-227Ø
TCMS
 1745 S2[I1]=SØ[I1]
 175Ø NEXT I1
 176Ø E1=.ØØØØ01
 177Ø PRINT "CRITERIUM FOR CONVERGENCE IS THE VARIANCE <=";E1
 178Ø PRINT "DO YOU WISH TO CHANGE THIS VALUE?"
 179Ø INPUT E$[1]
 18ØØ IF E$[1]="N" THEN 183Ø
 181Ø PRINT "CONVERGENCE TEST VALUE="
 182Ø INPUT E1
 183Ø Z8=1
 184Ø Z5=1
 185Ø PRINT "CONVERGENCE IS TESTED EVERY";Z8;"ITERATIONS"
 186Ø PRINT "DO YOU WISH TO CHANGE THIS FREQUENCY?"
 187Ø INPUT E$[1]
 188Ø IF E$[1]="N" THEN 191Ø
 189Ø INPUT "CHECK FOR CONVERGENCE WITH FREQUENCY ",Z8
 191Ø PRINT "RESULTS ARE PRINTED EVERY";Z5;"ITERATIONS"
 192Ø PRINT "DO YOU WISH TO CHANGE THIS FREQUENCY "
 193Ø INPUT E$[1]
 194Ø IF E$[1]="N" THEN 195Ø
 195Ø PRINT "PRINT RESULTS WITH FREQUENCY?"
 1955 INPUT Z5
 196Ø IF Q7=2 THEN 197Ø
 1961 INPUT "WEIGHTING FACTOR IN SUSQ CALCULATION: 1(1),OTHER (2) ",Q8
 1962 IF Q8=1 THEN 1969
 1963 INPUT "WEIGHTING FACTOR ",W
 1964 GOTO 197Ø
 1969 W=1
 197Ø A9=1
 198Ø B9=.5
 199Ø C9=2
 2ØØØ Z7=Ø
 2Ø1Ø Z4=Ø
 2Ø2Ø REM GENERATE THE INITIAL SIMPLEX
 2Ø3Ø FOR I1=1 TO N
 2Ø4Ø   BØ[I1,N1]=SØ[I1]
 2Ø5Ø NEXT I1
 2Ø6Ø FOR I1=1 TO N
 2Ø7Ø   SØ[I1]=SØ[I1]+S1[I1]
 2Ø8Ø   FOR J1=1 TO N
 2Ø9Ø     BØ[J1,I1]=SØ[J1]
 21ØØ   NEXT J1
 212Ø   SØ[I1]=SØ[I1]-S1[I1]
 2125 NEXT I1
 213Ø REM INITIAL SIMPLEX DETERMINED
 216Ø REM EVALUATE FUNCTION AT ORIGINAL SIMPLEX POINTS
 217Ø ZØ=1
 218Ø FOR I1=1 TO N
 219Ø   X[I1]=BØ[I1,ZØ]
 22ØØ NEXT I1
 221Ø IF ZØ>1 THEN 223Ø
 222Ø GOSUB 5ØØ
 223Ø GOSUB 1ØØØ
 224Ø BØ[N1,ZØ]=Y
 225Ø ZØ=ZØ+1
 226Ø IF ZØ<=N1 THEN 218Ø
 227Ø Z9=N1
>
```

