A PC COMPUTER PROGRAM MINIMIX FOR THE CALCULATION OF THE STABILITY CONSTANTS OF $M_qL_pL'_{p'}$, TYPE MIXED OR THE MIXTURE OF M_qL_p and $M_{q'}L'_{p'}$, COMPLEXES FROM SPECTROPHOTOMETRIC MEASUREMENTS

F. GAIZER*

Institute of Inorganic and Analytical Chemistry, Attila József University,

Dóm tér 7, H–6720 Szeged, Hungary

and

H. B. SILBER

San José State University, School of Chemistry,

San José, California, 95192, USA

(*Received May 21, 1990*)

THE PC COMPUTER PROGRAM MINIMIX HAS BEEN CONSTRUCTED IN BASIC AND FORTRAN PROGRAMING LANGUAGE FOR THE CALCULATION OF STABILITY CONSTANTS AND MOLAR ABSORPTIVITIES OF THE $M_qL_pL'_{p'}$, TYPE MIXED OR THE MIXTURE OF M_qL_p AND $M_{q'}L'_{p'}$, COMPLEXES FROM SPECTROPHOTOMETRIC MEASUREMENTS. BY THE FORTRAN VERSION OF THIS PROGRAM THE EQUILIBRIA OF THE $M_qL_pL_p, H_r$ PROTONATED/DEPROTONATED COMPLEXES CAN ALSO BE TREATED.

Introduction

In our previous papers we published desk computer programs for 16 Kbyte computers to evaluate stability constants from potentiometric [1] and spectrophotometric [2] measurements. In the present paper we introduce a more complex program which is capable of calculating the optimum value of stability constants and molar absorp–

* Present permanent address: Department of Chemistry, Janus Pannonius University, H–7624 Pécs, Ifjúság u. 6., Hungary

tivities of mixed ligand complexes of type $M_qL_pL'_{p'}$ from spectrophotometric measurements. The program also applies to mixtures of complexes such as M_qL_p and $M_{q'}L'_{p'}$, but it cannot be used for metals bound to only a single ligand (M_qL_p alone). The present program MINIMIX written in BASIC programming language differs from the previous ones not only in the problem to be solved, but in its capability of evaluating measurements at any number of wavelengths. By the FORTRAN version of this program, which is available upon request, the equilibria of mixed ligand complexes of type $M_qL_pL'_{p'}H_r$, *i.e.*, protonated (+r) or hydroxo (−r) complexes, can be treated too.

Fundamentals

For the formation of mixed ligand complexes the following equilibria can be written (the charges on the components are omitted for simplicity):

$$q\,M + p\,L + p'\,L' \rightleftharpoons M_qL_pL'_{p'} \quad \text{or}$$
$$q\,M + p\,L + p'\,L' + r\,H^+ \rightleftharpoons M_qL_pL'_{p'}H_r\,.$$

For these equilibria the following cumulative formation constant can be written and are used in this program:

$$\beta = [M_qL_pL'_{p'}H_r] \cdot [M]^{-q} \cdot [L]^{-p} \cdot [L']^{-p'} \cdot [H^+]^{-(r)}$$

In the present BASIC program the last term is not included.

Assuming the validity of BEER's law, the absorbance (A) of a solution in 1 cm path length can be written as the sum of the absorbances of each component:

$$A = \epsilon_M \cdot [M] + \epsilon_L \cdot [L] + \epsilon_{L'} \cdot [L'] + \sum_j \epsilon_j \cdot c_j,$$

where $c_j = \beta_j \cdot [M]^{q_j} \cdot [L]^{p_j} \cdot [L']^{p'_j} \cdot [H^+]^{(r_j)}$. If we assume the composition of complexes (q, p, p', and r values) formed in the system studied and their stability constants (β values), we can calculate the concentration of free species ([M], [L] and [L'] by solving the mass balance equations written for the total concentrations of components (T_M, T_L and $T_{L'}$) [3], and having these values we can calculate the concentration of the individual c_j complexes. If we assume molar absorptivities for each of the absorbing species too, we can calculate the absorbance (A_c) and compare it with the measured one (A_m) for each solution. If the agreement is not satisfactory, the program adjusts the assumed parameters (β and ϵ values) until satisfactory agreement is reached, or a completely new calculation with another model (with new q, p, p' and r values) is required. Consequently the steps of the evaluation with a model are as follows: First, we assume the stoichiometric coefficients of the complexes formed; second, β and ϵ values are estimated. Third, the program calculates the concentration of the free species and complexes, and from these then calculates the absorbancies.

To calculate the optimum values of the parameters resulting in the minimum sum of the unweighted squares of residuals in absorbances, the same method is used in this program as in MINISPEF [2].

The program

With the exception of renumbering and a few changes detailed later, the portions of the program (lines 260–295) were kept identical with that of MINISPEF [2]. The names of variables and arrays have been kept, only a few new ones had to be introduced. To make the usage of the program easier, we reduced the number of input variables to the minimum, based on our experiences. In order to help user in extending or modifying the program, the arrays and their dimensions have been summarized

in Table I.

