

DTIC FILE COPY

ESL-TR-86-66
VOL III



**EVALUATION AND PREDICTION OF
HENRY'S LAW CONSTANTS AND
AQUEOUS SOLUBILITIES FOR
SOLVENTS AND HYDROCARBON FUEL
COMPONENTS
VOL III: EXPERIMENTAL SOLUBILITY
DATA**

G.B. HOWE, M.E. MULLINS, T.N. ROGERS

RESEARCH TRIANGLE INSTITUTE
P.O. BOX 12194
RESEARCH TRIANGLE PARK NC 27709

SEPTEMBER 1987

FINAL REPORT

FEBRUARY 1985 - SEPTEMBER 1986

DTIC
ELECTRONIC
S E D
DEC 12 1988
C D

AD-A202 263

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED



**ENGINEERING & SERVICES LABORATORY
AIR FORCE ENGINEERING & SERVICES CENTER
TYNDALL AIR FORCE BASE, FLORIDA 32403**

NOTICE

PLEASE DO NOT REQUEST COPIES OF THIS REPORT FROM
HQ AFESC/RD (ENGINEERING AND SERVICES LABORATORY).
ADDITIONAL COPIES MAY BE PURCHASED FROM:

NATIONAL TECHNICAL INFORMATION SERVICE
5285 PORT ROYAL ROAD
SPRINGFIELD, VIRGINIA 22161

FEDERAL GOVERNMENT AGENCIES AND THEIR CONTRACTORS
REGISTERED WITH DEFENSE TECHNICAL INFORMATION CENTER
SHOULD DIRECT REQUESTS FOR COPIES OF THIS REPORT TO:

DEFENSE TECHNICAL INFORMATION CENTER
CAMERON STATION
ALEXANDRIA, VIRGINIA 22314

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED		1b. RESTRICTIVE MARKINGS N/A		
2a. SECURITY CLASSIFICATION AUTHORITY N/A		3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release. Distribution Unlimited		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A				
4. PERFORMING ORGANIZATION REPORT NUMBER(S)		5. MONITORING ORGANIZATION REPORT NUMBER(S) ESL-86-66		
6a. NAME OF PERFORMING ORGANIZATION Research Triangle Institute		6b. OFFICE SYMBOL (if applicable)	7a. NAME OF MONITORING ORGANIZATION HQ AFESC/RDVW	
6c. ADDRESS (City, State, and ZIP Code) P.O. Box 12194 Research Triangle Park NC 27709		7b. ADDRESS (City, State, and ZIP Code) Tyndall AFB FL 32403		
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Same as Block 7		8b. OFFICE SYMBOL (if applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F08635-85-C-0054	
8c. ADDRESS (City, State, and ZIP Code)		10. SOURCE OF FUNDING NUMBERS		
		PROGRAM ELEMENT NO. 62601F	PROJECT NO. 1900	TASK NO. 70
11. TITLE (Include Security Classification) Evaluation and Prediction of Henry's Law Constants and Aqueous Solubilities for Solvents and Hydrocarbon Fuel Components				
12. PERSONAL AUTHOR(S) Howe, G.B.; Mullins, M.E.; Rogers, T.N.				
13a. TYPE OF REPORT Final		13b. TIME COVERED FROM Feb 85 to Sep 86	14. DATE OF REPORT (Year, Month, Day) September 1987	15. PAGE COUNT 114
16. SUPPLEMENTARY NOTATION Availability of this report is specified on reverse of front cover.				
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number) Volatile organics, solvents, fuels, Henry's law constants, aqueous solubilities, air-water distribution.	
FIELD	GROUP	SUB-GROUP		
07	01			
07	04			
19. ABSTRACT (Continue on reverse if necessary and identify by block number) Laboratory measurements of Henry's law constants are reported for 51 chemicals spanning a wide range of chemical structures and volatilities. A static headspace method (Equilibrium Partitioning in Closed Systems, referred to as EPICS) was used to measure Henry's Law Constant, with the standard batch air-stripping method used as a check. An average precision of 5 percent was obtained for the EPICS runs, and the Henry's law constants agreed reasonably well (within 10 percent) with the batch air-stripping results and other reported experimental values. Measurements were conducted over a temperature range of 10-30°C, and the data were correlated with a temperature regression equation coupled with a temperature-dependent error term based on 95 percent confidence limits. The aqueous solubilities of the study compounds were also determined via the shake-flask method at temperatures of 10, 20, and 30 degrees Celsius. Finally, the results of this study were incorporated into a thermodynamic correlation (UNIFAC), based on chemical structure, which allows the prediction of Henry's law constants and aqueous solubilities for a wide variety of pure compounds and mixtures (Cont'd)				
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT <input type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION	
22a. NAME OF RESPONSIBLE INDIVIDUAL			22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL

DD Form 1473, JUN 86

Previous editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

ITEM 19. ABSTRACT (Cont'd)

in dilute aqueous solutions. Volume III: Experimental Solubility Data (Volume III of III)

This report is presented in three volumes. Volume I contains the technical discussion and tabulated values of Henry's law constants and aqueous solubilities. Volume II contains the experimental Henry's law data. Volume III contains the experimental solubility data and the Fortran source code for the simplex UNIFAC parameter fitting and the interactive program for calculating Henry's law constants and aqueous solubilities.

EXECUTIVE SUMMARY

The Installation Restoration program (IRP) underway at numerous Air Force bases has identified several sites with contaminated soil and groundwater. This subsurface contamination is the result of fuels, cleaning solvents, and degreasers entering the subsurface environment from accidental spills, leaking storage tanks, and past disposal practices. HQ AFESC/ROVW is conducting research aimed at developing treatment strategies for groundwater cleanup, and studying the fate and transport of contaminants in subsurface systems. Many of the contaminants of concern are volatile by nature, and a knowledge of their air-water distribution and aqueous solubility is needed to assess the compounds treatability and to support the basic laboratory studies.

The objectives of this research were to develop Henry's law constants and aqueous solubilities as a function of temperature, for a variety of organic compounds of Air Force concern (Table 1). Secondary objectives were to determine what effect mixed organics, in an aqueous solution, exhibit on individual Henry's law constants and evaluate various methods used to predict Henry's law constants.

This report documents experimentally determined values of Henry's law constants and aqueous solubility for 51 compounds of Air Force concern. The report is presented in three volumes. Volume I contains the technical discussion and tabulated values of Henry's law constants and aqueous solubilities. Volume II and III contain all the raw data and the fortran source code for an interactive program used to predict the chemical parameters. (R)

Many of the contaminants of concern are volatile by nature, and a knowledge of their air-water distribution is required for the design of treatment processes and for providing insight into their environmental fate and transport. A static headspace method (Equilibrium Partitioning In Closed Systems, referred to as EPICS) was used to measure the Henry's law constants, with the standard batch air stripping method used as a check.

The Henry's constants were determined as a function of temperature from 10 to 30 °C (Table 11) and these values were then used to generate temperature regression equations (Table 8). Generally speaking the EPICS' results from this study agree well with other published results (Table 12). However, for many of the compounds reported here, confirmed values of Henry's constant do not exist in the literature, and if they do, values are rarely reported as a function of temperature with rigorous statistics.

Solubility data for organic compounds in water are important for environmental studies because they provide fundamental information necessary to predict transport in aqueous systems. This data may also be used to predict carbon sorption of contaminants, and the air- or stream-stripping behavior for a given compound. The aqueous solubility of the 51 study

compounds were determined at 10, 20, and 30°C (Table 14). Three different methods were used, but the majority of the data were collected using a shake-flask technique. Although the solubilities were not a strong function of temperature over the range studied (i.e., 10-30 °C), several general trends were noted. First, the solubility of the halogenated hydrocarbons increased with temperature. Second, the solubility of the substituted aromatic hydrocarbons increased with temperature. Finally, maxima and minima are observed for a wide range of compounds without any general trend that can be demonstrated to be statistically significant.

Groundwater contamination is often characterized by the presence of several different contaminants rather than one single compound. For this reason, studies were conducted to determine whether the presence of other compounds would affect the Henry's law constant of a single compound. Deviations from ideal behavior were observed (pg 52), but confirming experiments were not performed. Although the results were not conclusive, the project team believes the observed interactions were real and reproducible.

It would not be feasible to experimentally determine Henry's law constants for all chemical compounds. There will be times when a Henry's law constant is needed but an experimentally determined value is not reported and the situation does not permit a laboratory study to determine the constant. For this reason, a technique to accurately estimate Henry's constant using a minimum of physiochemical properties would be useful. Three different thermodynamic techniques for correlating experimental Henry's law constants were examined (page 61). The techniques were examined to determine their applicability to environmental systems and their predictive capacity for unmeasured multicomponent systems. The UNIFAC method proved to be the most effective way of utilizing the data base developed during this project. A computer algorithm to fit the current data to a new environmental UNIFAC binary interaction data base was developed and a portion of the experimental data collected was incorporated into this new data base. The new data base creates improvement in the predictions generated by UNIFAC in the dilute concentration regime (Figures 13 through 16).

PREFACE

This report was prepared by the Research Triangle Institute, Research Triangle Park NC 22707, under Contract No. F08635-85-C-0054. The AFESC/RDVW Project Officer was Captain Richard A. Ashworth.

The report documents Henry's law constants and aqueous solubilities, as a function of temperature, for 51 compounds of Air Force concern. The study was performed between February 1985 and September 1986.

This report is presented in three volumes. Volume I contains the technical discussion and the tabulated values of Henry's law constants and aqueous solubilities. Volume II contains the experimental Henry's law data. Volume III contains the experimental solubility data and the Fortran source code for the simplex UNIFAC parameter fitting and the interactive program for calculating Henry's law constants and aqueous solubilities.

Mention of trademarks and trade names of material and equipment does not constitute endorsement or recommendation for use by the Air Force, nor can the report be used for advertising the product.

This report has been reviewed by the Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.

Richard A. Ashworth

RICHARD A. ASHWORTH, Capt, USAF, BSC
Project Officer

Lawrence D. Hokanson

LAWRENCE D. HOKANSON, Lt Col, USAF
Director, Engineering and Services
Laboratory

Thomas J. Walker

THOMAS J. WALKER, Lt Col, USAF, BSC
Chief, Environics Division



SEARCHED	<input type="checkbox"/>
SERIALIZED	<input type="checkbox"/>
INDEXED	<input type="checkbox"/>
FILED	<input checked="" type="checkbox"/>
A-1	

TABLE OF CONTENTS

Section	Title	Page
APPENDIX C	Summary of Aqueous Solubility Measurements.....	1
APPENDIX D	Computer Documentation.....	45

APPENDIX C

SUMMARY OF AQUEOUS SOLUBILITY MEASUREMENTS

This is a self-contained document with its own internal style,
which varies from our format.

SOLUBILITY MEASUREMENTS

The shake-flask measurements to determine solubility required several calibration points to insure their statistical validity. In this study, at least five independently prepared solutions of unknown concentration were calibrated against the response of the measurement instrument. (In many, if not most, previous studies serial dilutions of one sample were used instead). In the case of aromatic compounds this instrument was a ultra violet (U.V.) spectrometer. The calibration samples were prepared with distilled water in gas tight amber bottles containing no headspace. Since water was employed for U.V. tests, only concentration below the saturation level could be prepared. In some cases, these calibration samples might be too far below the measured U.V. response and additional samples closer to the true solubility limit would be prepared.

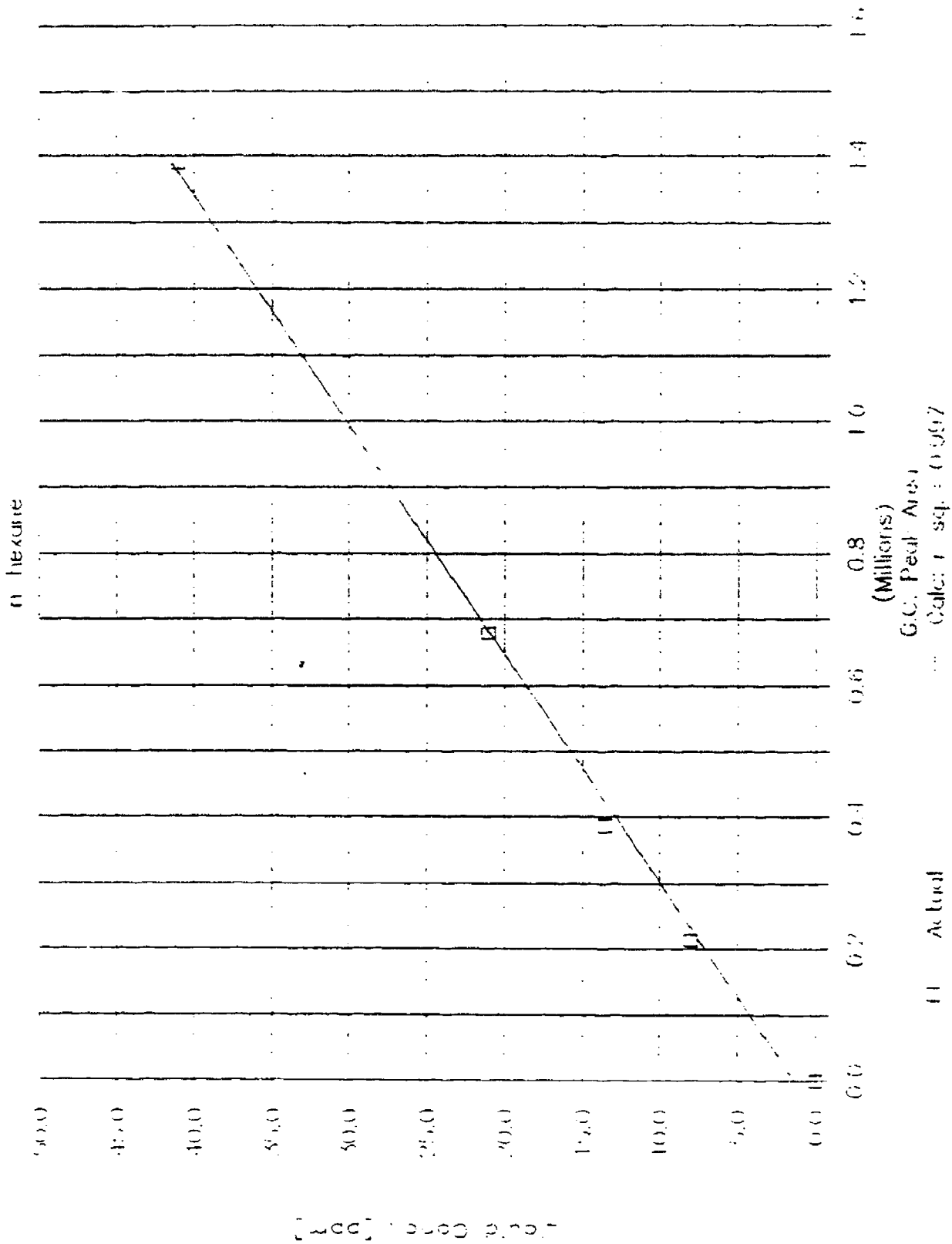
The data was logged onto a Lotus^(R) spreadsheet and a linear regression of the calibration concentrations versus detector response determined. The detector response for the saturated solution at all three temperatures could then be entered, and the corresponding solubility limits determined from extrapolation of the linear regression.

In the case of the gas chromatograph samples, a similar scheme was employed using integrator peak area as the detector response instead of U.V. absorbance. In this case, however, calibration samples were prepared in methanol, so calibration concentrations above the aqueous solubility limit might be prepared.