```
 list 228Ø-28ØØ
TCMS
 228Ø REM FUNCTION WAS DETERMINED AT EACH SIMPLEX POINT
 231Ø REM DETERMINE INDEX OF LARGEST AND SMALLEST SUSQ
 232Ø H8=BØ[N1,1]
 233Ø H9=1
 234Ø L9=1
 235Ø L8=H8
 236Ø FOR I1=2 TO N1
 237Ø   IF H8>=BØ[N1,I1] THEN 241Ø
 238Ø   H8=BØ[N1,I1]
 239Ø   H9=I1
 24ØØ   GOTO 244Ø
 241Ø   IF L8<=BØ[N1,I1] THEN 244Ø
 242Ø   L8=BØ[N1,I1]
 243Ø   L9=I1
 244Ø NEXT I1
 245Ø REM H9=INDEX OF LARGEST VALUE AND H8=LARGEST VALUE
 246Ø REM L9=INDEX OF SMALLEST VALUE AND L8=SMALLEST VALUE
 247Ø FOR I=1 TO N
 2471   FOR J=1 TO 3
 2472     CØ[I,J]=Ø
 2473   NEXT J
 2474 NEXT I
 248Ø REM DETERMINE CENTROID OF POINTS NOT INCLUDING LARGEST VALUE
 249Ø FOR I1=1 TO N1
 25ØØ   IF I1=H9 THEN 254Ø
 251Ø   FOR J1=1 TO N
 252Ø     CØ[J1,1]=CØ[J1,1]+BØ[J1,I1]
 253Ø   NEXT J1
 254Ø NEXT I1
 255Ø REM CALCULATE REFLECTION THRU CENTROID
 256Ø FOR I1=1 TO N
 257Ø   CØ[I1,1]=CØ[I1,1]/N
 258Ø   CØ[I1,2]=(1+A9)*CØ[I1,1]-A9*BØ[I1,H9]
 259Ø   X[I1]=CØ[I1,2]
 26ØØ NEXT I1
 261Ø REM CENTROID AND REFLECTION CALCULATED
 263Ø GOSUB 1ØØØ
 264Ø REM FUNCTION CALCULATED AT REFLECTION
 265Ø Z9=Z9+1
 266Ø REM CHECK IF REFLECTION < SIMPLEX MINIMUM
 267Ø F1=Y
 268Ø IF Y>=L8 THEN 29ØØ
 269Ø REM EXPANSION
 27ØØ FOR I1=1 TO N
 271Ø   CØ[I1,3]=(1-C9)*CØ[I1,1]+C9*CØ[I1,2]
 272Ø   X[I1]=CØ[I1,3]
 273Ø NEXT I1
 274Ø REM EXPANSION CALCULATED
 275Ø GOSUB 1ØØØ
 276Ø Z9=Z9+1
 277Ø IF Y>=L8 THEN 284Ø
 278Ø FOR I1=1 TO N
 279Ø   BØ[I1,H9]=X[I1]
 28ØØ NEXT I1
 281Ø BØ[N1,H9]=Y
 282Ø GOTO 334Ø
 283Ø REM REPLACE WORST RESPONSE BY REFLECTION
 284Ø FOR I1=1 TO N
 285Ø   BØ[I1,H9]=CØ[I1,2]
 286Ø NEXT I1
 287Ø BØ[N1,H9]=F1
 288Ø GOTO 334Ø
 289Ø REM CHECK IF REFLECTION>YI FOR I<>H9
 29ØØ J1=Ø
 291Ø FOR I1=1 TO N1
 292Ø   IF I1=H9 OR F1<BØ[N1,I1] THEN 294Ø
 293Ø   J1=J1+1
 294Ø NEXT I1
 295Ø IF J1<N THEN 284Ø
 296Ø IF F1>H8 THEN 3Ø3Ø
 297Ø FOR I1=1 TO N
```

```
2980   BØ[I1,H9]=CØ[I1,2]
2990 NEXT I1
3000 BØ[N1,H9]=F1
3010 H8=F1
3020 REM CALCULATE CONTRACTION
3030 FOR I1=1 TO N
3040   CØ[I1,3]=B9*BØ[I1,H9]+(1-B9)*CØ[I1,1]
3050   X[I1]=CØ[I1,3]
3060 NEXT I1
3070 GOSUB 1000
3080 Z9=Z9+1
3090 F1=Y
3100 IF F1>H8 THEN 3170
3110 FOR I1=1 TO N
3120   BØ[I1,H9]=CØ[I1,3]
3130 NEXT I1
3140 BØ[N1,H9]=F1
3150 GOTO 3340
3160 REM CONTRACT ALL OF SIMPLEX
3170 FOR I1=1 TO N1
3180   IF I1=L9 THEN 3220
3190   FOR J1=1 TO N
3200     BØ[J1,I1]=.5*(BØ[J1,I1]+BØ[J1,L9])
3210   NEXT J1
3220 NEXT I1
3230 ZØ=1
3240 IF ZØ=L9 THEN 3300
3250 FOR I1=1 TO N
3260   X[I1]=BØ[I1,ZØ]
3270 NEXT I1
3280 GOSUB 1000
3290 BØ[N1,ZØ]=Y
3300 ZØ=ZØ+1
3310 IF ZØ<N1 THEN 3240
3320 Z9=Z9+N
3330 REM CHECK IF DONE
3340 S2=Ø
3350 Z7=Z7+1
3360 Z4=Z4+1
3370 IF Z7<Z8 AND Z4<Z5 THEN 2320
3380 L9=1
3390 L8=BØ[N1,1]
3400 FOR I1=1 TO N1
3410   S2=S2+BØ[N1,I1]
3420   IF L8<=BØ[N1,I1] THEN 3450
3430   L9=I1
3440   L8=BØ[N1,I1]
3450 NEXT I1
3460 S2=S2/N1
3470 F1=Ø
3480 FOR I1=1 TO N1
3490   F1=F1+(BØ[N1,I1]-S2)**2
3500 NEXT I1
3510 F1=F1/N
3520 IF F1<=E1 THEN 3610
3525 IF Z9<200 THEN 3530
3526 GOTO 5000
3530 IF Z4=Z5 THEN 3560
3540 Z7=Ø
3550 GOTO 2320
3560 GOSUB 3820
3570 Z7=Z7*(Z7<Z8)
3580 Z4=Ø
3600 GOTO 2320
3610 GOSUB 3820
3620 GOSUB 4040
3640 PRINT "CONVERGENCE WAS OBTAINED FOR INITIAL VALUES"
3650 FOR I1=1 TO N
3670   PRINT S2[I1];
3690 NEXT I1
3695 PRINT
3700 PRINT "INITIAL CONCENTRATIONS";
3710 FOR I1=1 TO Q1
3720   PRINT C[I1];
```