In the main program, the following substantial changes have been made:

- It is capable of treating data from measurements made at a maximum of four wavelengths.
- The error in the individual parameters are saved and may be printed out.
- The values of the elements of correction vector appear on the screen. This facilitates following the refinement cycles and the "behavior" of each parameter during refinements.
- On the basis of our systematic examination, we have found it necessary to modify automatically the value of individual increments after each refinement cycle. As an optimum value for the increment, half of the error in each parameter was found. In spite of this modification, it is advisable to try different starting values for increments, if the interaction between the reactants is weak or there are parameters which have only a slight effect on the sum of the square of residuals. Leaving out the T(J) = U7/2 statement in the line 680, no modification of increments is made.

The function of subroutines EQUSOL and PRINT has not been changed, but the former one has been extended to solve equation systems with three unknowns. It seemed to be expedient to locate them in a separate subroutine (CPXES) for the calculation of the concentration of complexes.

Instructions for the use of the program

The percentage distribution of complexes is calculated by the formula

$$\% = \frac{q_j * C_{ij} * 100}{T}, \quad \text{where:}$$

Table I

Name of Array	Chemical notation or Reference	Identifier in the Program	Dimension of the Array
Total concentrations	T_M, T_L, $T_{L'}$	M, L, CZ	MP
Concentration of free species	[M], [L], [L']	A, B, Z	MP
Measured and calculated absorbances	A_m, A_c	E, El	MP x NW
Stoichiometric coefficients of complexes	q, p, p'	Q1, P1, R1	NC
Parameters	log β, ϵ	P	NP
Increments of parameters		T	NP
Errors in parameters	$\sigma(\log\beta)$, $\sigma(\epsilon)$	ER	NP
Complexes	$M_qL_pL'_p$,	C1	MP x NC
Serial No. parameters to be refined		S1	PF
Serial No. of wavelength involved into refinement		SW	NW
Transposed of gradient matrix	G^{T*}	G	PFx(NW x MP)
Product of the gradient matrix and its transposition	$C = G^T \times G^*$	C	PF x PF
Inverse of C	C^{-1*}	C5	PF x PF
Error vector	$y_i - y_i^{o*}$	D	NW x MP
Absorbances calculated by the starting values of parameters	y_i^{o*}	F	NW x MP
Product of transposed gradient matrix and error vector	$G^T \times d^*$	G1	PF
Correction vector	h^*	G2	PF
Cumulative stability constans	β	E2	NC
Molar absorptivites	ϵ	E3	NWx(NC + 3)

Abbreviations: NC: No. of complexes; NP: No. of parameters (= NC + NW * (NC + 3))
NW: No. of wavelengths; MP: No. of measured points; PF: No. of parameters to be refined.
*) Ref. 3.

if N%=1 then T = TOT M, if N%=2 then T = TOT L, if N%=3 then T = TOT L'.

Divisors (X1, X2, X3). To make easier the typing of TOT concentrations of reactants, their multiplied values can be input; *e.g.*, if the concentrations are in mM, then the divisor X1 = 1000, etc. (see line 85).

Factors generating initial values of free species are necessary for solving mass balance equations by iterative methods. These initial values are calculated in the following way:

- $[M] = T_M * FM$, $[L] = T_L * FL$ and $[L'] = T_{L'} * FZ$. (line 840 and 1015).

The input of parameters into the array P must be made in the following order: first by β's, then the respective molar absorptivity of complexes, then ϵ_M, ϵ_L and $\epsilon_{L'}$, for the first, second, etc. wavelength. The serial number of parameters obtained in this way is used in the refinement or search procedure.

Experimental points involved in the calculations: there are options to use only one part of the experimental points, such as with only every second, third, etc. points.

Task: The program can execute three tasks:

1.) Optimizing simultaneously a maximum of 4 parameters given by their serial numbers.

2.) Search for the value of one of the parameters between input limits and by input step.

3.) Search for the value of one of the parameters between limits and steps generated by the actual value of parameter to be searched for. *E.g.*, if the value of the parameter searched for is P and the value of T3 is given for the lower limit of the search, the actual starting value will be T3 * P, etc., for T4 and T5 (line 335).

Note: 1.) The search is carried out only into the direction of higher values.

2.) If the value of the parameter to be searched for is negative, the step–
–factor T4 must be negative, too.

Print: For uneven numbers the output appears only on the screen.

For even numbers it is sent to the printer also.

Printout options: Depending upon the number chosen, various printing options are allowed:

if R% = 1 or 2: Refined parameters and their errors, square of residuals and standard deviation are shown.

if R% = 3 or 4: All the parameters and their errors, square of residuals and standard deviation are shown.

if R% = 5 or 6: The measured and calculated absorbances and the difference between them for the wavelengths involved into the calculations and the percentage distribution of complexes, that of free metal and free ligand are shown.

Wavelength(s) involved into refinements must be given by their input serial number.