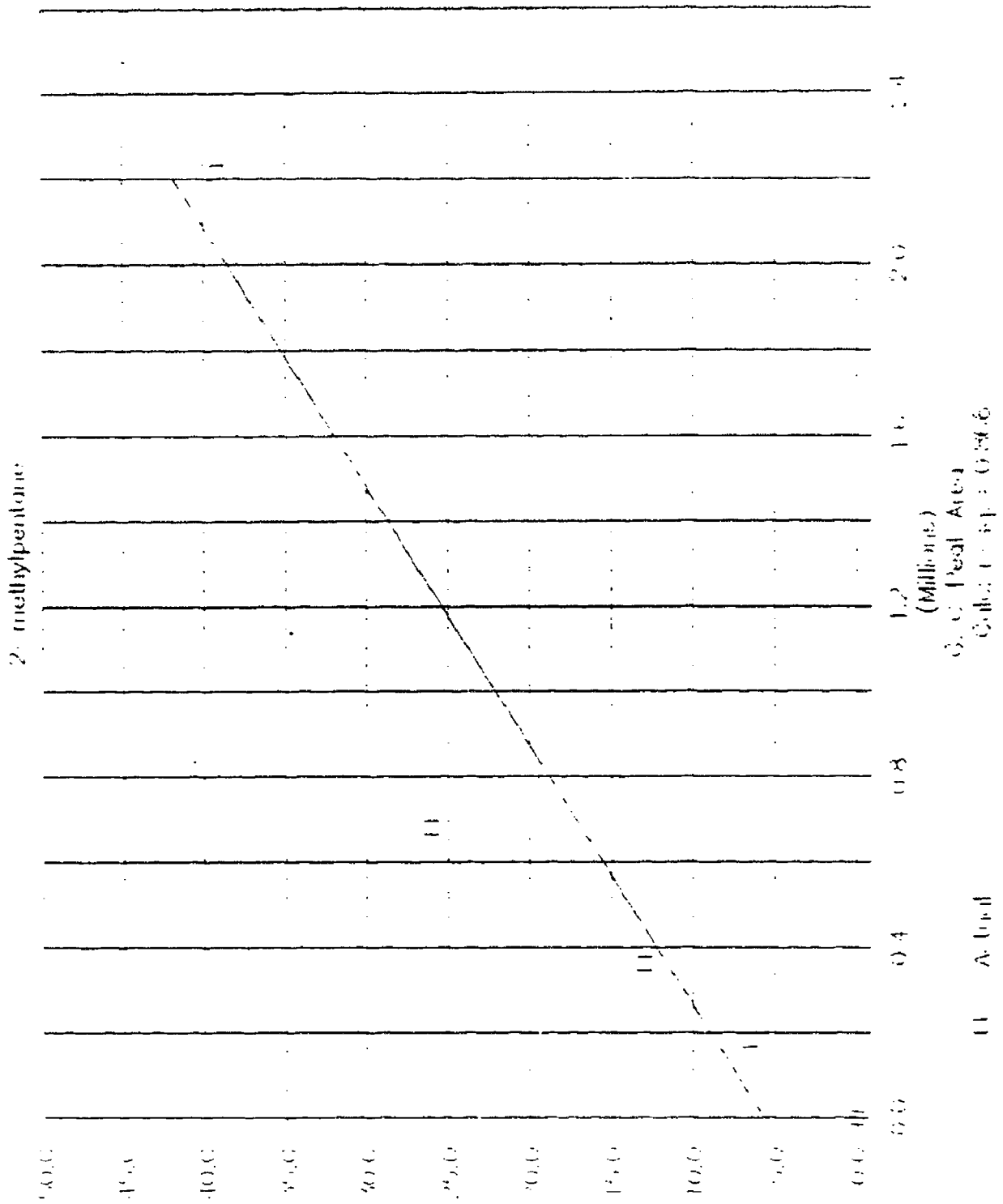
Based upon the correlation coefficient for the linear regression fit on the calibration samples (which is shown in the following plots), 95 percent confidence bands can be placed on the saturated simple concentrations. These are also shown on the following plots. Notice, of course, that the better the correlation coefficient the narrower the bands. Unfortunately, in most cases (even with correlation coefficients in excess of 0.95), little statistical confidence may be placed in the temperature dependency of the solubilities measures. As a consequence, the data has not been tabulated explicitly as a function of temperature dependence.

The nephelometry results are not listed in the same fashion as the other results due to differences in the preparation procedure. However, a discussion of the statistics of nephelometry is contained in Section IV of the main text.