```
 3730 NEXT I1
 3740 PRINT
 3750 PRINT "WITH VARIANCE LESS THAN ";E1
 3751 IF Q7=2 THEN 3755
 3753 PRINT "WEIGHTING FACTOR ";W
 3756 PRINT "NUMBER OF SUSQ EVALUATIONS ";Z9
 3760 PRINT "SUSQ VALUES ";B0[N1,L9],B0[N1,H9]
 3770 PRINT "BEST ANSWER           NEXT BEST ANSWER           DIFFERENCE"
 3780 FOR I1=1 TO N
 3790   PRINT K$[I1],B0[I1,L9],B0[I1,H9],B0[I1,L9]-B0[I1,H9]
 3800 NEXT I1
 3801 FOR I=1 TO Q1
 3802   PRINT "Y";I;"=";
 3803   FOR J=1 TO Q1
 3804     IF D1[I,J]=0 THEN 3805
 3805     PRINT D1[I,J];"*EXP(";E[J];"*T)";
 3805   NEXT J
 3807   PRINT
 3808 NEXT I
 3810 PRINT "AS ONE COMPARTMENT"
 3811 PRINT "Y=";
 3812 FOR J=1 TO Q1
 3813   PRINT D1[1,J]+D1[2,J];"*EXP(";E[J];"*T)";
 3814 NEXT J
 3815 PRINT
 3816 END
 3820 H9=1
 3830 L8=B0[N1,1]*(L9>1)+B0[N1,2]*(L9=1)
 3840 FOR I1=1 TO N1
 3850   IF I1=L9 OR B0[N1,I1]>L8 THEN 3880
 3860   L8=B0[N1,.1]
 3870   H9=I1
 3880 NEXT I1
 3890 PRINT "BEST ANSWER           NEXT BEST ANSWER           DIFFERENCE"
 3900 FOR I1=1 TO N
 3910   IF I1>9 THEN 3940
 3920   PRINT K$[I1];"= ";
 3930   GOTO 3950
 3940   PRINT K$[I1];"= ";
 3950   PRINT B0[I1,L9],B0[I1,H9],B0[I1,L9]-B0[I1,H9]
 3960 NEXT I1
 3970 PRINT "SUSQ VALUES"
 3980 PRINT B0[N1,L9],B0[N1,H9]
 3990 PRINT "VARIANCE=";F1
 4000 PRINT "NUMBER OF FUNCTION EVALUATIONS=";Z9
>list 4000-9000
TCMS
 4000 PRINT "NUMBER OF FUNCTION EVALUATIONS=";Z9
 4030 RETURN
 4040 REM XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
 4050 REM OUTPUT OF DATA USED BY THE PROGRAM
 4055 Q9=0
 4050 FOR I=1 TO N
 4065   X[I]=B0[I,L9]
 4070 NEXT I
 4075 GOSUB 1050
 4091 PRINT "##################################################"
 4092 PRINT R$[1]
 4093 PRINT "##################################################"
 4100 PRINT "TIME            COMP.1           COMP.2  "
 4105 PRINT "       ENTERED  CALCD     ENTERED CALCD  "
 4110 PRINT "--------------------------------------------------"
 4120 FOR I=1 TO N2
 4125   IF Q7=2 THEN 4135
 4130   PRINT USING 4145;X1[I],Y1[I],Y3[I],Y2[I],Y4[I]
 4131   GOTO 4140
 4135   PRINT USING 4146;X1[I],Y1[I],Y3[I]
 4140 NEXT I
 4145 IMAGE 3D.D6X,2D.3D2X,2D.3D4X,2D.3D2X,2D.3D
 4146 IMAGE 3D.D6X,2D.3D2X,2D.3D
 4150 PRINT "----------------------------------------------------"
 4160 RETURN
```

PROGRAM TCMS (cont'd)

```
5ØØØ REM XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
5Ø1Ø REM NOT CONVERGING ROUTINE
513Ø PRINT "CONVERGENCE NOT ACHIEVED AFTER ";Z9;"EVALUATIONS"
514Ø GOSUB 4Ø4Ø
515Ø PRINT "INITIAL VALUES"
516Ø GOTO 355Ø
>
```

```
####################################################################
5/1.29/1.02/50//0/0/.1 TEST RUN
####################################################################
TIME            COMP .1          COMP .2
            ENTERED  CALCD     ENTERED CALCD
----------------------------------------------------------------------
  1.0       1.740   1.688     1.150   1.130
  2.0       1.830   1.792     2.110   2.104
  4.0       1.930   1.902     3.170   3.201
  8.0       1.990   1.974     3.840   3.923
----------------------------------------------------------------------
CONVERGENCE WAS OBTAINED FOR INITIAL VALUES
 5     1.29        1.02         50
INITIAL AMOUNTS   0        0          .1
WITH VARIANCE LESS THAN    .000001
WEIGHTING FACTOR   1
NUMBER OF SUSQ EVALUATIONS   94
SUSQ VALUES  2.45638E-03          3.13705E-03
BEST ANSWER           NEXT BEST ANSWER            DIFFERENCE
K2               4.35626         4.18307            .173195
K4               .49913          .551319           -5.21896E-02
K3               1.02922         1.09706           -6.78439E-02
K1               86.8079         83.1724            3.63556
Y 1    =-1.54967    *EXP(-5.48845    *T)-.443047        *EXP(-.396167     *T)
 1.99271    *EXP( 0    *T)
Y 2    = .319675    *EXP(-5.48845    *T)-4.42871        *EXP(-.396167     *T)
 4.10903    *EXP( 0    *T)
Y 3    = .1         *EXP( 0     *T)
AS ONE COMPARTMENT
Y=-1.22999    *EXP(-5.48845     *T)-4.87175   *EXP(-.396167     *T)
 6.10175    *EXP( 0    *T)
```