Control number (V1): After execution of a task, it is possible to continue the calculations at different points of the programs (see lines 375 and 695). Options:

if V1 = 1: Calculation with new experimental data (from line 20)

if V1 = 2: New model (new Q, P, P', R values, from line 120)

if V1 = 3: Calculations with another points (from line 175)

if V1 = 4: New task (from line 265)

if V1 = 5: New refinement cycle (from line 385)

if V1 = 6: End of calculations.

Acknowledgement

This work was done under the Grant AX–659. The Robert A. Welch Foundation of Houston, Texas. One of the authors (F.G.) expresses his gratitude to the Foundation for the financial support.

References

[1] *Gaizer, F., A. Puskás:* Talanta, *28*, 565 (1981).

[2] *Gaizer, F., A. Puskás:* Talanta, *28*, 925 (1981).

[3] *Gaizer, F., M. Máté:* Acta Chim. Acad. Sci. Hung., *103*, 355 (1980).

```
0 REM MINIMIX-MW4
5 DIM A(50),B(50),C(4,4),C1(50,8),C5(4,4),CZ(50),D(200),E(50,8),E1(50,8)
10 DIM E2(8),E3(4,11),ER(52),F(200),G(4,200),G1(4),G2(4),L(50),M(50),P(52)
15 DIM P1(8),Q1(8),R1(8),S1(4),SW(4),T(52),Z(50):ZI=LOG(10):TI=1/ZI:E5=1000
20 PRINT "? # OF MSD POINTS":INPUT N1:PRINT "? # OF PARAMETERS":INPUT N2
25 PRINT "? # OF CPXES":INPUT N3:PRINT "? PERCENTAGE DISTRIBUTION RELATED TO
    1: TOT M,   2: TOT L,   3: TOT L'   -   SEE INSTRUCTIONS":INPUT N4%
30 PRINT "PRINTING OF MSD. DATA ?  -  NO: 0,  YES: 1":INPUT R%
35 PRINT "? DIVISOR OF TOT M TO BE INPUT":INPUT X1
40 PRINT "? DIVISOR OF TOT L TO BE INPUT":INPUT X2
45 PRINT "? DIVISOR OF TOT L' TO BE INPUT":INPUT X3
50 PRINT "? FACTOR GENERATING [M] FROM TOT M":INPUT FM
55 PRINT "? FACTOR GENERATING [L] FROM TOT L":INPUT FL
60 PRINT "? FACTOR GENERATING [L'] FROM TOT L'":INPUT FZ
65 PRINT "? # OF WAVELENGTHS":INPUT NW
70 FOR I=1 TO N1:PRINT  "? TOT M";I:READ M(I):PRINT "? TOT L";I:READ L(I)
75 PRINT "? TOT L'";I:READ CZ(I)
80 FOR J=1 TO NW:PRINT "? ABSORBANCE FOR WAVELENGTH #";J:READ E(I,J):NEXT J
85 M(I)=M(I)/X1:L(I)=L(I)/X2:CZ(I)=CZ(I)/X3:V2=E5
90 NEXT I
95 IF R%=0 THEN 105
100 LPRINT "        MEASURED DATA, CONCNS IN MMOL, AND ABSORBANCES":LPRINT
105 PRINT "        MEASURED DATA, CONCNS IN MMOL, AND ABSORBANCES":PRINT
110 FOR I=1 TO N1:PRINT USING "####.### ";I,M(I)*V2,L(I)*V2,CZ(I)*V2,E(I,1),E(I
2),E(I,3),E(I,4):NEXT I:IF R%=0 THEN 120
115 FOR I=1 TO N1:LPRINT USING "####.### ";I,M(I)*V2,L(I)*V2,CZ(I)*V2,E(I,1),E(
,2),E(I,3),E(I,4):NEXT I
120 FOR I=1 TO N3:PRINT "? Q FOR CPX #";I:INPUT Q1(I)
125 PRINT "? P FOR CPX #";I:INPUT P1(I):PRINT "? P' FOR CPX #";I:INPUT R1(I)
130 NEXT I
135 FOR I=1 TO N3:PRINT "? LOG BETA OF CPX #";I:INPUT P(I):
    PRINT "? ITS INCREMENT":INPUT T(I):NEXT I
140 I=N3+1:FOR J=1 TO NW:PRINT "WAVELENGTH #";J:FOR J1=1 TO N3