AQUEOUS SOLUBILITY CALIBRATION

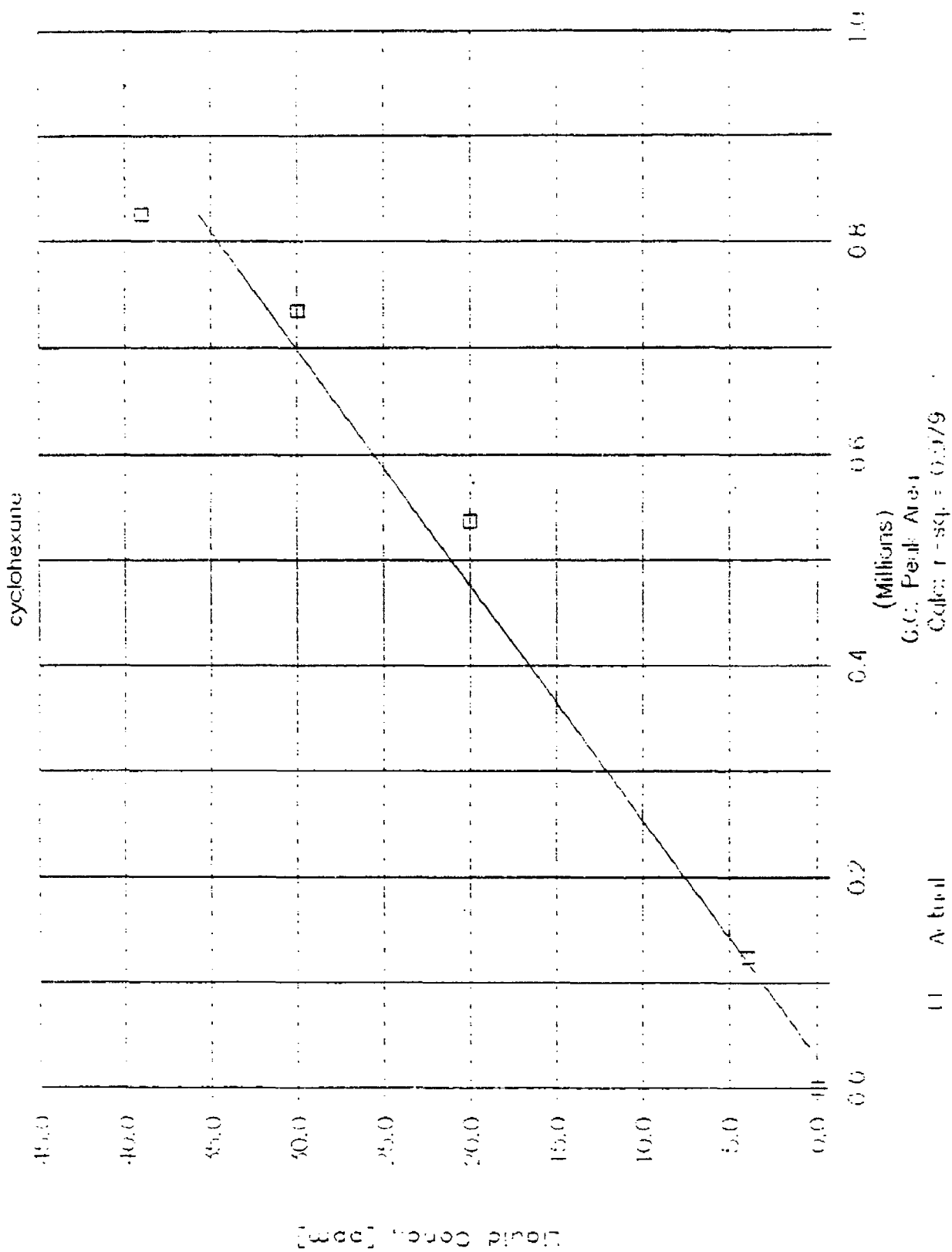


AQUICUP, SOLUBILITY CALIBRATION

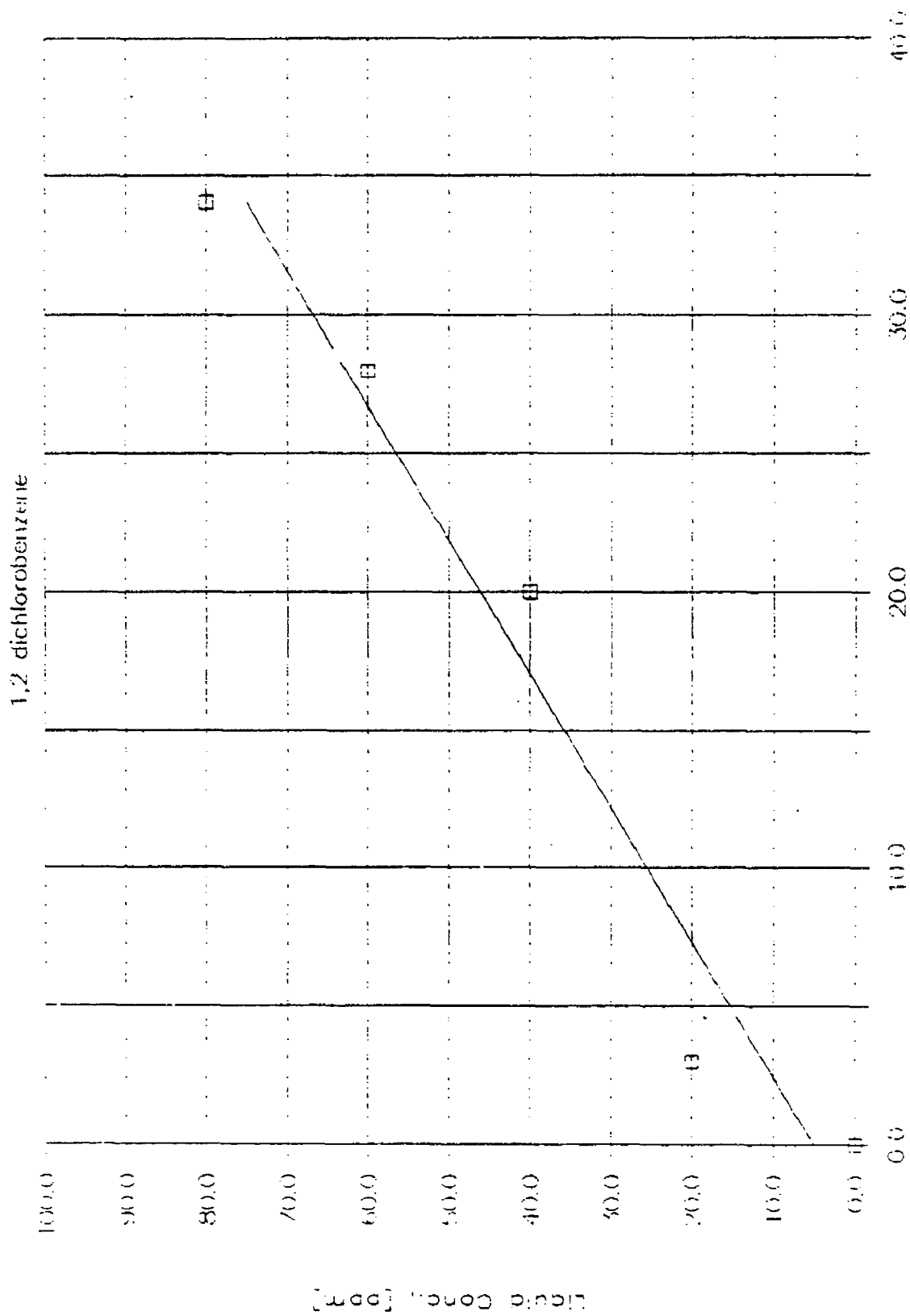


1000 1000 1000 1000

AQUEOUS SOLUBILITY CALIBRATION



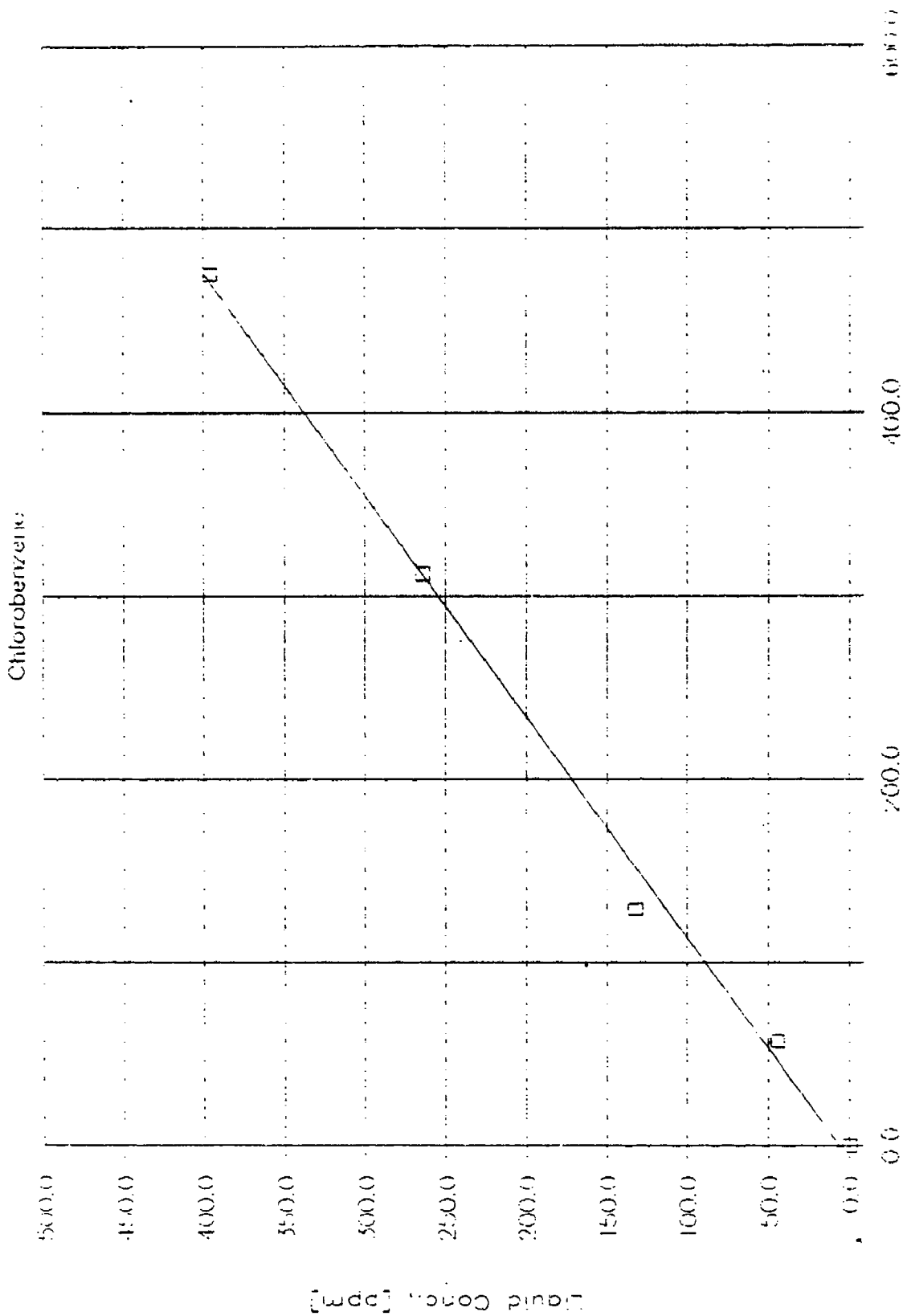
AQUEOUS SOLUBILITY CALIBRATION



U.V. Detector Scale
Calc: $r = 0.997$

□ Actual

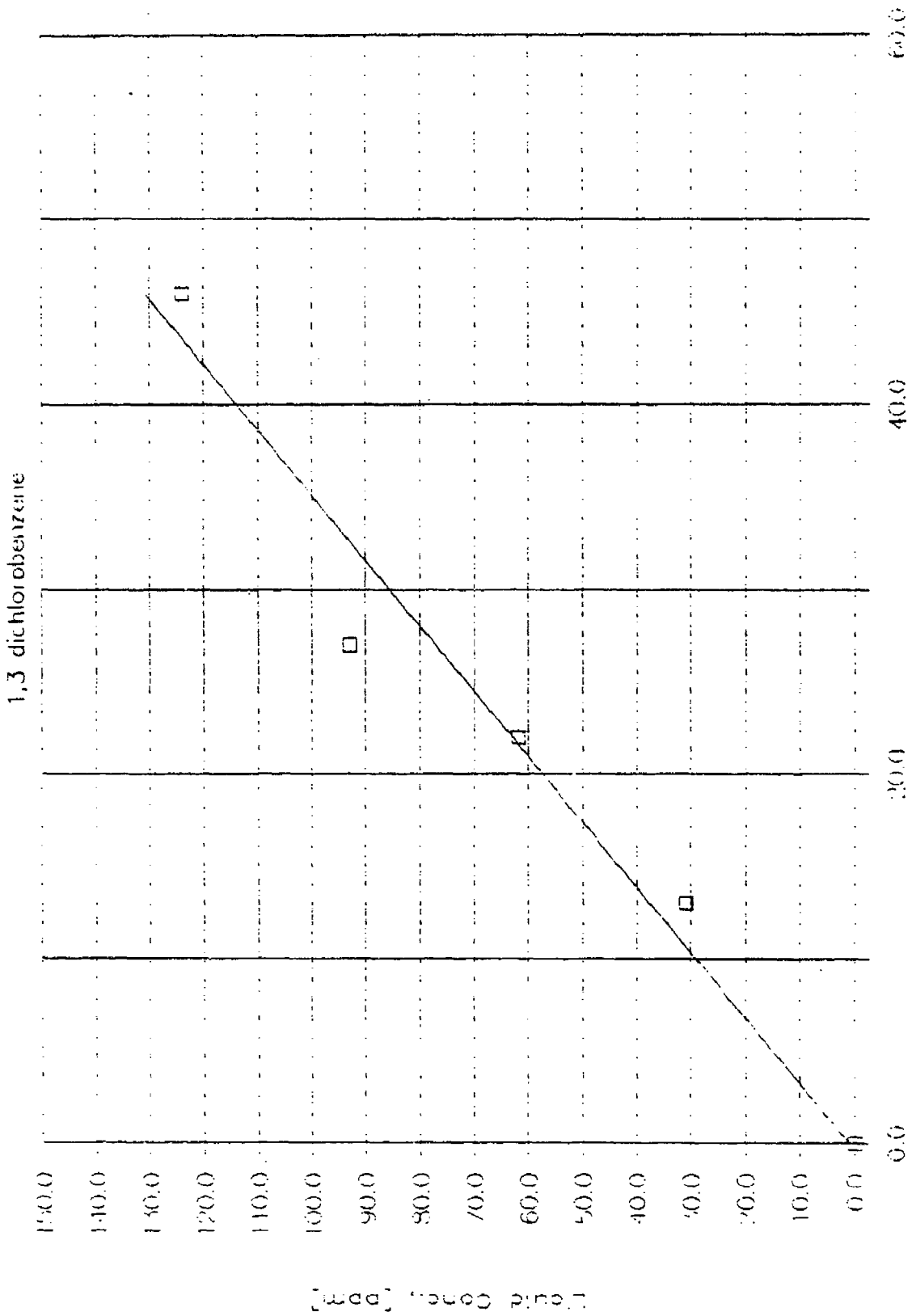
AQUEOUS SOLUBILITY CALIBRATION



U.V. Detector Scale
Calc: $r = \text{sq} = 0.995$

□ Actual

AQUEOUS SOLUBILITY CALIBRATION

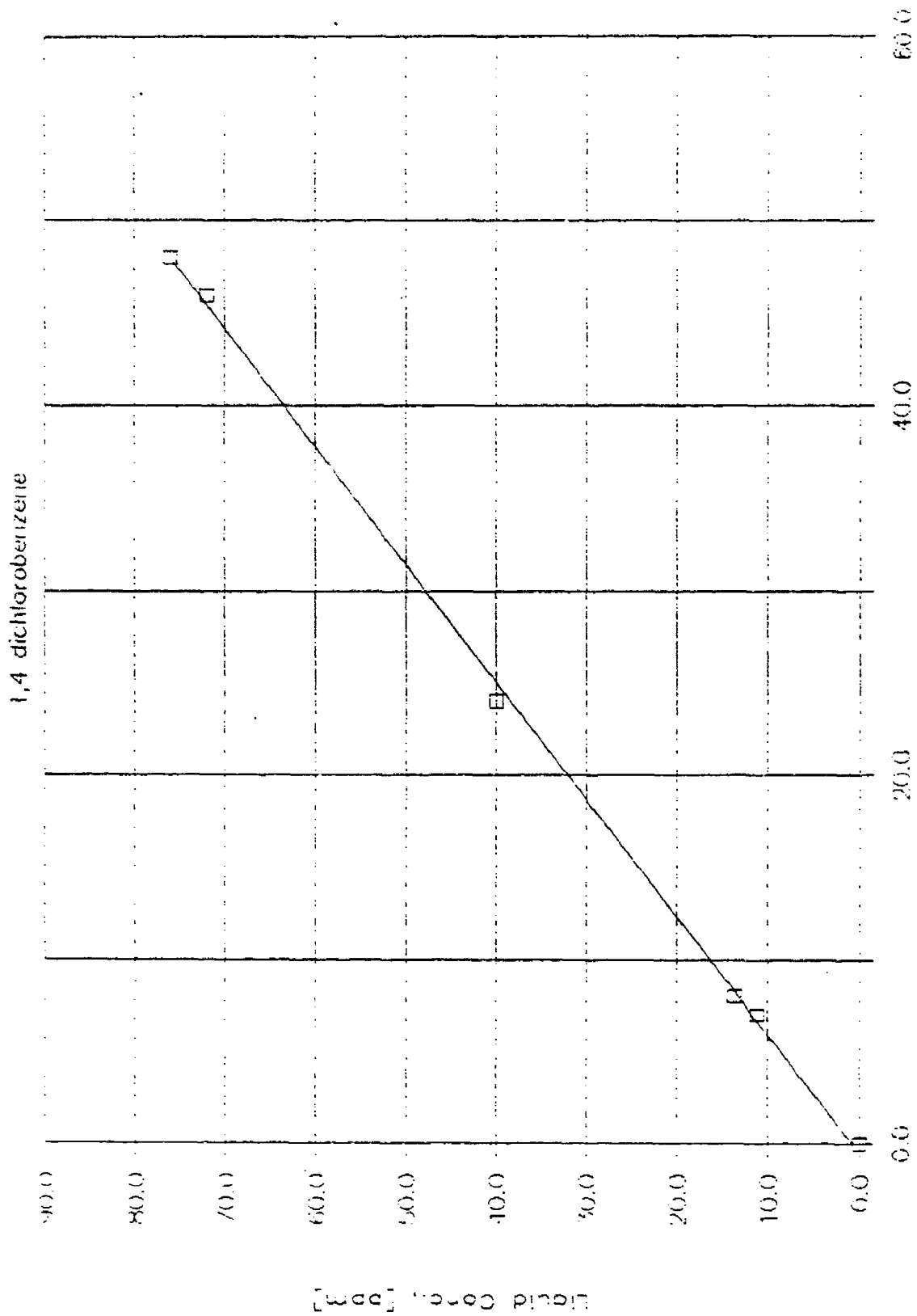


U.V. Detector Scale
Calculated by Eq. 3-0.164

() Actual

U.V. Detector Scale [ppm]