```
####################################################################
TEST DATA EXCRETION 1/.1/2///2/4
####################################################################
TIME            COMP.1           COMP.2
            ENTERED  CALCD     ENTERED CALCD
----------------------------------------------------------------------
   .0       2.000   2.000     3.930   4.000
  1.0        .250    .254     2.800   2.699
  2.0        .166    .162     1.850   1.804
  4.0        .073    .072      .816    .806
  8.0        .014    .014      .158    .161
----------------------------------------------------------------------
CONVERGENCE WAS OBTAINED FOR INITIAL VALUES
 1      .1         2
INITIAL AMOUNTS  2      4
WITH VARIANCE LESS THAN  .000001
WEIGHTING FACTOR  1
NUMBER OF SUSQ EVALUATIONS  139
SUSQ VALUES  4.66545E-03        5.34827E-03
BEST ANSWER           NEXT BEST ANSWER            DIFFERENCE
K2               4.88328        4.83755          .045722
K4               .412093        .414957        -2.86371E-03
K3               .104146        9.91449E-02     5.00138E-03
Y 1    = 1.63712    *EXP(-4.99678    *T) .362881       *EXP(-.402732      *T)
Y 2    =-3.71894E-02    *EXP(-4.99678    *T) 4.03719       *EXP(-.402732
*T)
AS ONE COMPARTMENT
Y= 1.59993    *EXP(-4.99678    *T) 4.40007    *EXP(-.402732     *T)
```

PROGRAM SUSQS (SUSQ SURFACE)

This program determines the TCM SUSQ surface for given rate constants. With minor modifications it may be used to study SUSQ surfaces of other functions as well. The knowledge of these surfaces helps to appreciate problems encountered in fitting functions to experimental data. The program is written in Tektronix Plot 50 BASIC. Both numerical and graphical output are provided. The PRINT@37 and PRINT@40 statements are due to our peculiar setup which uses a teletype as a printer.

The program is initialized and parameters are entered in statements 100-720. A convenient projection angle is 45° (statement 190). The deviation in parameter range (statement 200) must be <1. Too small a step (statement 220) will cause inconveniently long execution times and crowd the graphics display.

S1 (statement 240) is a graphics scaling factor; T3 (270) is the number of function points. As in the other programs, the TCM rate constants are renumbered for programming convenience. The subroutine 3000 is again TCMP. The STOP statement (550) gives a chance to examine the function values printed by statement 540 and to abort the execution in case of unsuitable function values. Two rate constants are selected for variation (statements 590-600). The remaining rate constants may either keep their "true" values or the values may be changed (640-720).

SUSQ values are calculated by the loop in statements 759-940 and are placed in the array P(). Values of the two rate constants selected for variation are obtained by statements 830 and 840. Maximum value of P() is found in statements 970-1050 and a scaling factor to a maximum P() value of 30 is found (1080). The graphics display is a three-dimensional projection and requires further scaling since points of different A coordinates (the three-dimensional coordinate perpendicular to the screen) have different projected Y coordinates (see below). This scaling is performed by statements 1140-1480, so that in no case do the values of Z() exceed 50 (space on the screen for graphics display).

Two options are available for graphics display (1490). "Spikes" give a somewhat better appreciation of relative SUSQ values; the "net" presentation may give a better picture of the shape of the surface.

The Tektronix screen coordinates are 0-130 on the horizontal and 0-100 on the vertical axis. Other screens are likely to have other coordinates and the display must be modified accordingly. The center of the screen (coordinates 65,50) is selected as the origin (statements 1540, 1550). Statement 1550 automatically transforms the screen coordinates to new coordinates X, Y, with origin 0, 0 in the center of the screen. Statements 1560-1710 draw a projection of the three-dimensional axes A, B, and SUSQ and statements 1740-1790 print the labels.

The transformation from the three-dimensional coordinates A, B, C4 (SUSQ) to planar coordinates X, Y (Fig. 1A) is in the subroutine 2800-2830. To plot the "net" (statements 1820-2160) only the X1 coordinate has to be determined (1920) since the array Z() already contains the Y coordinates. Statements 1940 and 2110 find the "edge" of the "net" in which case the cursor "MOVEs" rather than "DRAWs." Statements 2200-2540 perform the plotting in the "spike" format. Statements 2300-2480 draw the coordinates of individual points, and statements 2490-2520 draw a diagonal cross to emphasize the point.

Statements 2570-2710 print information about the plot. Statement 2760 changes the value of Q8 (plot format flag). If the "current" value of Q8 is 1, then statement 2760 returns 2 and vice versa: since Q8=1, (Q8$<>$1)=0 and (Q8$<>$2)=1, consequently Q8 becomes 2.

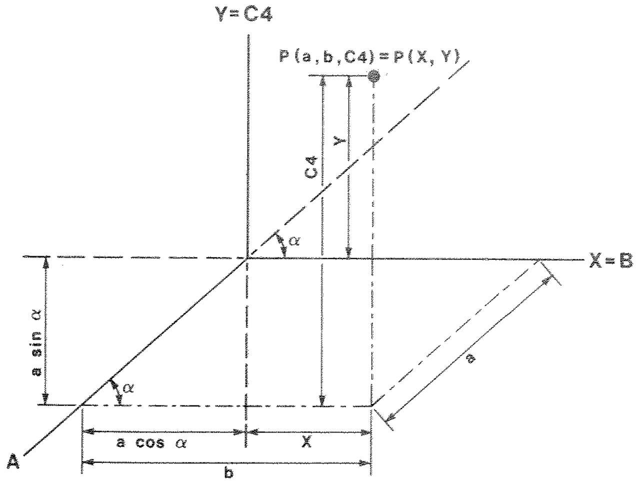


Fig. 1A. Transformation of three-dimensional coordinates A, B, C4 to planar coordinates X, Y.

PROGRAM SUSQ