145 PRINT "? MOLAR ABSORPTIVITY OF CPX #";J1:INPUT P(I):
    PRINT "? ITS INCREMENT":INPUT T(I):I=I+1:NEXT J1
150 PRINT "? MOLAR ABSORPTIVITY OF METAL": INPUT P(I):
    PRINT "? ITS INCREMENT":INPUT T(I):I=I+1
155 PRINT "? MOLAR ABSORPTIVITY OF THE FIRST LIGAND"
160 INPUT P(I):PRINT "? ITS INCREMENT":INPUT T(I):I=I+1
165 PRINT "? MOLAR ABSORPTIVITY OF THE SECOND LIGAND":INPUT P(I)
170 PRINT "? ITS INCREMENT":INPUT T(I):I=I+1:NEXT J
175 PRINT "? SERIAL # OF THE FIRST EXPT. POINT TO BE EVALUATED":INPUT K
180 PRINT "? SERIAL # OF THE LAST EXPT. POINT TO BE EVALUATED":INPUT V
185 PRINT "? STEP BETWEEN THE EXPT. POINTS":INPUT N:T1=1:V2=1:MK=1
260 REM ************************************************************
265 PRINT "?  TASK;  OPTIONS:   0: REFINEMENT;  1: SEARCH WITH INPUT DATA;
                               2: SEARCH WITH GENERATED DATA":INPUT F
270 PRINT "? PRINT FORMAT;  OPTIONS 0-6,  SEE INSTRUCTIONS":INPUT R%
275 PRINT "? # OF WAVELENGTHS TO BE INVOLVED INTO CALCULATIONS":INPUT NWF
280 FOR LW=1 TO NWF:PRINT "? SERIAL # OF WAVELENGTH #";LW;"INVOLVED INTO CALCULA
TIONS":INPUT SW(LW):NEXT LW
285 GOSUB 830:GOSUB 1280:U1=U:IF F=0 THEN 385
```

```
290 REM *** SEARCH PROCEDURE FOR PARAMETER VALUE **********************
295 PRINT "? STARTING VALUE OR FACTOR OF PARAMETER TO BE SEARCHED FOR":INPUT T3
300 PRINT "? STEP VALUE OR FACTOR OF PARAMETER TO BE SEARCHED FOR":INPUT T4
305 PRINT "? UPPER LIMIT OR FACTOR OF PARAMETER TO BE SEARCHED FOR":INPUT T5
310 PRINT "? SERIAL # OF PARAM. TO BE SEARCHED FOR":INPUT T6
315 PRINT "? PRINT FORMAT, OPTIONS 0-6, SEE INSTRUCTIONS":INPUT R%
320 PRINT "? # OF STEPS AFTER U-MINIMUM":INPUT T7
325 PRINT "? WHERE TO GO AFTER EXECUTION THIS TASK; OPTIONS 1-6, SEE INSTRUCTION
S":INPUT V1
330 IF F=1 THEN 340
335 T3=P(T6)*T3:T4=P(T6)*T4:T5=P(T6)*T5
340 P(T6)=T3:V2=T6:GOSUB 830:GOSUB 1280:T8=H:T9=P(T6):GOTO 360
345 V2=T6:GOSUB 830:GOSUB 1280:IF H>T8 THEN 355
350 T8=H:T9=P(T6):GOTO 360
355 T7=T7-1:IF T7=0 THEN 365
360 P(T6)=P(T6)+T4:IF P(T6)<=T5 THEN 345
365 P(T6)=T9:PRINT USING "THE OPTIMUM VALUE OF PARAMETER : -########.######";T9
370 PRINT USING "THE SERIAL NUMBER OF PARAMETER :                ###";T6:V2=T6
375 ON V1 GOTO 20,120,175,265,385:END:REM ******************************
380 REM *** REFINEMENT PROCEDURE FOR PARAMETER VALUES ******************
385 PRINT "*** REFINEMENT FOLLOWS !***"
390 PRINT "? # OF PARAMETERS TO BE REFINED":INPUT F1:F2=1
395 PRINT "? WHERE TO GO AFTER EXECUTION THIS TASK; OPTIONS 1-6, SEE INSTRUCTION
S":INPUT V1
400 PRINT "PRINT FORMAT;  OPTIONS 0-6,  SEE INSTRUCTIONS":INPUT R%
405 FOR J=1 TO F1: C(F1,F1)=0:C5(F1,F1)=0:NEXT J
410 FOR J=1 TO F1:PRINT "? SERIAL # OF PARAMETER #";J;"TO BE REFINED":INPUT S1(J
):NEXT J
415 REM *** ERROR VECTOR, D ***