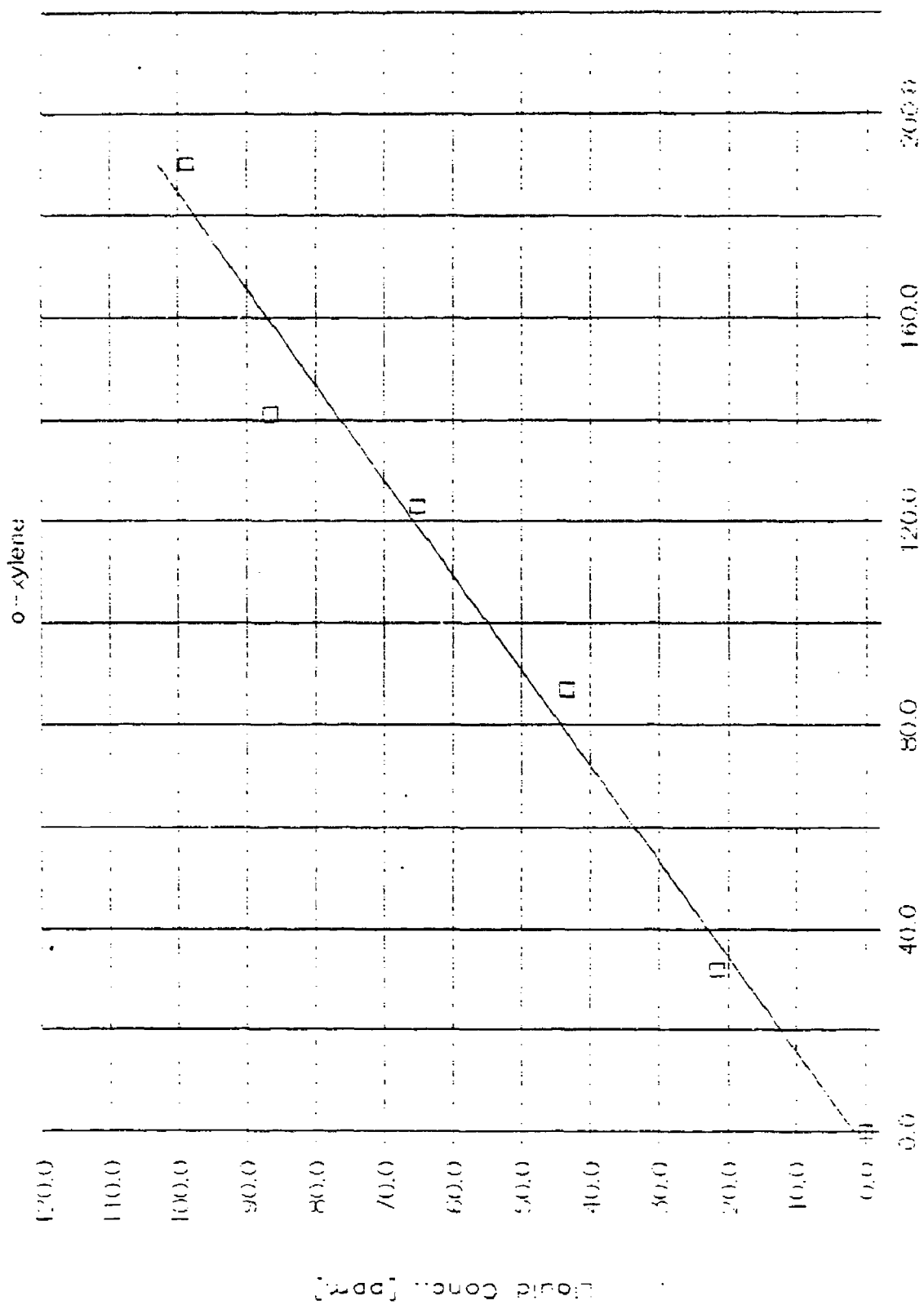
AQUEOUS SOLUBILITY CALIBRATION



U.V. Detector Scale
Calculated: 0.999

Actual

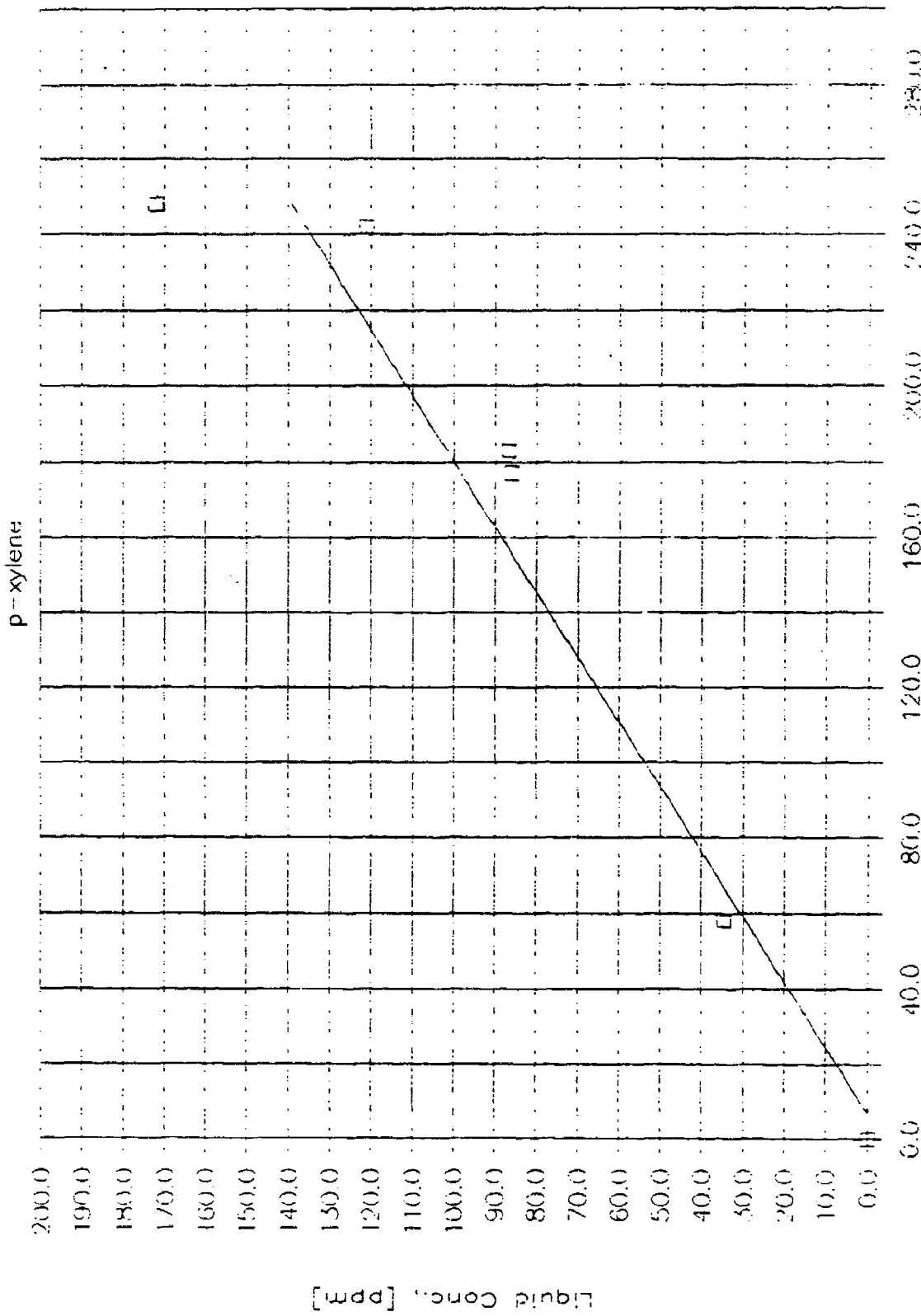
AQUEOUS SOLUBILITY CALIBRATION



U.V. Detector Scale
Cal: 1-19-79

Actual

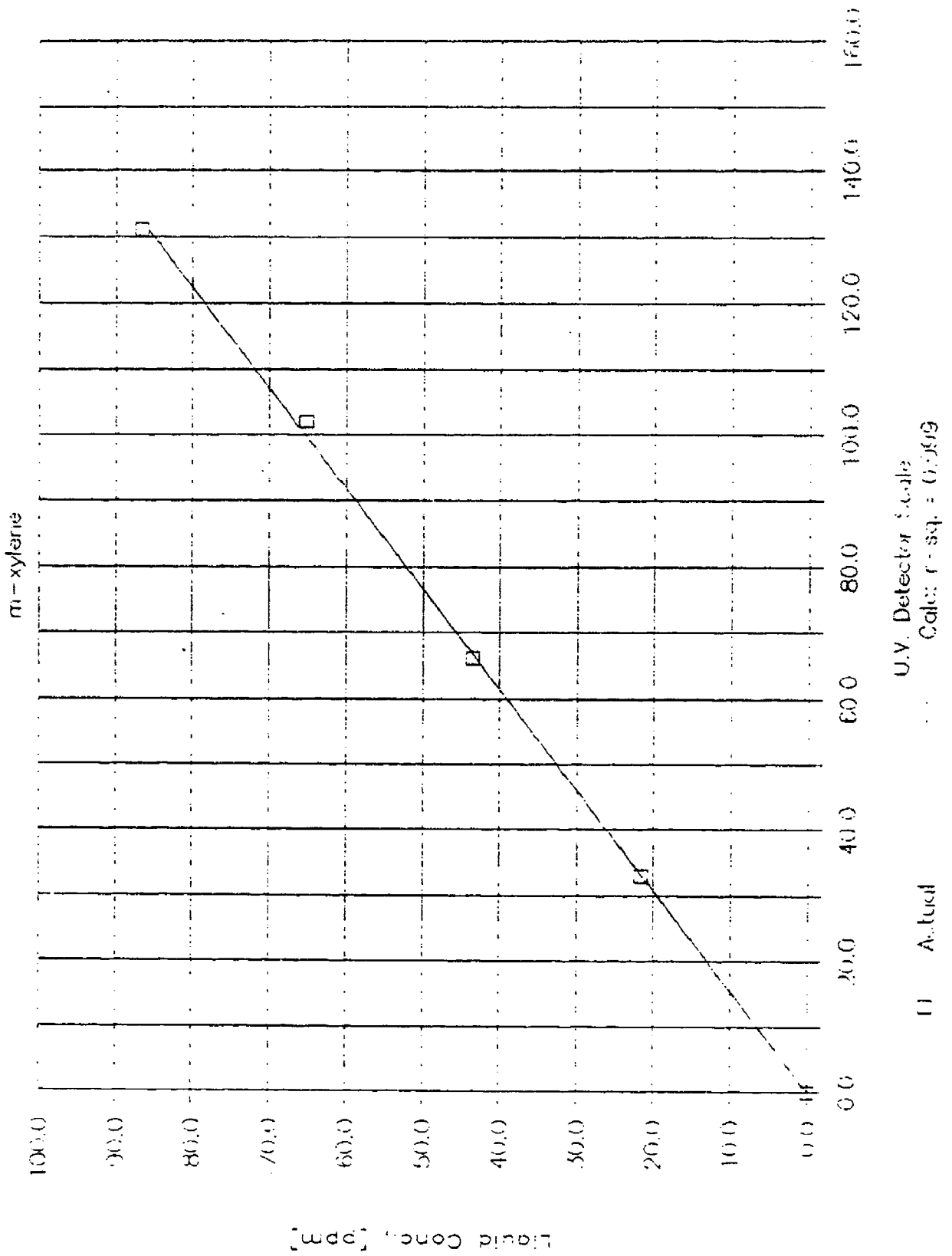
AQUEOUS SOLUBILITY CALIBRATION



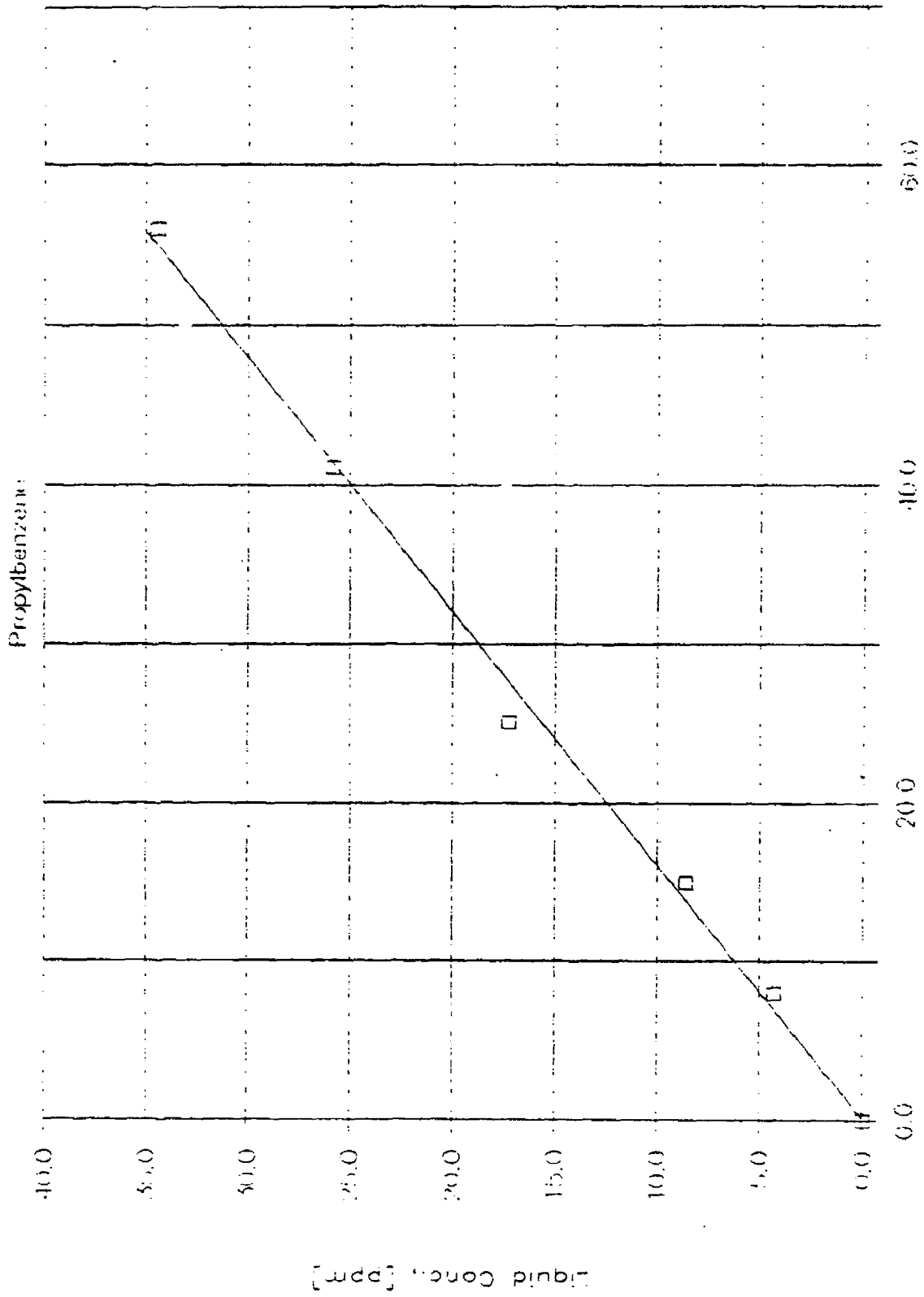
U.V. Detector Scale
Calc: $r = \text{sq} = 0.998$

|| Actual

AQUEOUS SOLUBILITY CALIBRATION



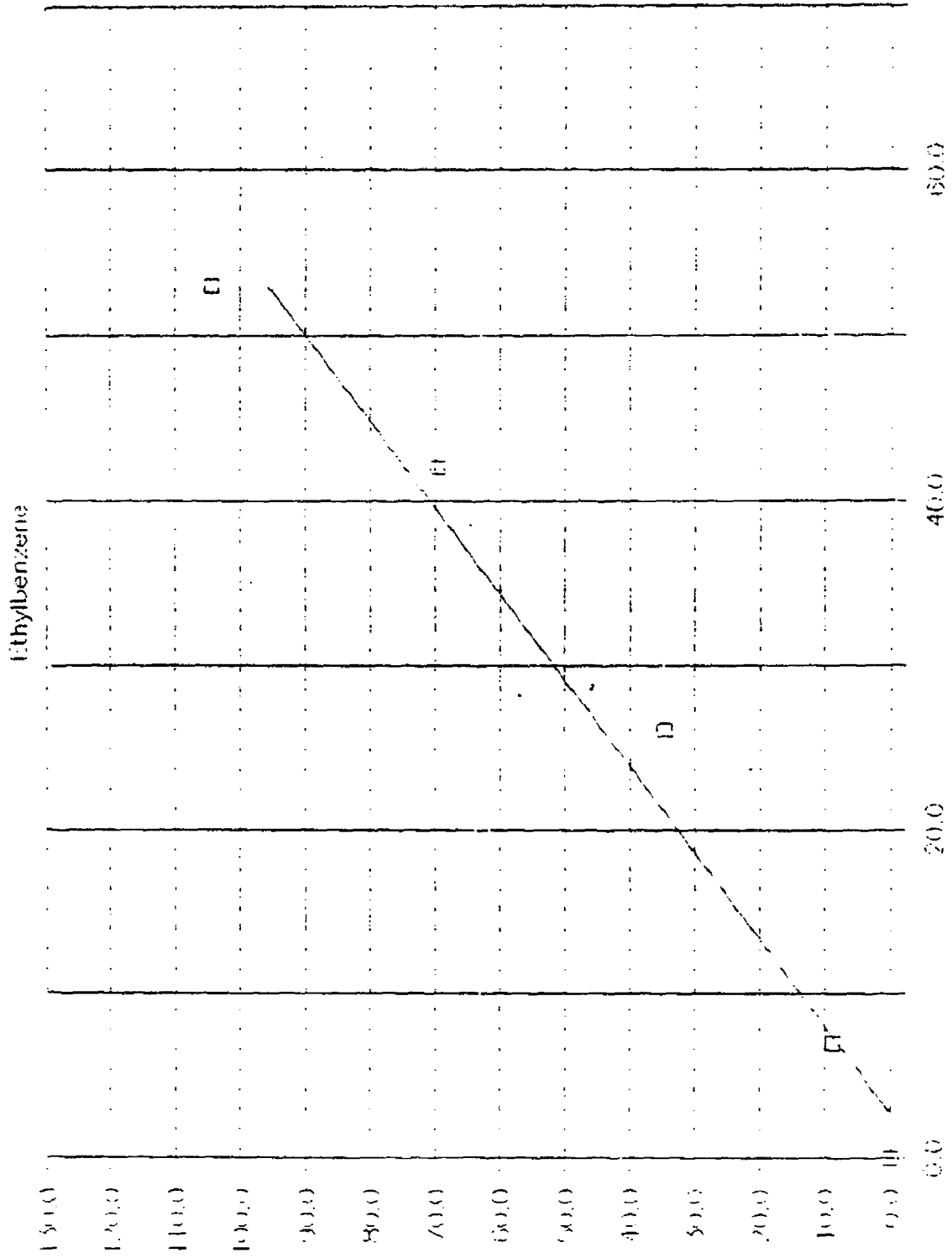
AQUEOUS SOLUBILITY CALIBRATION



UV Detector Scale
Cal: 1-3-86 0.006

□ Actual

AQUEOUS SOLUBILITY CALIBRATION

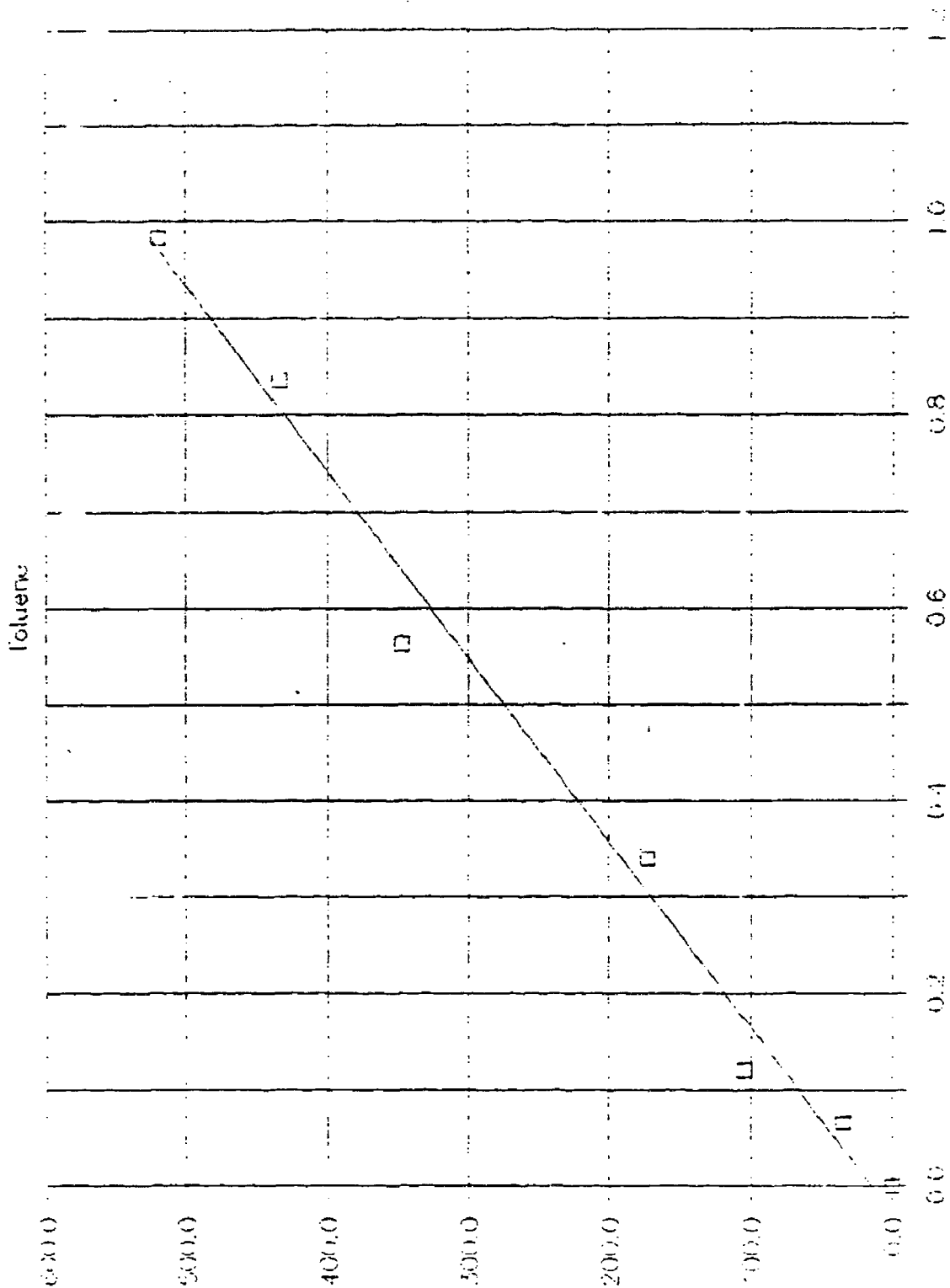


U.V. Detector Scale
Calibration Curve (D)

D = Actual

[Scale Correction]