```
100 PRINT "**********SUSQ SURFACE************"
110 PRINT "FOR THE TWO COMPARTMENT MODEL"
160 INIT
170 SET DEGREES
180 REM  PROJECTION ANGLE  V
190 V=45
200 PRINT "DEVIATION IN PARAMETER RANGE"
210 INPUT V1
220 PRINT "STEP (DØN'T OVERCROWD IT ABOUT 1/5 OF DEVIATION IS GOOD)"
230 INPUT V2
240 S1=25/(100*V1)
250 PRINT "TIME RANGE Ø---->"
260 INPUT T2
270 T3=INT(T2+1)
280 PRINT "PRINTOUT   YES(1)   NO (2) "
290 INPUT Q9
300 IF Q9=2 THEN 365
310 REM ##############################################################
320 PRINT @37,26:1
330 PRINT @40:"TCM SUSQ SURFACE "
340 PRINT @37,26:Ø
365 Q7=Ø
366 REM Q7 IS A FLAG FOR SUSQ SUBROUTINE
370 PRINT "UPTAKE (3),EXCRETION (2)"
380 INPUT Q1
385 N=Q1+1
390 V5=INT(2*V1/V2+1)
400 DIM K(N),M(N),X(N),Y1(T3),Y3(T3),Y4(T3)
410 REM K() PARAMETERS,M()MODIFIED PARAMETERS,X()PARAMETERS IN SUBROUT.
411 REM Y1()=Y3()+Y4(),Y3() COMP.1,Y4() COMP.2
420 DIM P(V5,V5),Z(V5,V5)
421 REM P() SUSQ ARRAY Z()SCALED SUSQ Y COORDINATES
430 DIM E(Q1),D(Q1,Q1),D1(Q1,Q1),C(Q1),F(Q1)
431 REM E()EIGENVALUES,D()EIGENVECTORS,D1()FINAL COEFFICIENTS
432 REM C()INITIAL VALUES,F()CONSTANTS CALCULATED FROM INITIAL VALUES
440 PRINT "INITIAL AMOUNT: COMP.1,2 (WATER)"
441 INPUT C
442 PRINT "VALUES FOR RATE CONSTANTS "
443 PRINT "K2"
444 INPUT K(1)
445 PRINT "K4"
446 INPUT K(2)
447 PRINT "K3"
448 INPUT K(3)
449 IF Q1=2 THEN 455
450 PRINT "K1"
451 INPUT K(4)
455 X=K
456 M=K
460 IF Q9=2 THEN 500
465 PRINT @37,26:1
470 PRINT @40:"PARAMETERS K2,K4,K3,(K1)"
480 PRINT @40:K
485 PRINT @40:"INITIAL AMOUNTS COMP.1,COMP.2, (WATER)"
490 PRINT @40:C
500 GOSUB 3000
510 Q7=1
520 IF Q9=2 THEN 580
530 PRINT @40:"FUNCTION VALUES"
540 PRINT @40:Y1
550 STOP
560 PRINT @40:"ERROR IN PARAMETER                      SUSQ"
570 REM ##############################################################
580 REM SUSQ CALCULATION ROUTINE
590 PRINT "SELECT TWO PARAMETERS FOR VARIATION"
595 PRINT "1-->K2,2-->K4,3-->K3,4-->K1. 4 NOT AVAILABLE FOR EXCRETION"
600 INPUT I9,J9
610 IF Q9=2 THEN 640
620 PRINT @40:I9,J9
630 PRINT @40:"=================================================="
640 PRINT "REMAINING PARAMETERS AS ENTERED (1) OR DIFFERENT (2)"
650 INPUT Q8
660 IF Q8=1 THEN 730
670 FOR I1=1 TO N
680 IF I1=I9 OR I1=J9 THEN 720
690 PRINT "INPUT VALUE FOR K(";I1;")"
700 INPUT K(I1)
710 M(I1)=K(I1)
720 NEXT I1
730 C1=Ø
740 REM ROW COUNTER
750 FOR I2=-V1 TO V1 STEP V2
760 C1=C1+1
770 C2=Ø
780 REM COLUMN COUNTER
790 IF Q9=2 THEN 810
```

PROGRAM SUSQ (cont'd)