420 KW=0:FOR NN=1 TO NWF:LW=SW(NN):FOR I=K TO V STEP N:KW=KW+1:D(KW)=E(I,LW)-E1(
I,LW):F(KW)=E1(I,LW):NEXT I:NEXT NN
425 REM *** G MATRIX ***
430 FOR K1=1 TO F1:J=S1(K1):P(J)=P(J)+T(J):T6=J:V2=J:GOSUB 830:KW=0:FOR NN=1 TO
NWF:LW=SW(NN)
435 FOR I=K TO V STEP N:KW=KW+1:G(K1,KW)=(E1(I,LW)-F(KW))/T(J):NEXT I:NEXT NN
440 P(J)=P(J)-T(J):V2=1:NEXT K1
445 REM *** C MATRIX ***
450 FOR K1=1 TO F1:FOR K2=1 TO F1:C(K1,K2)=0:FOR I=1 TO KW
455 C(K1,K2)=C(K1,K2)+G(K1,I)*G(K2,I): NEXT I
460 C(K2,K1)=C(K1,K2):NEXT K2:NEXT K1
465 REM *** INVERSION OF MATRIX C INTO C5 ***
470 FOR I=1 TO F1:FOR K1=1 TO F1:C5(I,K1)=0:C5(K1,I)=0:NEXT K1:C5(I,I)=1
475 NEXT I:C5(F1,F1)=1:FOR K1=1 TO F1:V3=C(K1,K1):FOR I=1 TO F1
480 C(K1,I)=C(K1,I)/V3:C5(K1,I)=C5(K1,I)/V3:NEXT I:FOR L=1 TO F1
485 IF L-K1=0 THEN 500
490 V3=C(L,K1):FOR I=1 TO F1
495 C(L,I)=C(L,I)-V3*C(K1,I):C5(L,I)=C5(L,I)-V3*C5(K1,I):NEXT I
500 NEXT L:NEXT K1
505 REM *** MULTIPL. OF ERROR VECT. AND G-TRANSP. ***
510 FOR I=1 TO F1:S=0:FOR J=1 TO V STEP N:S=S+G(I,J)*D(J):NEXT J:G1(I)=S:NEXT I
```

```
515 REM *** CORRECTION VECTOR, G2 ***
520 FOR I=1 TO F1:S=0:FOR J=1 TO F1
525 S=S+C5(I,J)*G1(J):NEXT J:G2(I)=S:NEXT I:A6=1:I6=1
530 FOR K1 = 1 TO F1:J=S1(K1):IF J<=N3 THEN 550
535 F3=P(J)+G2(K1)
540 IF F3>0 THEN 550
545 G2(K1)=-P(J)
550 NEXT K1
555 IF F2=0 THEN 570
560 PRINT "U1=",U1
565 K6%=R%:R%=0:I6=0:K7=1:A6=.5:GOTO 575
570 PRINT "REFINED PARAMETERS:":PRINT
575 FOR K1=1 TO F1:J=S1(K1):P(J)=P(J)+G2(K1)*A6:NEXT K1
580 IF I6<1 THEN 610
585 FOR K1=1 TO F1:J=S1(K1)
590 IF J<=N3 THEN 605
595 IF P(J)>0 THEN 605
600 P(J)=0
605 NEXT K1
610 V2=S1(F1):GOSUB 830
615 IF I6>0 THEN 680
620 IF K7=0 THEN 635
625 U2=U:PRINT "U2=",U2
630 K7=0:GOTO 575
635 U3=U:PRINT "U3=",U3:R%=K6%:U4=U1-2*U2+U3:IF U4>0 THEN 655
640 A6=1:IF U3>U1 THEN 650
645 A6=1:GOTO 660
650 A6=0:PRINT "CONCAVE ALFA, IS MADE EQUAL WITH",A6:GOTO 665
655 U4=(U1-U3)/(4*U4):A6=.5+U4
660 PRINT "ALFA=",A6
665 A6=1-A6:I6=1:IF ABS(A6)<3 THEN 675
670 A6=2
675 A6=-A6:GOTO 570
680 U1=U:FOR I=1 TO F1:J=S1(I):U7=SQR(ABS(C5(I,I)))*H:ER(J)=U7:T(J)=U7/2
685 NEXT I:GOSUB 1280
690 PRINT "CORR. VECTOR:":FOR J=1 TO F1:PRINT USING" +##.##### ";G2(J),J:NEXT J
695 ON V1 GOTO 20,120,175,265,385:END:REM ****************************
830 REM EQUSOLV ********************************************************
835 FOR I=1 TO N3:E2(I)=EXP(ZI*P(I)):NEXT I:IK=N3:FOR JJ=1 TO NW:FOR II=1 TO N3+
3:IK=IK+1:E3(JJ,II)=P(IK):NEXT II:NEXT JJ:U=0:S2=0:TOL=.002
840 MK=MK-1:A=L(K)*FL:B=M(K)*FM:Z=CZ(K)*FZ
845 FOR I=K TO V STEP N:IF V2>N3 GOTO 985
850 KK=0:MKK=0:IX=0:ITT=0:CA=L(I):CB=M(I):CZ=CZ(I):AT=CA*TOL:BT=CB*TOL:ZT=CZ*TOL
:TYA=AT/1000:TYB=BT/1000:TYZ=ZT/1000:ITER=0
855 IF MK<0 THEN GOTO 870
860 IF (MK >=0) AND (I=K) THEN GOTO 1000