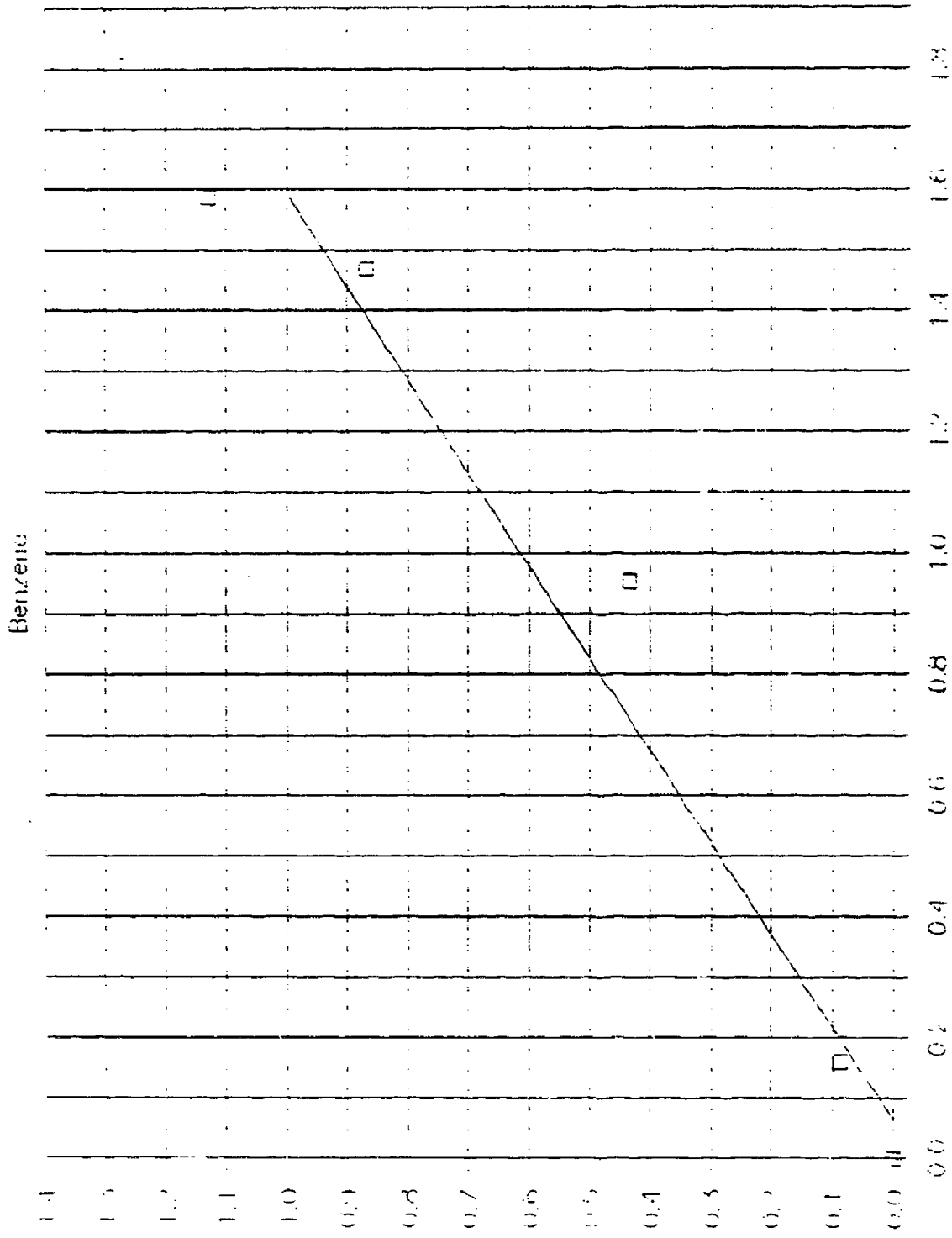
AQUEOUS SOLUBILITY CALIBRATION



U.V. Detector Model
 Calculator Model 10087

U - Actual

AQUEOUS SOLUBILITY CALIBRATION

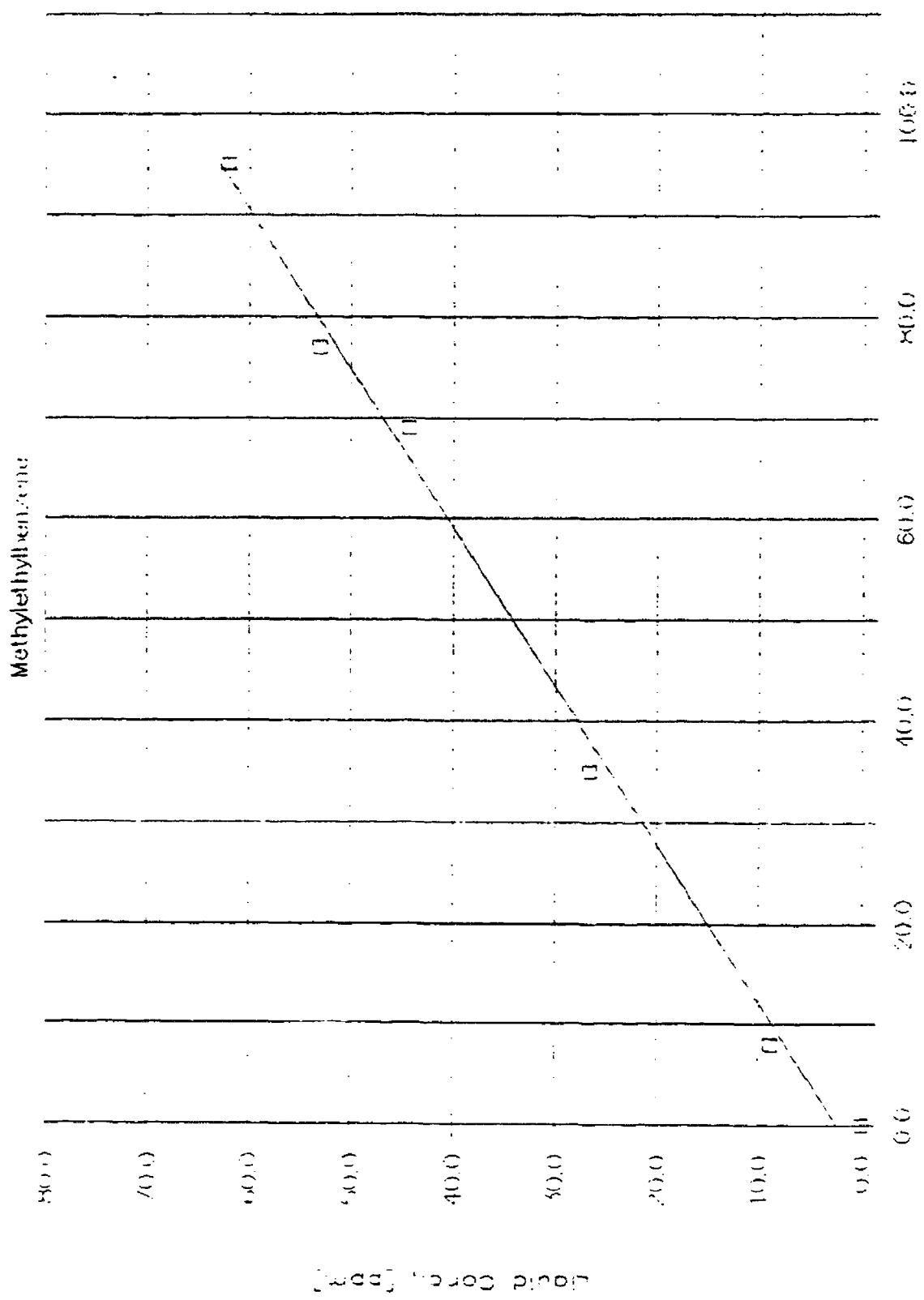


UV Detector Model
 Code: 1-51-0003

Actual

(50,000,000)
 100,000,000
 1,000,000,000

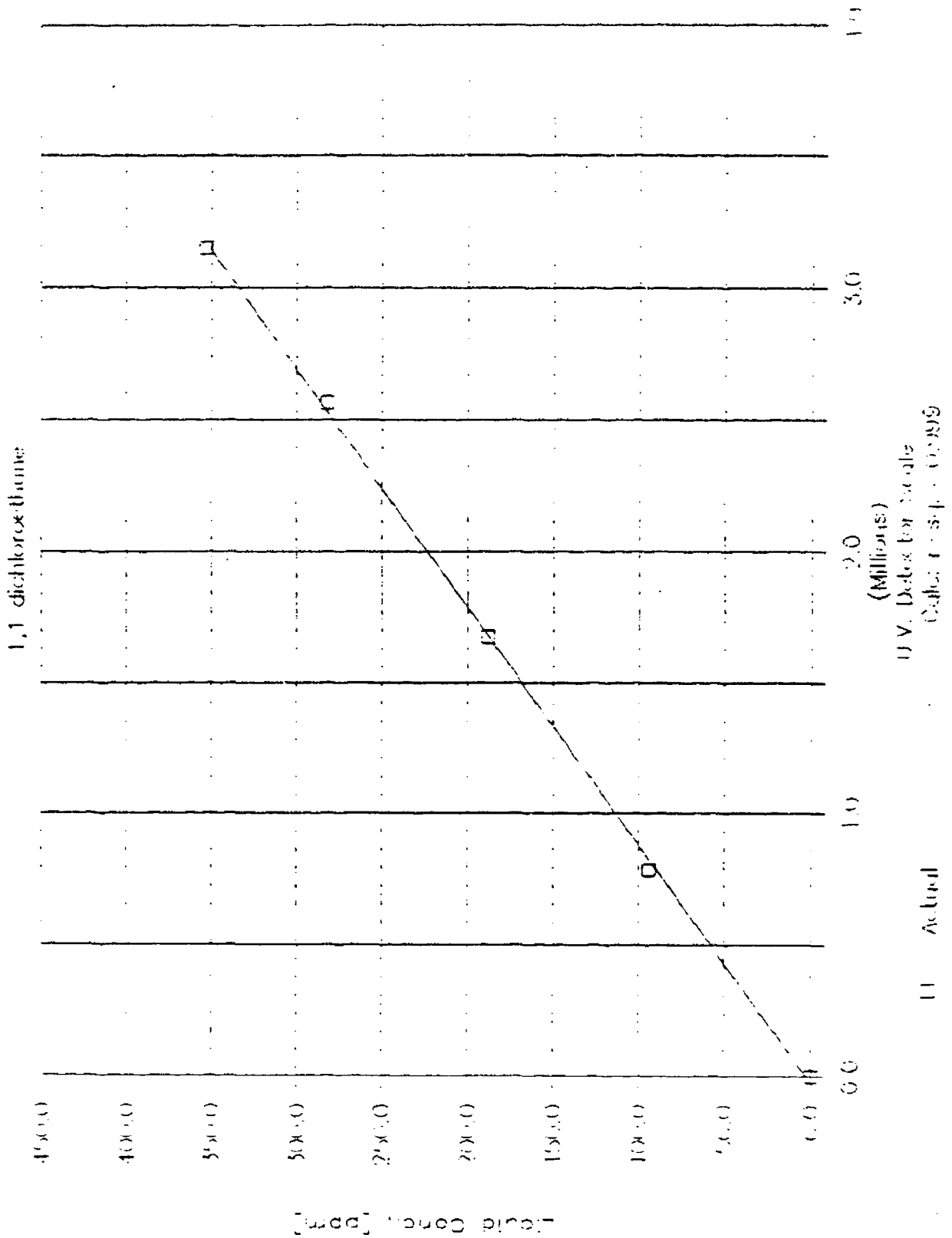
AQUEOUS SOLUBILITY CALIBRATION



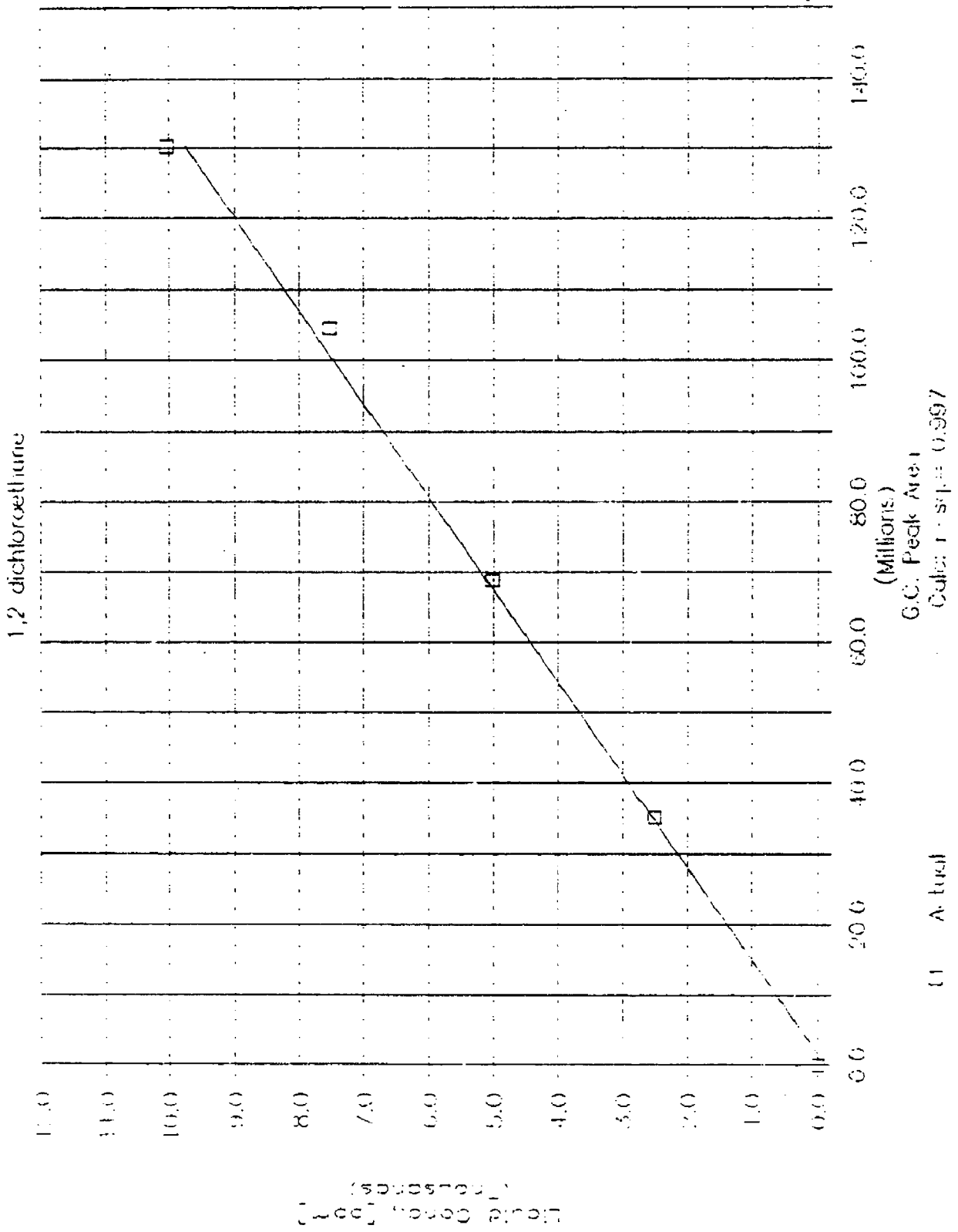
UV Detector Model