```
800 PRINT @40:I2
810 FOR I3=-V1 TO V1 STEP V2
820 C2=C2+1
830 M(I9)=K(I9)+I2*K(I9)
840 M(J9)=K(J9)+I3*K(J9)
850 REM SUSQ SUBROUTINE
860 S=0
865 X=M
870 GOSUB 3000
875 FOR I1=0 TO T2
880 S=S+(Y1(I1+1)-Y3(I1+1)-Y4(I1+1))↑2
885 NEXT I1
900 P(C1,C2)=S
910 IF Q9=2 THEN 930
920 PRINT @40:"                 ";I3,S
930 NEXT I3
940 NEXT I2
950 REM ############################################################
960 REM FINDING MAXIMUM SUSQ (PRELIMINARY SCALING)
970 P1=0
980 FOR I=1 TO V5
990 FOR J=1 TO V5
1000 IF P(I,J)<P1 THEN 1040
1010 P1=P(I,J)
1020 C8=I
1030 C9=J
1040 NEXT J
1050 NEXT I
1060 IF Q9=2 THEN 1080
1070 PRINT @40:"MAXIMUM ";P1,"AT ";C8,C9
1080 P1=30/P1
1090 IF Q9=2 THEN 1110
1100 PRINT @40:"P1 FACTOR ";P1
1110 PRINT @37,26:0
1120 REM ###########################################################
1130 REM SCALING FOR DISPLAY
1140 C1=0
1150 P8=0
1160 P9=0
1170 P7=0
1180 P6=0
1185 Q3=0
1190 PRINT @37,26:1
1200 FOR I=-V1 TO V1 STEP V2
1210 C1=C1+1
1220 C2=0
1230 V7=-100*I*S1*SIN(V)
1240 FOR J=-V1 TO V1 STEP V2
1250 C2=C2+1
1260 REM V6=SCALED TO 50;V7=Y COORDINATE
1270 V6=P(C1,C2)*P1
1280 Z(C1,C2)=V6+V7
1290 IF Q9=2 THEN 1310
1300 PRINT @40:C1;C2,V6,V7,Z(C1,C2)
1310 IF Z(C1,C2)<49 THEN 1390
1315 Q3=1
1320 P8=Z(C1,C2)
1330 P7=C1
1340 P6=C2
1350 IF P8<P9 THEN 1390
1360 P9=P8
1370 P7=C1
1380 P6=C2
1390 NEXT J
1400 NEXT I
1410 IF Q3=0 THEN 1480
1420 PRINT @40:"ADDITIONAL SCALING REQUIRED"
1430 PRINT @40:P7;P6,P8,P9
1440 V7=-100*(V1+(P7-1)*V2)*S1*SIN(V)
1450 P1=30/(P9-V7)
1460 PRINT @40:"V7= ";V7,"NEW P1= ";P1
1470 GO TO 1140
1480 PRINT @37,26:0
1490 PRINT "DISPLAY AS SPIKES (1) OR AS NET (2) "
1500 INPUT Q8
1510 GO TO 1530
1520 REM PLOTTING OF AXES
1530 PAGE
1540 MOVE 65,50
1550 SCALE 1,1
1560 MOVE 0,0
1570 DRAW 50,0
1580 MOVE 0,0
1590 DRAW 0,50
```