865 IF (MK>=0) AND (I>K) THEN GOTO 875
870 A=A(I):B=B(I):Z=Z(I)
875 MKK=MKK+1:IF MKK>30 THEN GOTO 1000
880 NRM=1:GOSUB 1165
885 YA=CASZ-CA:YB=CBSZ-CB:YZ=CZSZ-CZ:AYA=ABS(YA):AYB=ABS(YB):AYZ=ABS(YZ)
890 IF (AYA>AT) THEN GOTO 905
895 IF (AYB>BT) THEN GOTO 905
```

```
900 IF AYZ<=ZT THEN GOTO 980
905 IF AYA<TYA THEN YA=0
910 IF AYB<TYB THEN YB=0
915 IF AYZ<TYZ THEN YZ=0
920 AY=0-YA:BY=0-YB:ZY=0-YZ
925 DET=DA*DB*DZ+DBA*DZB*DAZ+DZA*DAB*DBZ-DZA*DB*DAZ-DZB*DBZ*DA-DZ*DBA*DAB
930 DT1=AY*DB*DZ+BY*DZB*DAZ+ZY*DAB*DBZ-ZY*DB*DAZ-DZB*DBZ*AY-DZ*BY*DAB
935 DT2=DA*BY*DZ+DBA*ZY*DAZ+DZA*AY*DBZ-DZA*BY*DAZ-ZY*DBZ*DA-DZ*DBA*AY
940 DT3=DA*DB*ZY+DBA*DZB*AY+DZA*DAB*BY-DZA*DB*AY-DZB*BY*DA-ZY*DBA*DAB
945 A=A+DT1/DET:IF A<0 THEN GOTO 1000
950 IF A>CA THEN GOTO 1000
955 B=B+DT2/DET:IF B<0 THEN GOTO 1000
960 IF B>CB THEN GOTO 1000
965 Z=Z+DT3/DET:IF Z<0 THEN GOTO 1000
970 IF Z > CZ THEN GOTO 1000
975 GOTO 875
980 A(I)=A:B(I)=B:Z(I)=Z
985 FOR NN=1 TO NWF:LW=SW(NN):E1(I,LW)=B(I)*E3(LW,N3+1)+A(I)*E3(LW,N3+2)+Z(I)*E3
(LW,N3+3)
990 FOR LC=1 TO N3:E1(I,LW)=E1(I,LW)+C1(I,LC)*E3(LW,LC):NEXT LC
995 DE=E(I,LW)-E1(I,LW):S2=S2+1:U=U+DE^2:NEXT NN:GOTO 1155
1000 AT=10*AT/(ITER+1):BT=10*BT/(ITER+1):ZT=10*ZT/(ITER+1):A=A(I):B=B(I):Z=Z(I)
1005 IF ITER=1 THEN GOTO 1020
1010 IF MK<0 THEN GOTO 1020
1015 A=CA*FL:B=CB*FM:Z=CZ*FZ
1020 ITER=ITER+1:DIA=A:DIB=B:DIZ=Z
1025 GA=2:GB=2:GZ=2:NRM=0
1030 GOSUB 1165
1035 ITT=ITT+1:IF ITT>500 THEN GOTO 1150
1040 DE=CA-CASZ:ADE=ABS(DE):IF ADE<=AT THEN GOTO 1070
1045 IF DE<=0 THEN GOTO 1055
1050 DIA=GA*DIA:A=A+DIA:GOTO 1060
1055 GA=.5:DIA=GA*DIA:A=A-DIA
1060 DLA=A/1000:IF DIA>DLA THEN GOTO 1030
1065 GA=2:DIA=A:GOTO 1025
1070 DE=CB-CBSZ:ADE=ABS(DE):IF ADE<=BT THEN GOTO 1100
1075 IF DE<=0 THEN GOTO 1085
1080 DIB=GB*DIB:B=B+DIB:GOTO 1090
1085 GB=.5:DIB=GB*DIB:B=B-DIB
1090 DLB=B/1000:IF DIB>DLB THEN GOTO 1025
1095 GB=2:DIB=B:GOTO 1025
1100 DE=CZ-CZSZ:ADE=ABS(DE):IF ADE<=ZT THEN GOTO 1140
1105 IF DE<=0 THEN GOTO 1115
1110 DIZ=GZ*DIZ:Z=Z+DIZ:GOTO 1125
1115 GZ=.5:DIZ=GZ*DIZ:Z=Z-DIZ
1120 IF DE<0 THEN GOTO 1025
1125 DLZ=Z/1000:IF DIZ>DLZ THEN GOTO 1025
1130 GZ=2:DIZ=Z:GOTO 1030
1135 GOTO 980
1140 IF ITER=2 THEN GOTO 980
1145 AT=ITER*AT/10:BT=ITER*BT/10:ZT=ITER*ZT/10:A(I)=A:B(I)=B:Z(I)=Z:ITT=0:MKK=0:
GOTO 875
1150 PRINT "UNSUCCESSFUL ITERATION!":GOTO 1140
1155 NEXT I:RETURN
1160 RETURN
```

```
1165 REM CPXES FOR MIXED LIGAND CPXES **********************************
1170 CASZ=A:CBSZ=B:CZSZ=Z:IF NRM=0 THEN GOTO 1180
1175 DA=1:DB=1:DZ=1:DAB=0:DAZ=0:DBZ=0
1180 FOR L=1 TO N3:JP=P1(L):JQ=Q1(L):JZ=R1(L)
1185 C1(I,L)=E2(L)*Z^JZ*B^JQ*A^JP:CPX=C1(I,L)
1190 CASZ=CASZ+JP*CPX:CBSZ=CBSZ+JQ*CPX:CZSZ=CZSZ+JZ*CPX:IF NRM=0 GOTO 1205
1195 DA=DA+(CPX/A)*JP*JP:DB=DB+(CPX/B)*JQ*JQ:DZ=DZ+(CPX/Z)*JZ*JZ
1200 DAB=DAB+JP*JQ*CPX:DBZ=DBZ+JQ*JS*CPX:DAZ=DAZ+JP*JS*CPX:NEXT L