 Calibration: 0-100

□ Actual

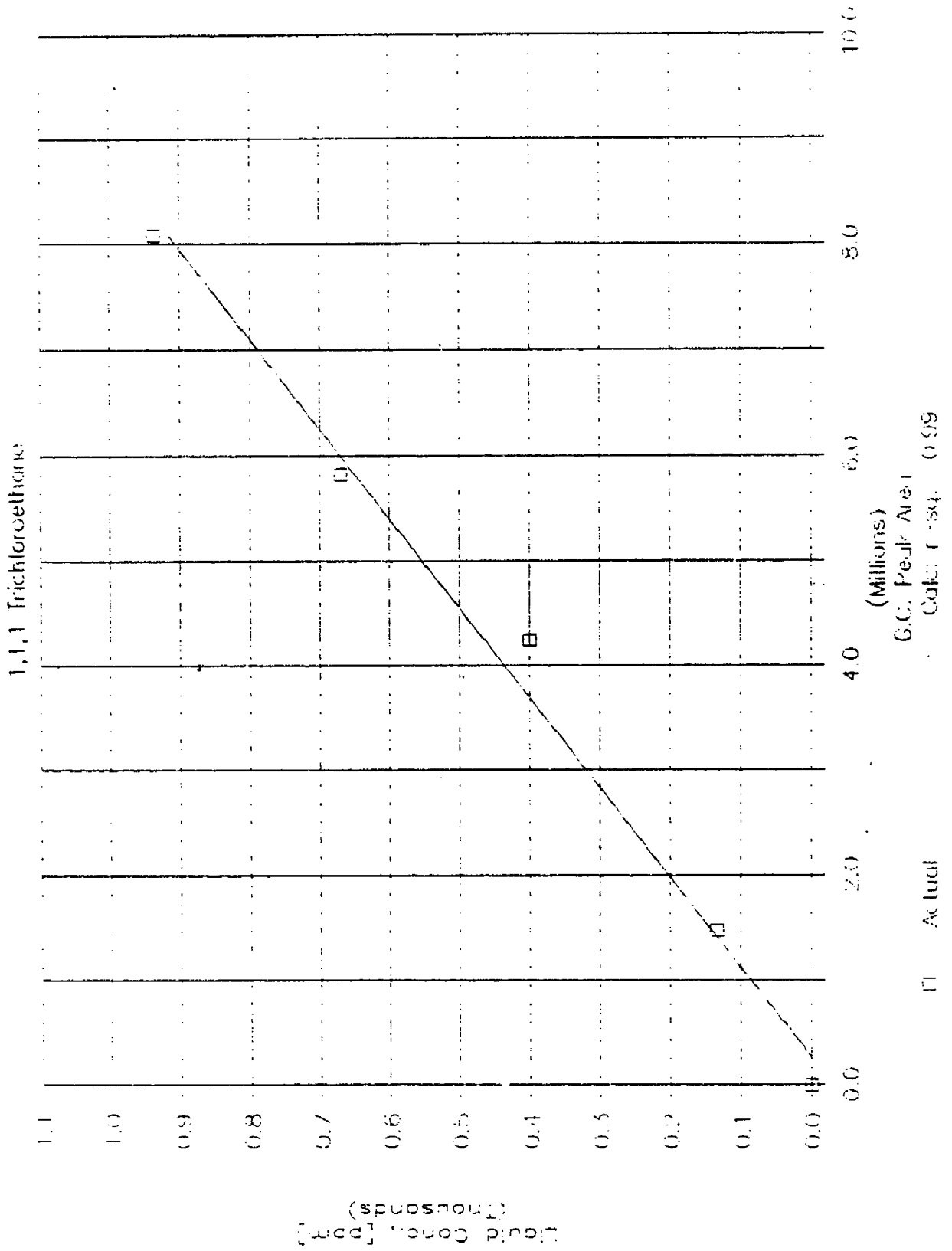
AQUEOUS SOLUBILITY CALIBRATION



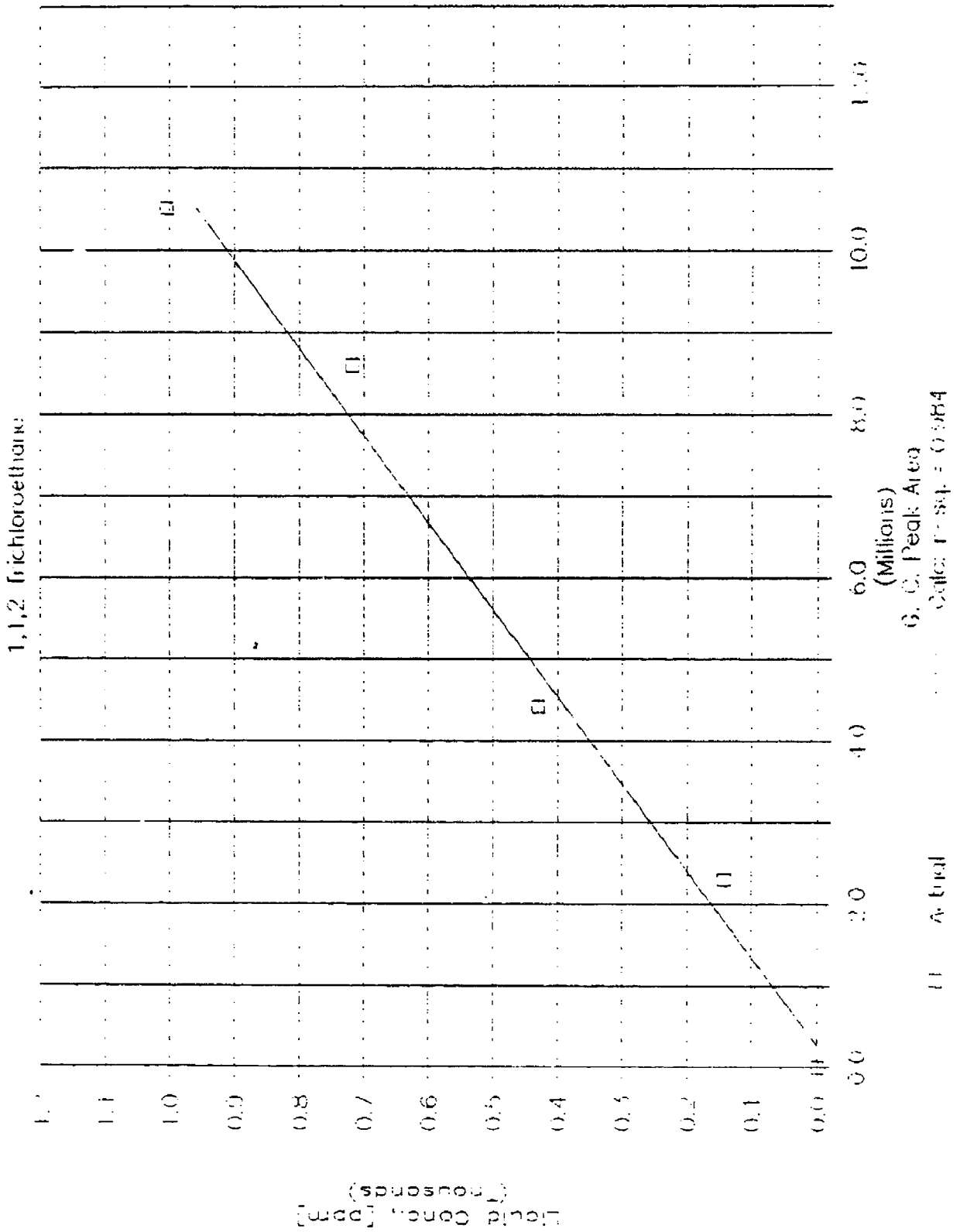
AQUEOUS SOLUBILITY CALIBRATION



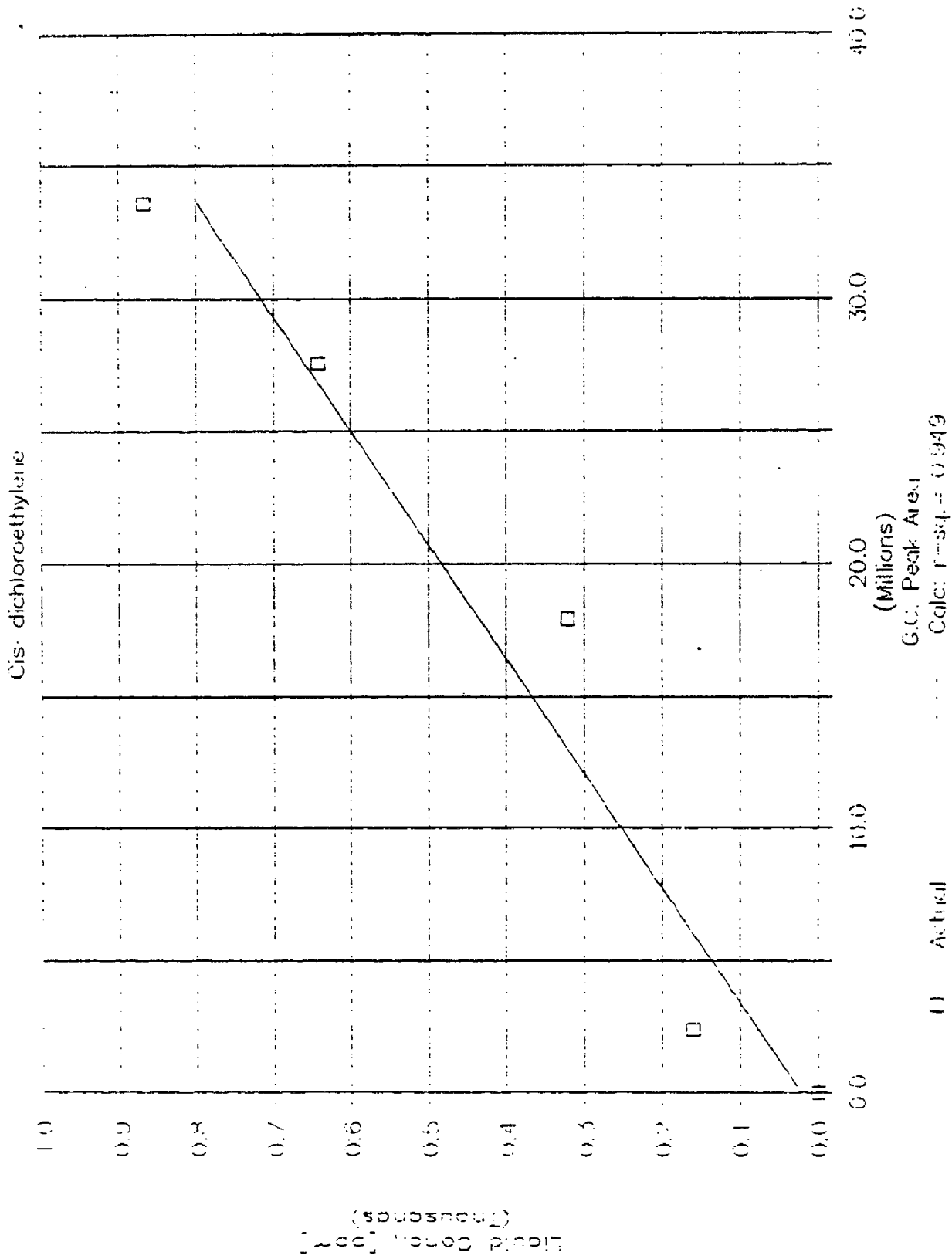
AQUEOUS SOLUBILITY CALIBRATION



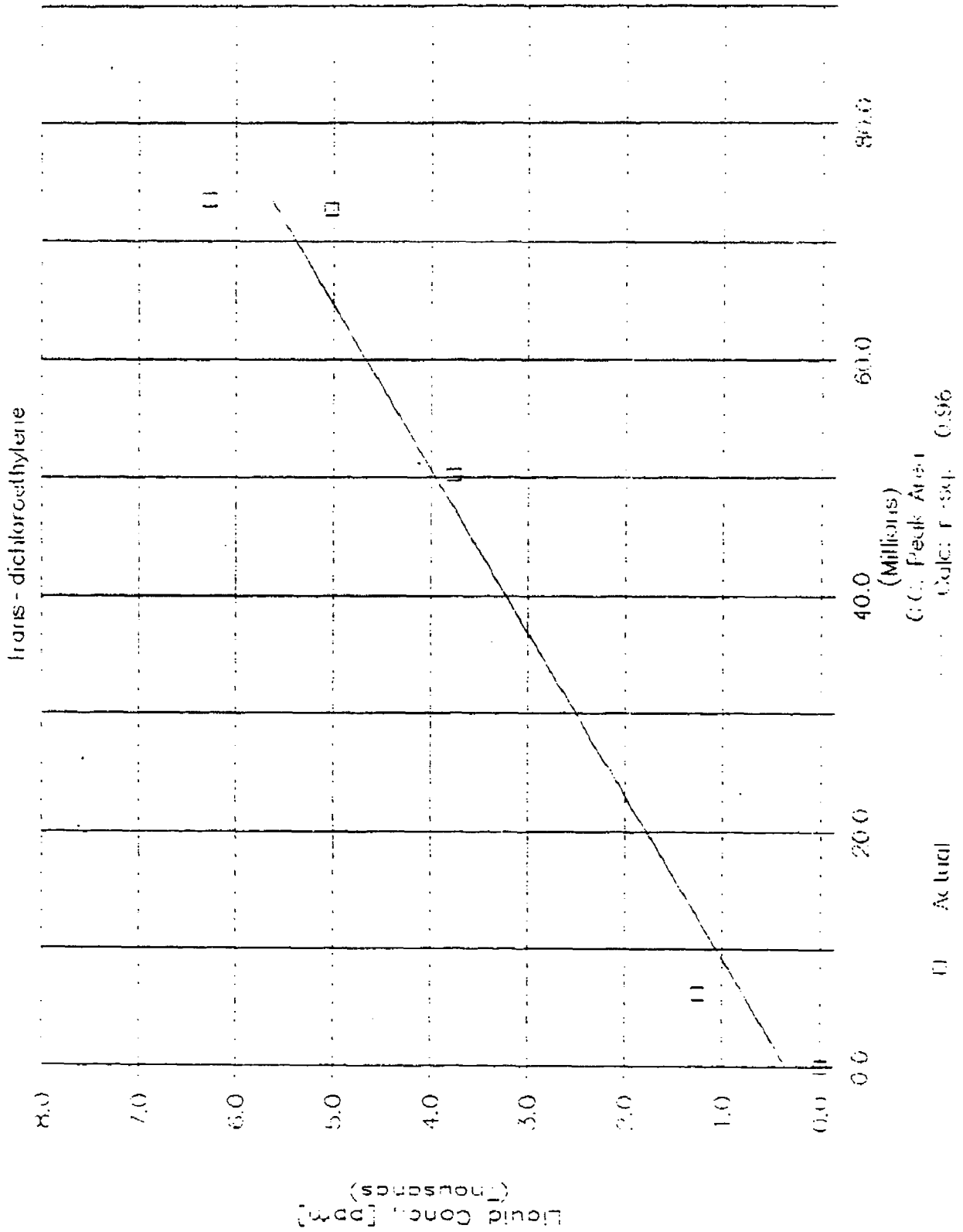
AQUEOUS SOLUBILITY CALIBRATION



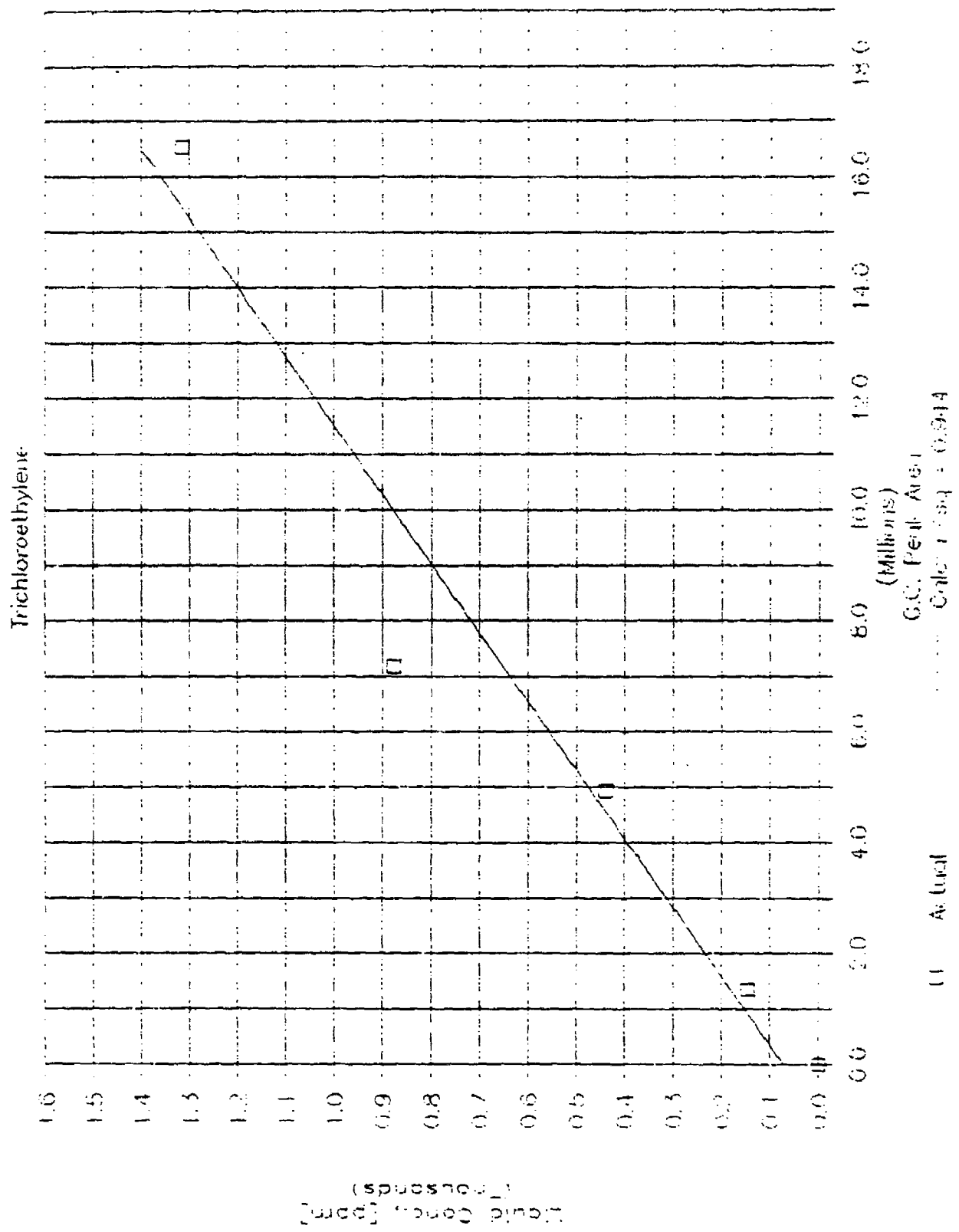
AQUEOUS SOLUBILITY CALIBRATION



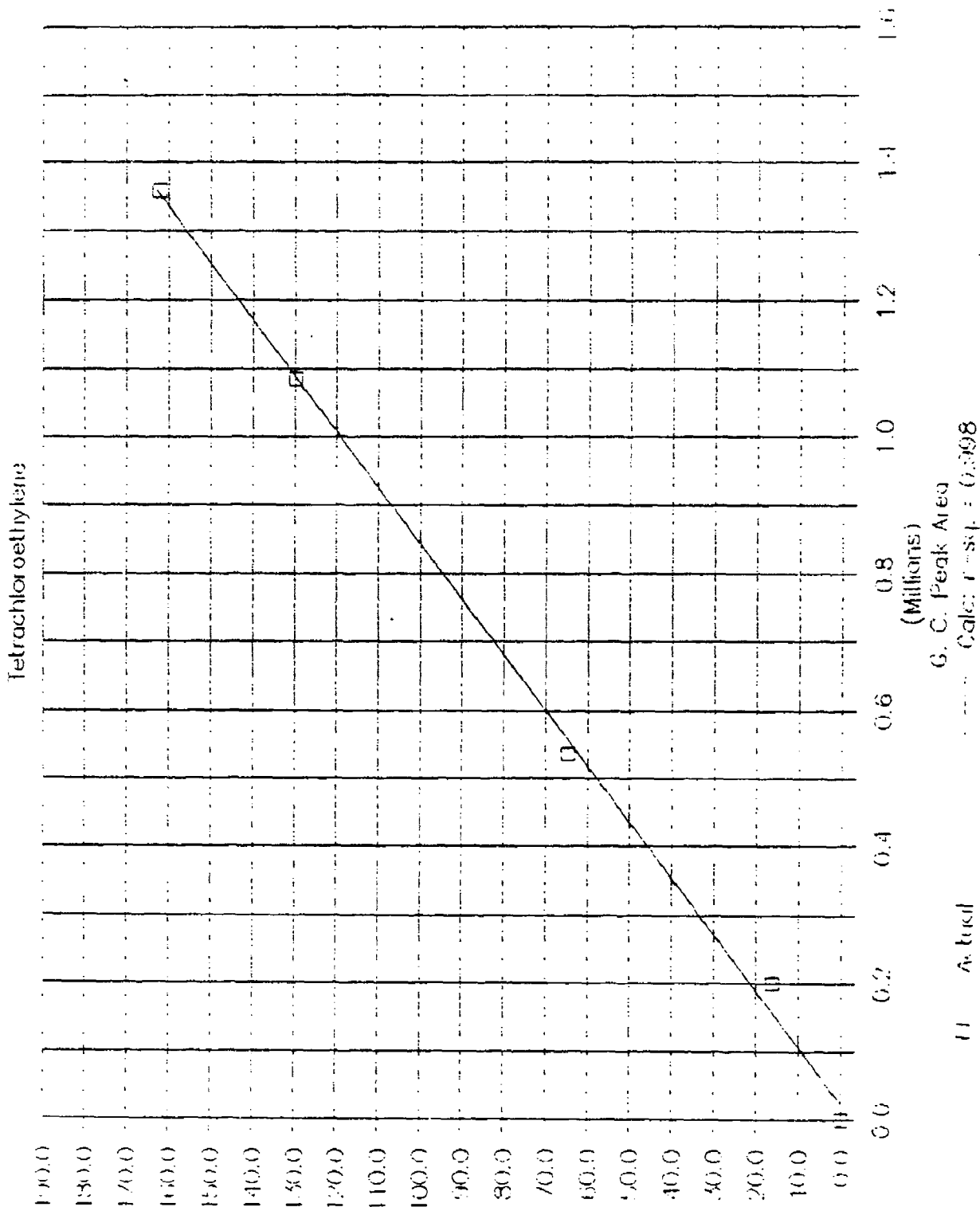