```
1600 MOVE Ø,Ø
1610 DRAW -5Ø*SIN(V),-5Ø*COS(V)
1620 MOVE Ø,Ø
1630 FOR I=1 TO 1Ø
1640 RDRAW -4,Ø
1650 RMOVE -1,Ø
1660 NEXT I
1670 MOVE Ø,Ø
1680 FOR I=1 TO 1Ø
1690 RDRAW 3*SIN(V),3*COS(V)
1700 RMOVE 2*SIN(V),2*COS(V)
1710 NEXT I
1720 REM ==============================================================
1730 REM LABELS ON AXES
1740 MOVE -55*SIN(V),-55*COS(V)
1750 PRINT "A"
1760 MOVE 5Ø,-5
1770 PRINT "B"
1780 MOVE 5,49
1790 PRINT "SUSQ"
1800 REM ===========================================================
1810 IF Q8=1 THEN 22ØØ
1820 REM PLOTTING OF NET
1830 REM FIRST DIRECTION
1840 C1=Ø
1850 FOR I=-V1 TO V1 STEP V2
1860 C1=C1+1
1870 C2=Ø
1880 A=1ØØ*I*S1
1890 FOR J=-V1 TO V1 STEP V2
1900 C2=C2+1
1910 B=1ØØ*J*S1
1920 X1=B-A*COS(V)
1930 Y=Z(C1,C2)
1940 IF C2<>1 THEN 197Ø
1950 MOVE X1,Y
1960 GO TO 198Ø
1970 DRAW X1,Y
1980 NEXT J
1990 NEXT I
2000 REM SECOND DIRECTION
2010 C2=Ø
2020 FOR J=-V1 TO V1 STEP V2
2030 C2=C2+1
2040 C1=Ø
2050 B=1ØØ*J*S1
2060 FOR I=-V1 TO V1 STEP V2
2070 C1=C1+1
2080 A=1ØØ*I*S1
2090 X1=B-A*COS(V)
2100 Y=Z(C1,C2)
2110 IF C1<>1 THEN 214Ø
2120 MOVE X1,Y
2130 GO TO 215Ø
2140 DRAW X1,Y
2150 NEXT I
2160 NEXT J
2170 REM ===============================================================
2180 GO TO 256Ø
2190 REM ==================================================================
2200 REM PLOTTING OF POINTS AS SPIKES
2210 C1=Ø
2220 FOR I=-V1 TO V1 STEP V2
2230 C1=C1+1
2240 C2=Ø
2250 A=1ØØ*I*S1
2260 FOR J=-V1 TO V1 STEP V2
2270 C2=C2+1
2280 B=1ØØ*J*S1
2290 Y=Z(C1,C2)
2300 A1=A
2310 B1=B
2320 C3=Y
2330 B=Ø
2340 C4=Ø
2350 GOSUB 281Ø
2360 MOVE X1,Y
2380 B=B1
```

```
2390 GOSUB 2810
2400 DRAW X1,Y
2410 A=0
2420 GOSUB 2810
2430 DRAW X1,Y
2440 A=A1
2450 GOSUB 2810
2460 MOVE X1,Y
2470 Y=C3
2480 DRAW X1,Y
2490 MOVE X1-0.3,Y-0.3
2500 DRAW X1+0.3,Y+0.3
2510 MOVE X1-0.3,Y+0.3
2520 DRAW X1+0.3,Y-0.3
2530 NEXT J
2540 NEXT I
2550 REM ============================================================
2560 REM PRINTING INFORMATION
2570 PRINT @37,26:0
2580 MOVE -20,-35
2590 LIST 500
2600 MOVE 5,-39
2610 PRINT "A=K(";I9;")= ";K(I9),"B=K(";J9;")= ";K(J9)
2620 MOVE 10,-42
2630 PRINT "RANGE ";-V1;" TO ";V1;" STEP ";V2
2640 MOVE 10,-45
2650 PRINT "TIME RANGE  0 TO ";T2
2660 MOVE -65,47
2670 PRINT "OTHER PARAMETERS"
2680 FOR I=1 TO N
2690 IF I=I9 OR I=J9 THEN 2710
2700 PRINT "K(";I;")=";K(I)
2710 NEXT I
2720 PRINT "ANOTHER PLOT FORMAT (1),END (2)"
2730 INPUT Q7
2740 IF Q7=2 THEN 2780
2750 PAGE
2760 Q8=(Q8<>1)+(Q8<>2)*2
2770 GO TO 1560
2780 END
2790 REM ------------------------------------------------------------
2800 REM TRANSFORMATION TO X/Y COORDINATES
2810 X1=B-A*COS(V)
2820 Y=C4-A*SIN(V)
2830 RETURN
3000 REM ===========================================================
3010 REM DIFF EQS.SOLVING ROUTINE
3170 A=-X(1)-X(2)-X(3)
3180 A1=SQR(A↑2-4*X(1)*X(2))
3190 E(1)=0.5*(A-A1)
3200 E(2)=0.5*(A+A1)
3210 IF Q1=2 THEN 3230
3220 E(3)=0
3230 D(1,1)=X(2)/(X(1)+X(3)+E(1))
3240 D(2,1)=1
3250 IF Q1=2 THEN 3270
3260 D(3,1)=0
3270 D(1,2)=X(2)/(X(1)+X(3)+E(2))
3280 D(2,2)=1
```