1205 IF NRM=0 THEN GOTO 1215
1210 DBA=DAB/A:DAB=DAB/B:DZB=DBZ/B:DBZ=DBZ/Z:DZA=DAZ/A:DAZ=DAZ/Z
1215 RETURN
1280 REM PRINT ***********************************************************
1290 IF R%=0 THEN GOTO 1680
1300 IF F>0 THEN PRINT "VALUE OF PARAM. SEARCHED FOR:  ";P(T6)
1310 IF F>0 AND R%>=2 THEN LPRINT "VALUE OF PARAM. SEARCHED FOR:  ";P(T6)
1320 IF F>0 THEN GOTO 1430
1330 IF R%>2 THEN GOTO 1380
1340 IF F1>0 THEN PRINT "REFINED PARAMETERS AND THEIR ERRORS:"
1350 FOR I=1 TO F1:J=S1(I):PRINT J,P(J),ER(J):NEXT I:IF R%=1 THEN GOTO 1430
1360 IF F1 >0 THEN LPRINT "REFINED PARAMETERS AND THEIR ERRORS:":LPRINT
1370 FOR I=1 TO F1:J=S1(I):LPRINT J,P(J),ER(J):NEXT I:GOTO 1430
1380 PRINT "PARAMETERS AND THEIR ERRORS:"
1390 FOR I=1 TO N2:PRINT USING "  ####.####  ";I,P(I),ER(I):NEXT I
1400 IF R%=3 OR R%=5 THEN GOTO 1430
1410 LPRINT" ":LPRINT "PARAMETERS AND THEIR ERRORS:"
1420 FOR I=1 TO N2:LPRINT USING "  ####.####  ";I,P(I),ER(I):NEXT I
1430 PRINT USING"SQUARE OF RESIDUALS:  .###### ";U
1440 IF R%=2 OR R%=4 THEN LPRINT USING"SQUARE OF RESIDUALS:  .###### ";U
1450 IF R%=6 THEN LPRINT USING"SQUARE OF RESIDUALS:  .###### ";U
1460 H=SQR(U/(S2-F1-1)):PRINT USING"STANDARD DEVIATION:   .###### ";H:PRINT
1470 IF R%=2 OR R%=4 THEN LPRINT " ":LPRINT USING"STANDARD DEVIATION:   .######
";H:LPRINT
1480 IF R%=6 THEN LPRINT :LPRINT USING"STANDARD DEVIATION:   .###### ";H:LPRINT
1490 IF R%<5 THEN GOTO 1680
1500 FOR JJ=1 TO NWF:J=SW(JJ):PRINT " WAVELENGTH # ",J:PRINT:FOR I=K TO V STEP N
:PRINT USING " +##.###  ";E(I,J),E1(I,J),E(I,J)-E1(I,J),I:NEXT I:NEXT JJ
1510 IF R%=5 THEN GOTO 1540
1520 FOR JJ=1 TO NWF:J=SW(JJ):LPRINT " WAVELENGTH # ",J:LPRINT:FOR I=K TO V STEP
 N:LPRINT USING " +##.###  ";E(I,J),E1(I,J),E(I,J)-E1(I,J),I:NEXT I:NEXT JJ
1530 ON N4% GOTO 1540,1590,1640
1540 PRINT "PERCENTAGE DISTRIBUTION OF TOT M IN CPXES AND FREE M"
1550 IF R%=6 THEN LPRINT "PERCENTAGE DISTRIBUTION OF TOT M IN CPXES AND FREE M"
1560 FOR I=K TO V STEP N:PRINT USING "  ###.##  ";C1(I,1)*100/M(I),C1(I,2)*100/M
(I),C1(I,3)*100/M(I),B(I)*100/M(I),I:NEXT I
1570 IF R%=5 THEN GOTO 1680
1580 FOR I=K TO V STEP N:LPRINT USING "  ###.##  ";C1(I,1)*100/M(I),C1(I,2)*100/
M(I),C1(I,3)*100/M(I),B(I)*100/M(I),I:NEXT I:GOTO 1680
1590 PRINT "PERCENTAGE DISTRIBUTION OF TOT L IN CPXES"
1600 IF R%=6 THEN LPRINT "PERCENTAGE DISTRIBUTION OF TOT L IN CPXES"
1610 FOR I=K TO V STEP N:PRINT USING "  ###.##  ";C1(I,1)*100/L(I),C1(I,2)*100/L
(I),A(I)*100/L(I),I:NEXT I
```

```
1620 IF R%=5 THEN GOTO 1680
1630 FOR I=K TO V STEP N:LPRINT USING "  ###.##  ";C1(I,1)*100/L(I),C1(I,2)*100/
L(I),A(I)*100/L(I),I:NEXT I:GOTO 1680
1640 PRINT "PERCENTAGE DISTRIBUTION OF TOT L' IN CPXES"
1650 IF R%=6 THEN PRINT "PERCENTAGE DISTRIBUTION OF TOT L' IN CPXES"