AQUEOUS SOLUBILITY CALIBRATION



AQUEOUS SOLUBILITY CALIBRATION

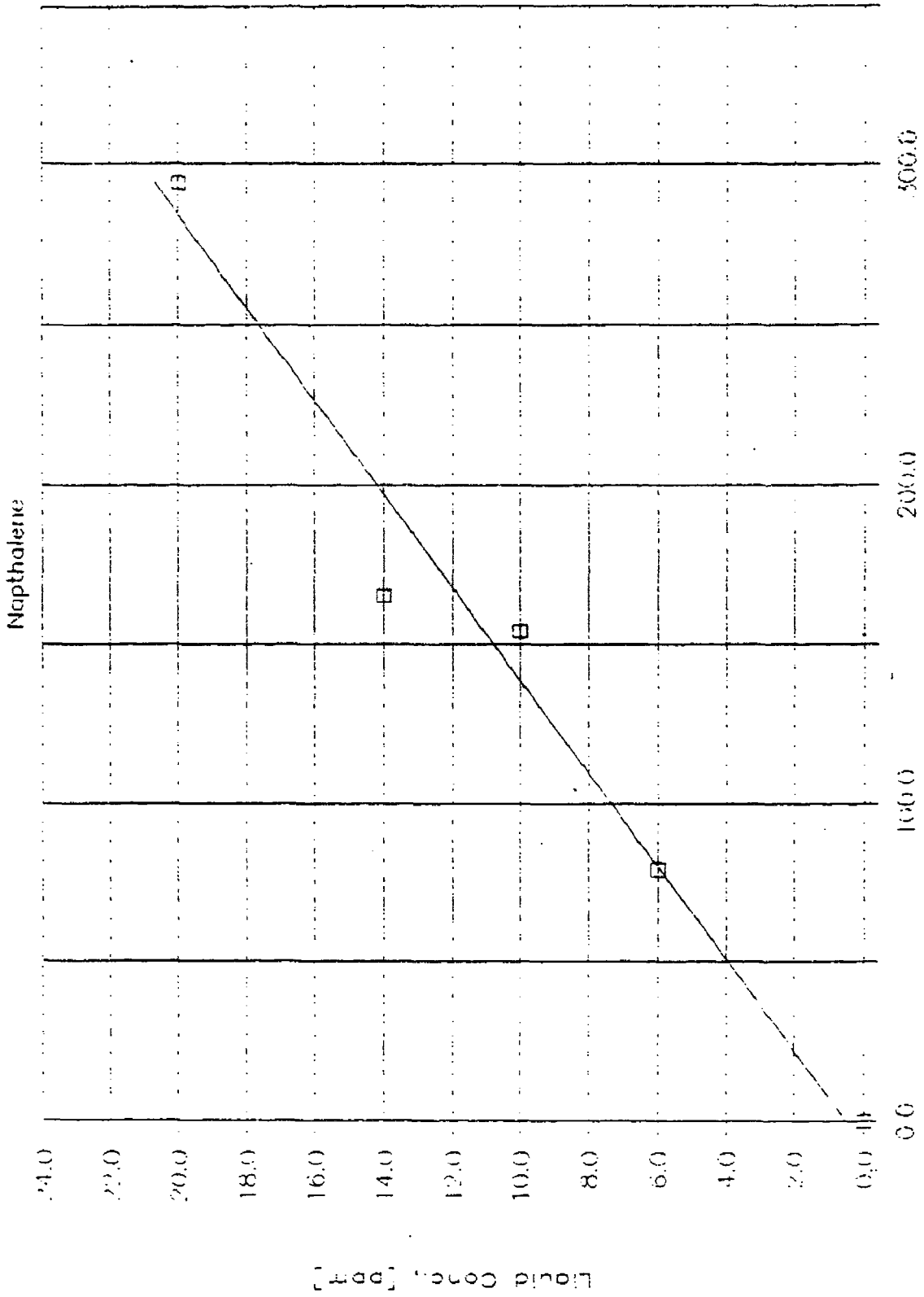


AQUEOUS SOLUBILITY CALIBRATION



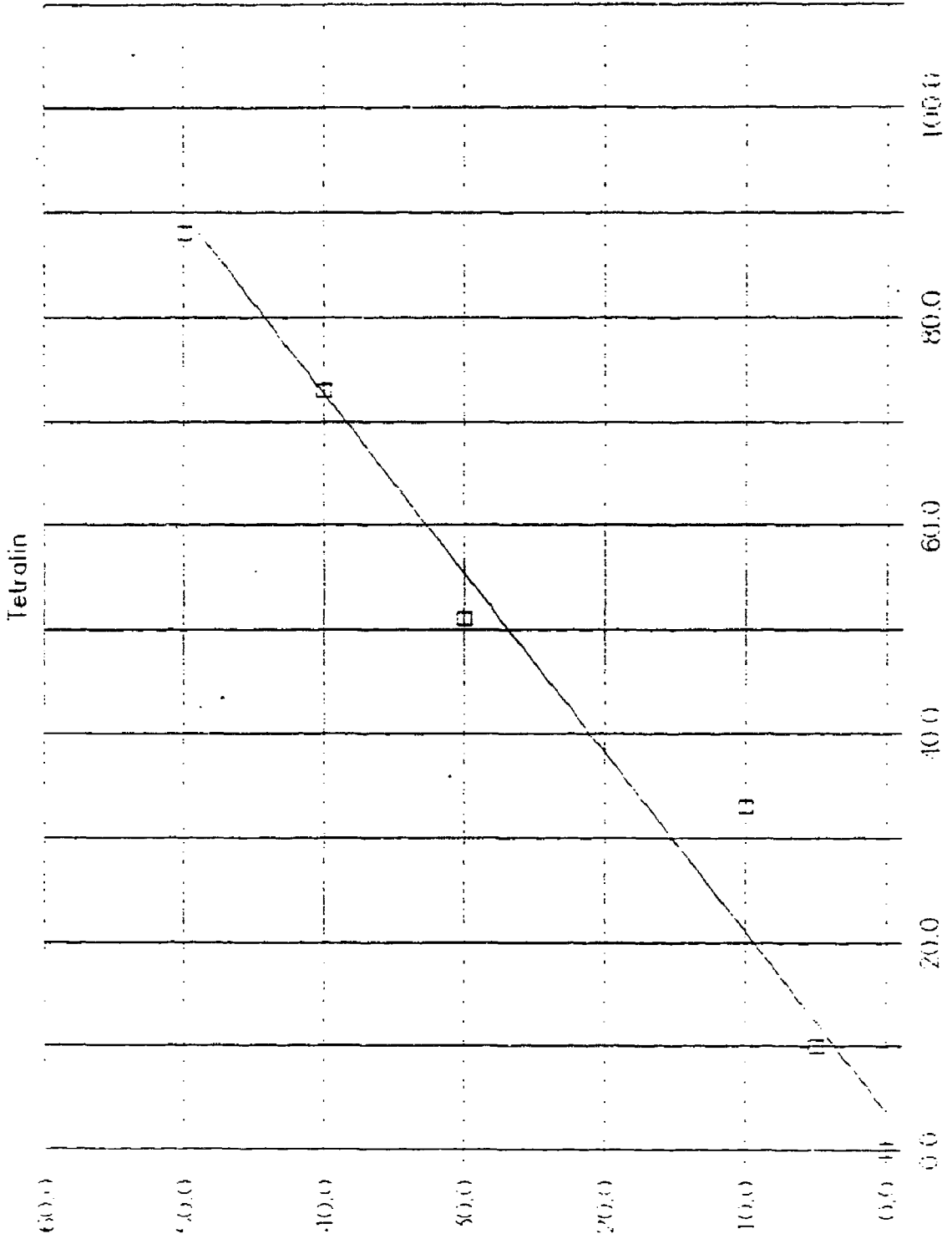
[200] 10 100 200 300

AQUEOUS SOLUBILITY CALIBRATION



U.V. Detector Scale
Calc: $r^2 = 0.972$

• • • AQUEOUS SOLUBILITY CALIBRATION

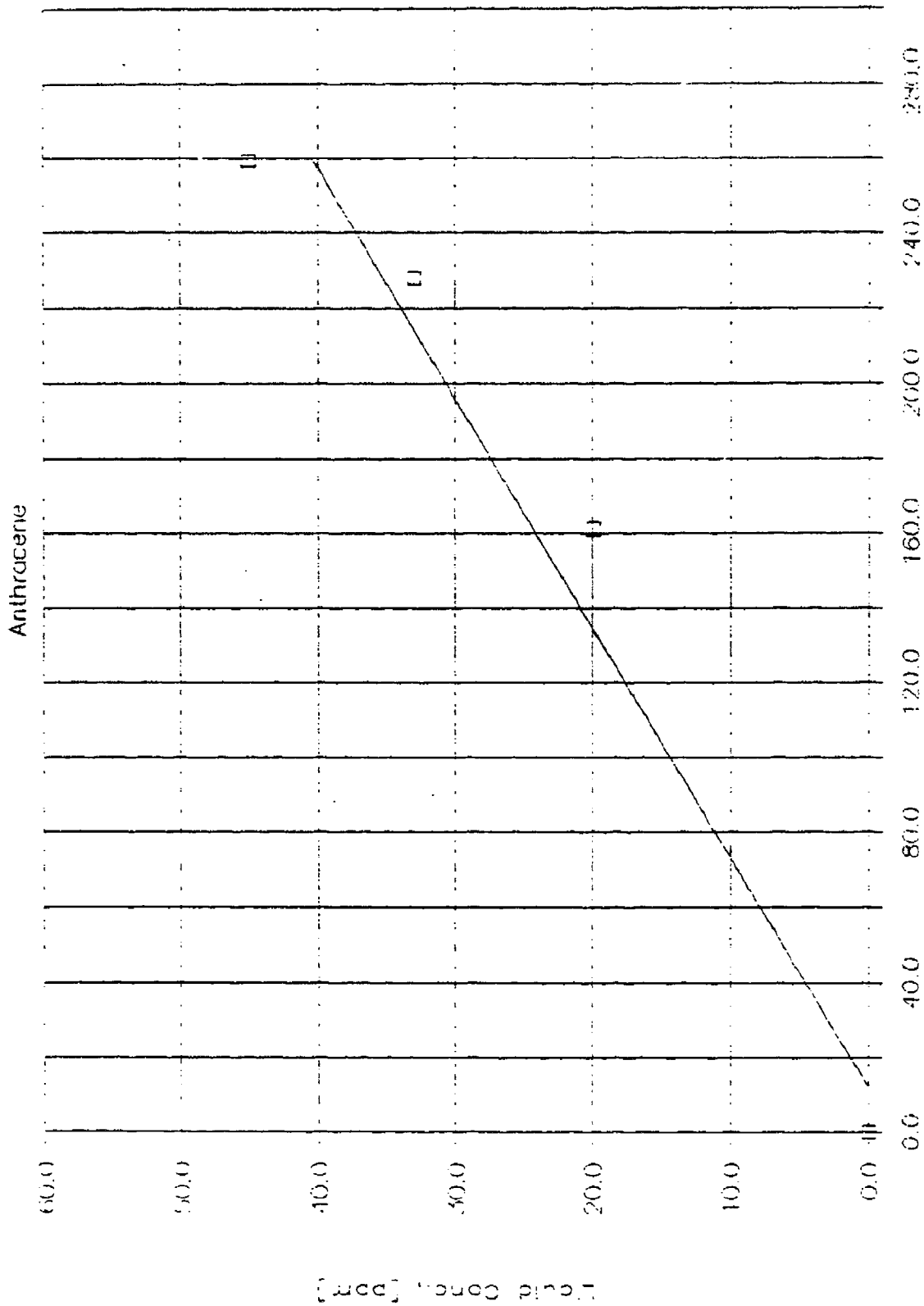


U.V. Detector Scale
Calc: r-sq = 0.97

U Actual

Liquid Conc., [ppm]