```
3290 IF Q1=2 THEN 3340
3300 D(3,2)=0
3310 D(1,3)=X(4)/X(1)
3320 D(2,3)=X(3)*X(4)/(X(1)*X(2))
3330 D(3,3)=1
3340 D3=D(1,1)-D(1,2)
3350 IF Q1=2 THEN 3400
3360 F(1)=(C(1)+C(3)*D(1,2)*D(2,3)-C(3)*D(1,3)-C(2)*D(1,2))/D3
3370 F(2)=(C(2)*D(1,1)+C(3)*D(1,3)-C(3)*D(2,3)*D(1,1)-C(1))/D3
3380 F(3)=(C(3)*D(1,1)-C(3)*D(1,2))/D3
3390 GO TO 3420
3400 F(1)=(C(1)-C(2)*D(1,2))/D3
3410 F(2)=(C(2)*D(1,1)-C(1))/D3
3420 FOR I=1 TO Q1
3430 FOR J=1 TO Q1
3440 D1(I,J)=F(J)*D(I,J)
3450 NEXT J
3460 NEXT I
3500 Y3=0
3510 Y4=0
3520 FOR J1=0 TO T2
3530 FOR J2=1 TO Q1
3540 Y3(J1+1)=Y3(J1+1)+D1(1,J2)*EXP(E(J2)*J1)
3550 Y4(J1+1)=Y4(J1+1)+D1(2,J2)*EXP(E(J2)*J1)
3560 NEXT J2
3570 NEXT J1
3580 IF Q7=1 THEN 3610
3581 REM TRUE VALUE/SUSQ BRANCH
3600 Y1=Y3+Y4
3610 RETURN
```

PROGRAM SUSQ (cont'd)

```
TCM SUSQ SURFACE
PARAMETERS K2,K4,K3,(K1)


 5                  Ø.5                  1                  1ØØ


INITIAL AMOUNTS COMP.1,COMP.2, (WATER)


 Ø                  Ø                    Ø.1


FUNCTION VALUES


 -7.1Ø5427358E-15  2.89754152964        3.94416464349      4.63639335763
 5.Ø9553593668     5.4ØØØ7974Ø28        5.6Ø2Ø8ØØ2437      5.736Ø6441116
 5.82493466193     5.88388124267        5.92297980883


ERROR IN PARAMETER                     SUSQ
 1                  2
===========================================
 -Ø.5
                 -Ø.5                   586.924232142
                 -Ø.25                  345.4681339Ø2
                 Ø 215.7Ø14816
                 Ø.25                   141.1982297Ø8
                 Ø.5                    96.Ø191133971
 -Ø.25
                 -Ø.5                   153.75Ø716327
                 -Ø.25                  64.3628478548
                 Ø 25.8869471537
                 Ø.25                   9.3699692483
                 Ø.5                    2.9Ø282961428
 Ø
                 -Ø.5                   35.8Ø54451671
                 -Ø.25                  5.7343182603l
                 Ø 1.ØØ9741959E-27
                 Ø.25                   2.82Ø91973372
                 Ø.5                    8.45951166816
 Ø.25
                 -Ø.5                   5.Ø47584Ø835
                 -Ø.25                  1.89Ø5738Ø957
                 Ø 9.88918242Ø64
                 Ø.25                   2Ø.2146119243
                 Ø.5                    3Ø.192595Ø261
 Ø.5
                 -Ø.5                   3.46395791923
                 -Ø.25                  13.6598833662
                 Ø 27.8675857532
                 Ø.25                   41.23Ø658393
                 Ø.5                    52.7351445569
MAXIMUM 586.924232142                  AT 1               1
PI FACTOR Ø.Ø511139229854
 1 1                3Ø                  17.6776695297      47.6776695297
 1 2                17.65823159Ø2       17.6776695297      35.3359Ø11199
 1 3                11.Ø253489183       17.6776695297      28.7Ø3Ø18448
 1 4                7.21719543897       17.6776695297      24.8948649686
 1 5                4.9Ø791356731       17.6776695297      22.5855830Ø97
 2 1                7.8588Ø22733        8.83883476483      16.69763Ø381
 2 2                3.28983764838       8.83883476483      12.1286724132
 2 3                1.32318342314       8.83883476483      1Ø.162Ø18188
 2 4                Ø.478935886534      8.83883476483      9.3177Ø65137
 2 5                Ø.148375ØØ9344      8.83883476483      8.9872Ø977418
 3 1                1.83Ø15676673       Ø                  1.83Ø15676673
 3 2                Ø.2931Ø35Ø1931      Ø                  Ø.2931Ø35Ø1931
 3 3                5.16118727lE-29     Ø                  5.16118727lE-29
 3 4                Ø.14418827Ø17       Ø                  Ø.14418827Ø17
 3 5                Ø.4323988279Ø1      Ø                  Ø.4323988279Ø1
 4 1                Ø.258ØØ18241Ø6     -8.83883476483     -8.58Ø83294Ø73
 4 2                Ø.Ø96634644lØØ8    -8.83883476483     -8.7422ØØ12Ø73
 4 3                Ø.5Ø5474908638     -8.83883476483     -8.33335985619
 4 4                1.Ø33248117Ø8      -8.83883476483     -7.8Ø558664775
 4 5                1.5432619769       -8.83883476483     -7.29557278794
 5 1                Ø.177Ø564783Ø9     -17.6776695297     -17.5ØØ613Ø514
 5 2                Ø.698210226371     -17.6776695297     -16.97945930333
 5 3                1.42442163198      -17.6776695297     -16.2532478977
 5 4                2.1Ø746Ø69774      -17.6776695297     -15.57Ø2088319
 5 5                2.6955ØØ11751      -17.6776695297     -14.9821694122
```