1660 FOR I=K TO V STEP N:PRINT USING "  ###.##  ";C1(I,1)*100/CZ(I),C1(I,2)*100/
CZ(I),Z(I)*100/CZ(I),I:NEXT I:IF R%=5 THEN GOTO 1680
1670 FOR I=K TO V STEP N:LPRINT USING "  ###.##  ";C1(I,1)*100/CZ(I),C1(I,2)*100
/CZ(I),Z(I)*100/CZ(I),I:NEXT I
1680 RETURN
3000 REM DATA FOR MINIMIX-MW4
3010 REM DATA 24, 22, 2, 1, 1, 1000,1000,1000, .9,.9,.9,  4
3020 DATA 53.095,   95.115,  1404.88,  .261, .214, .160, .148
3030 DATA 53.095,  190.23,   1309.77,  .289, .237, .195, .162
3040 DATA 53.095,  285.34,   1214.66,  .292, .252, .210, .171
3050 DATA 53.095,  380.46,   1119.54,  .305, .265, .211, .179
3060 DATA 53.095,  475.56,   1024.44,  .318, .281, .234, .190
3070 DATA 53.095,  570.69,    923.31,  .329, .295, .259, .201
3080 DATA 53.095,  665.8,     834.20,  .337, .304, .253, .205
3090 DATA 53.095,  760.93,    739.07,  .335, .309, .260, .211
3100 DATA 53.095,  856.03,    643.97,  .350, .321, .269, .220
3110 DATA 53.095,  951.16,    548.84,  .355, .331, .281, .229
3120 DATA 53.095, 1046.30,    453.70,  .358, .335, .282, .230
3130 DATA 53.095, 1141.40,    358.60,  .361, .338, .285, .232
3140 DATA 109.48,   95.115,  1404.88,  .523, .370, .305, .255
3150 DATA 109.48,  190.23,   1309.77,  .557, .441, .354, .284
3160 DATA 109.38,  285.34,   1214.66,  .582, .485, .384, .307
3170 DATA 109.38,  380.46,   1119.54,  .604, .521, .420, .335
3180 DATA 109.38,  475.56,   1024.44,  .620, .547, .440, .346
3190 DATA 105.57,  643,       923.31,  .625, .561, .454, .363
3200 DATA 109.48,  665.8,     834.20,  .667, .593, .484, .384
3210 DATA 109.48,  760.93,    739.07,  .675, .617, .509, .409
3220 DATA 109.48,  856.03,    857.00,  .687, .631, .520, .415
3230 DATA 109.48,  951.16,    548.84,  .695, .641, .525, .424
3240 DATA 109.48, 1046.30,    453.70,  .711, .659, .552, .441
3250 DATA 109.48, 1141.40,    358.60,  .719, .671, .564, .454
3260 REM DATA 1,1,0, 1,0,1, .17, .12, -1.3, .2
3270 REM DATA 5, .25, 5, .25, 5, .25, 0, .1, 0, .1
3280 REM DATA 5, .25, 5, .25, 5, .25, 0, .1, 0, .1
3290 REM DATA 5, .25, 5, .25, 5, .25, 0, .1, 0, .1
3300 REM DATA 5, .25, 5, .25, 5, .25, 0, .1, 0, .1
3310 REM DATA 1, 24, 1
3320 REM DATA 1,1, 1,1, 0,2,20, 3,2,2,4
3330 REM DATA 2,1, 1,1, .9, .01, 1.2, 3,2,2,4
3340 REM DATA 0,1, 1,1, 3,5,2, 3,4,5,  3,4,2, 3,4,5
3350 REM DATA 0,1, 1,2, 3,5,2, 8,9,10, 3,4,2, 8,9,10
3360 REM DATA 0,1, 1,3, 3,5,2, 13,14,15, 3,4,2, 13,14,15
3370 DATA 0,1, 1,4, 3,5,2, 18,19,20, 3,4,2, 18,19,20
3380 DATA 1,4, 4, 1,2,3,4, -1.00, 1, -0.99, 2,1,1,4
3390 DATA 0,1, 1,1, 3,5,2, 3,4,5,  3,4,2, 3,4,5
3400 DATA 0,1, 1,2, 3,5,2, 8,9,10, 3,4,2, 8,9,10
3410 DATA 0,1, 1,3, 3,5,2, 13,14,15, 3,4,2, 13,14,15
3420 DATA 0,1, 1,4, 3,5,2, 18,19,20, 3,4,2, 18,19,20
3430 DATA 0,4, 4, 1,2,3,4, 2,6,6, 1,2
```