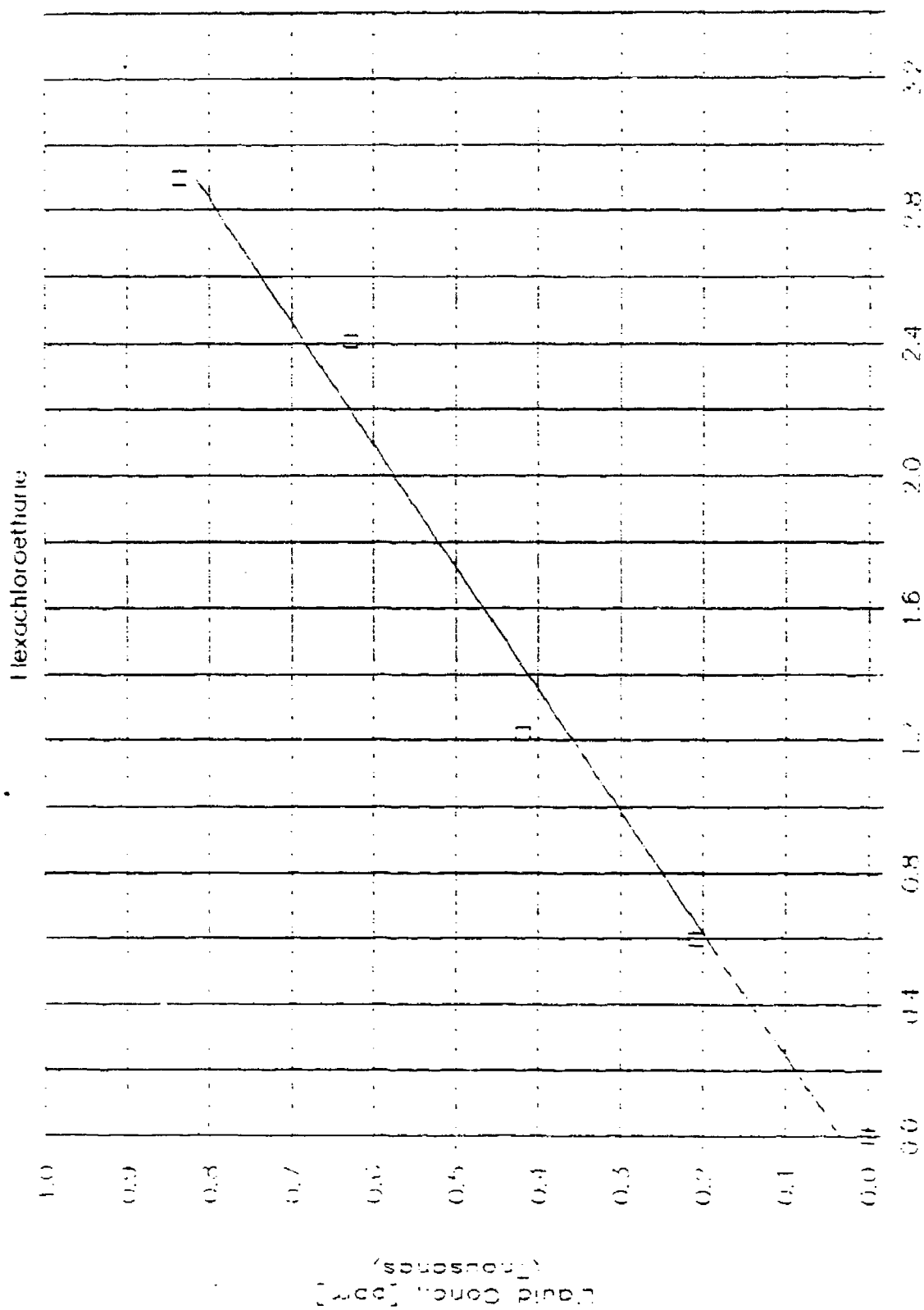
AQUEOUS SOLUBILITY CALIBRATION



U.V. Detector Scale
Calc. r. sq. = 0.956

U Anthracene

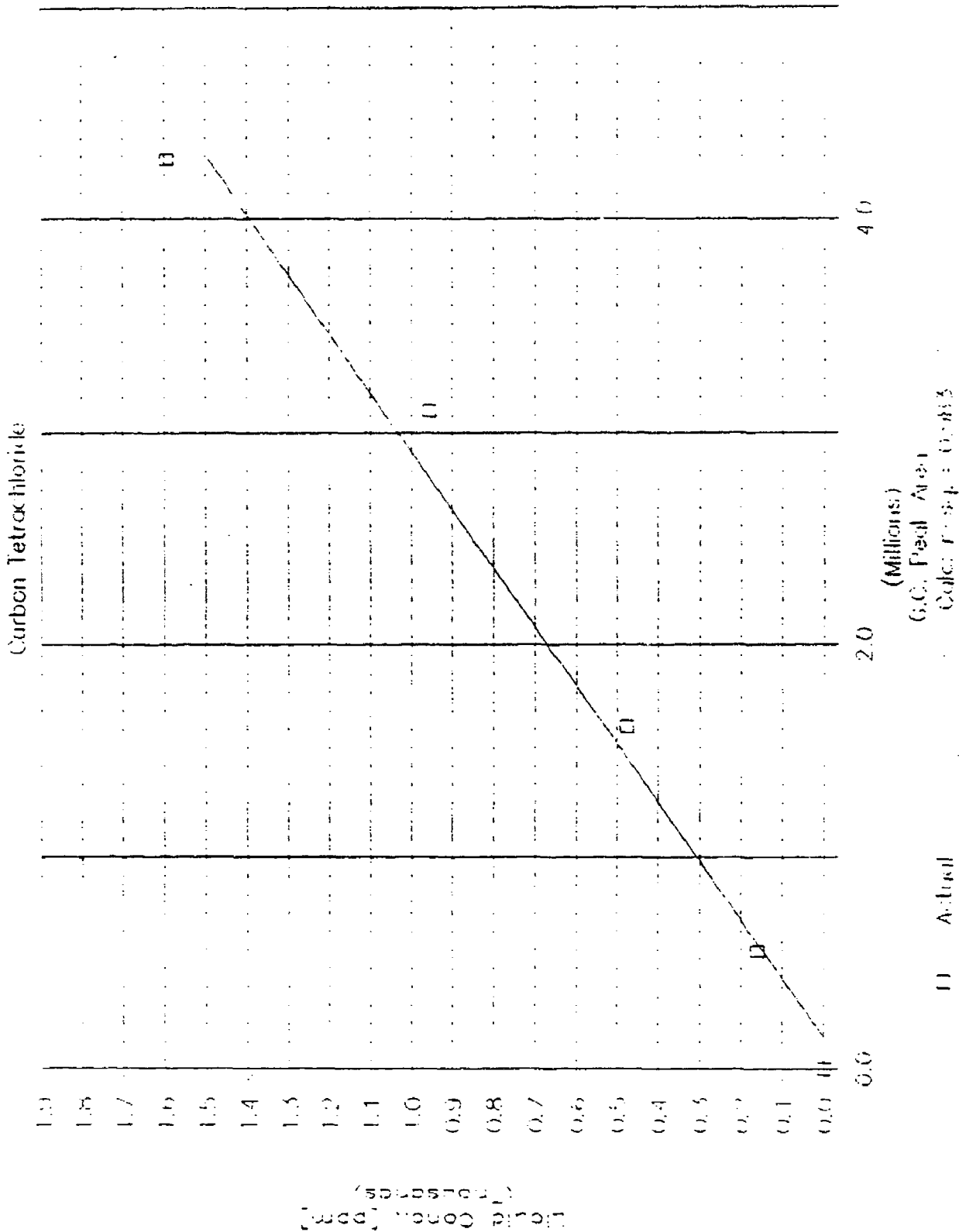
aqueous solubility calibration



G. C. Peak Area
 Calc. $r = -0.982$

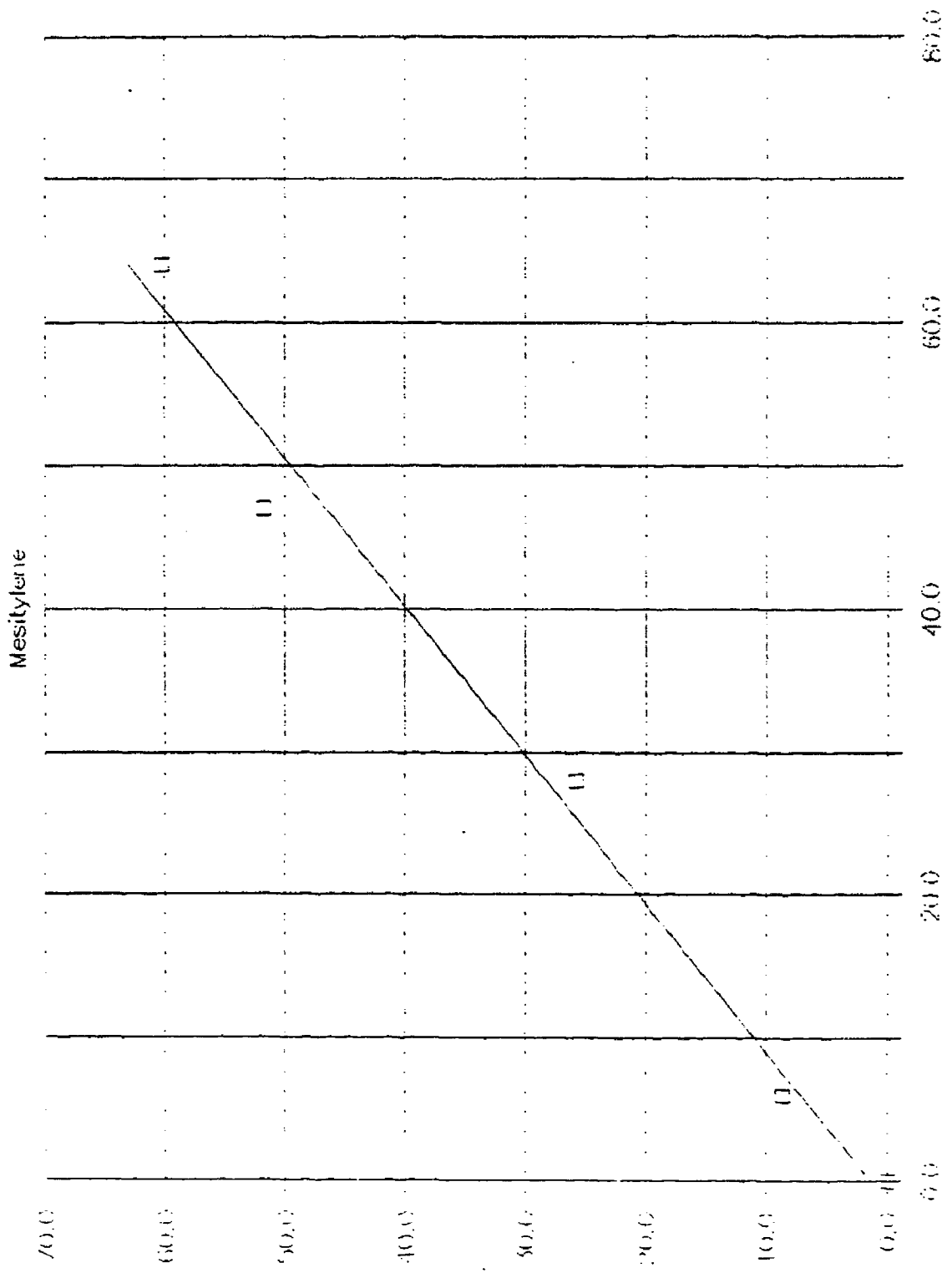
|| Actual

AQUEOUS SOLUBILITY CALIBRATION



Liquid Conductivity (ppm)

ACQUEOUS SOLUBILITY CALIBRATION

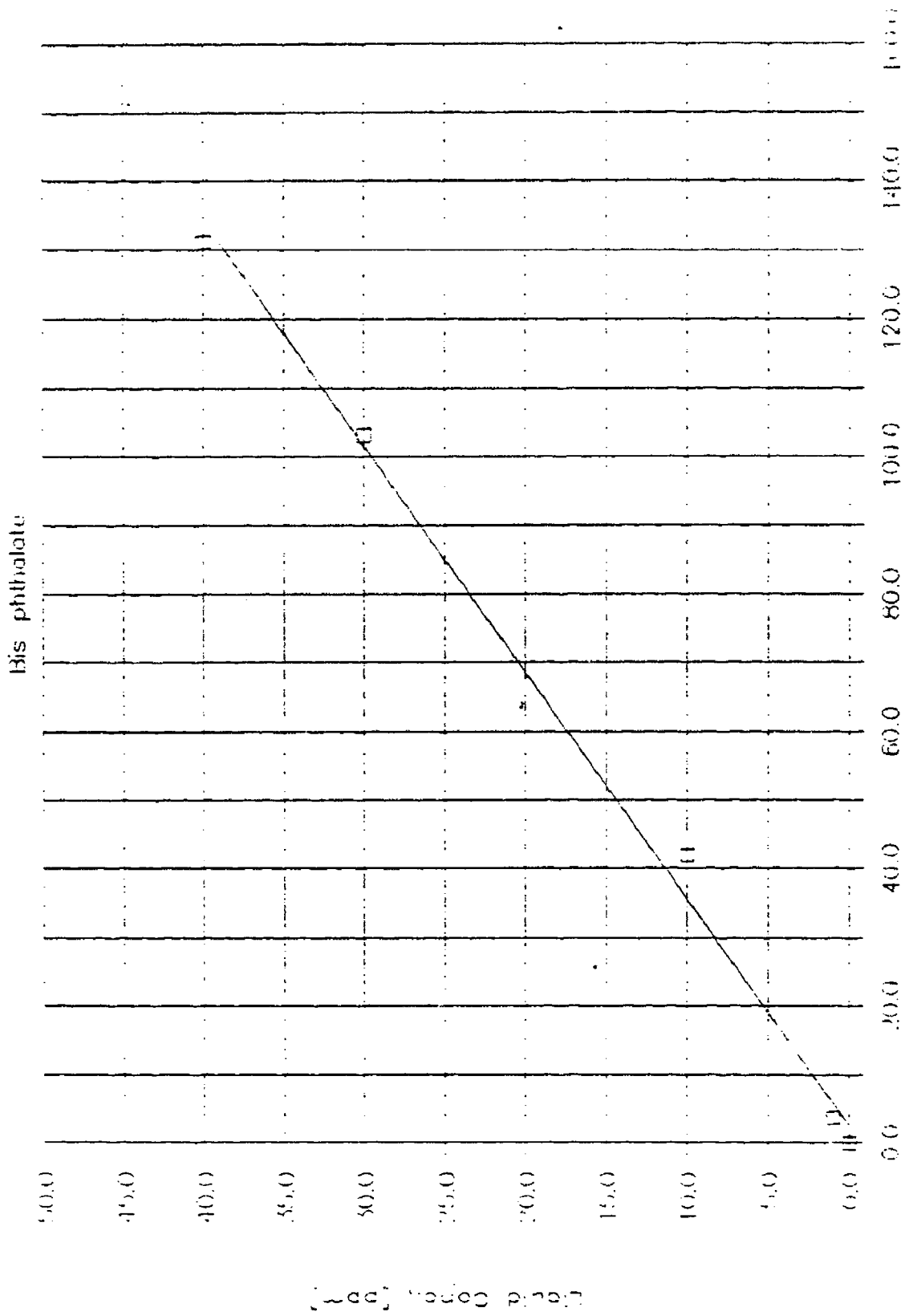


U.V. Detector Scale
Calibration: 0.184

○ Actual

[Liquid Conc., ppm]

AQUEOUS SOLUBILITY CALIBRATION

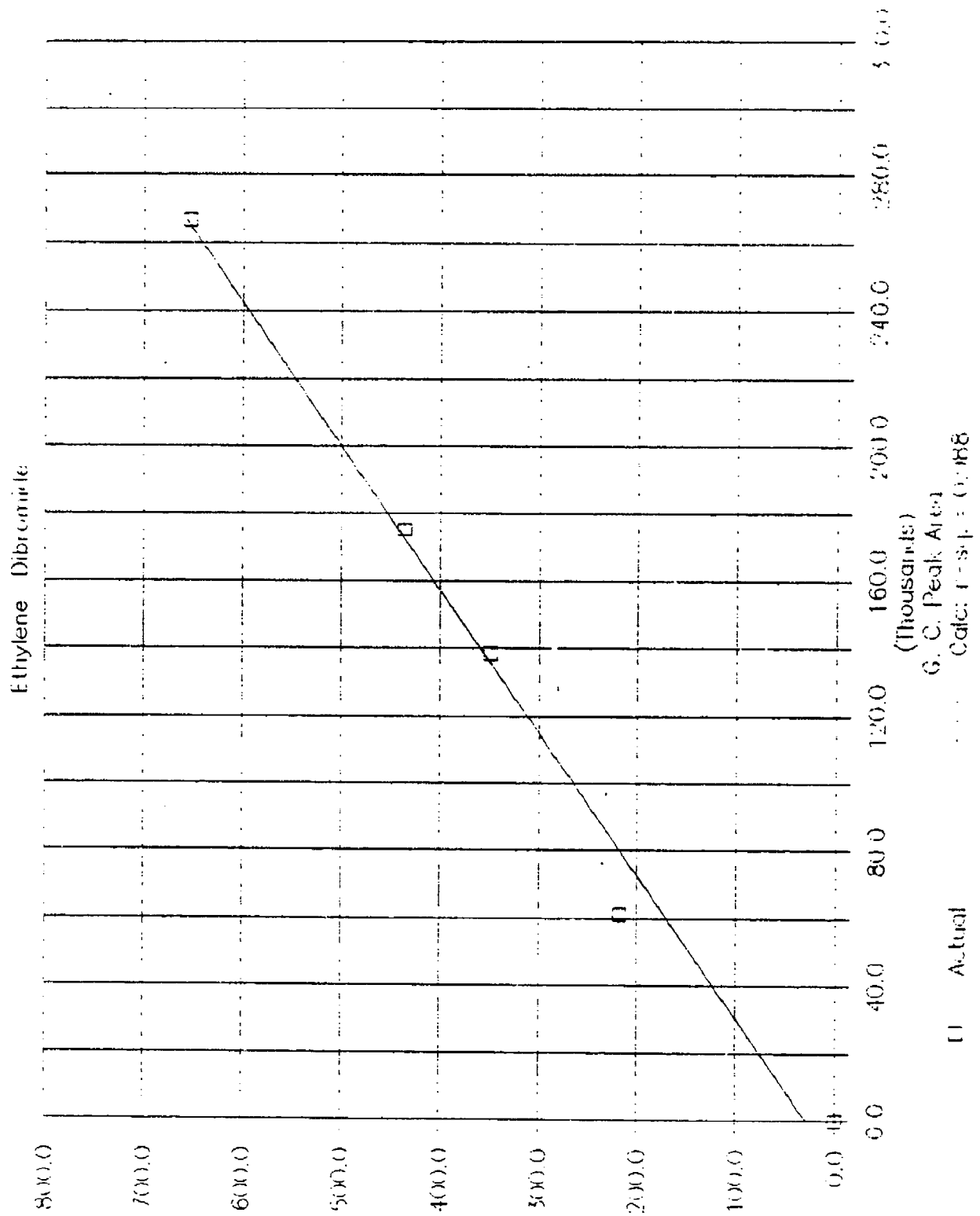


U.V. Detector Scale
Calc. F. = 0.9595

□ Actual

[mg/ml] Bis phthalate

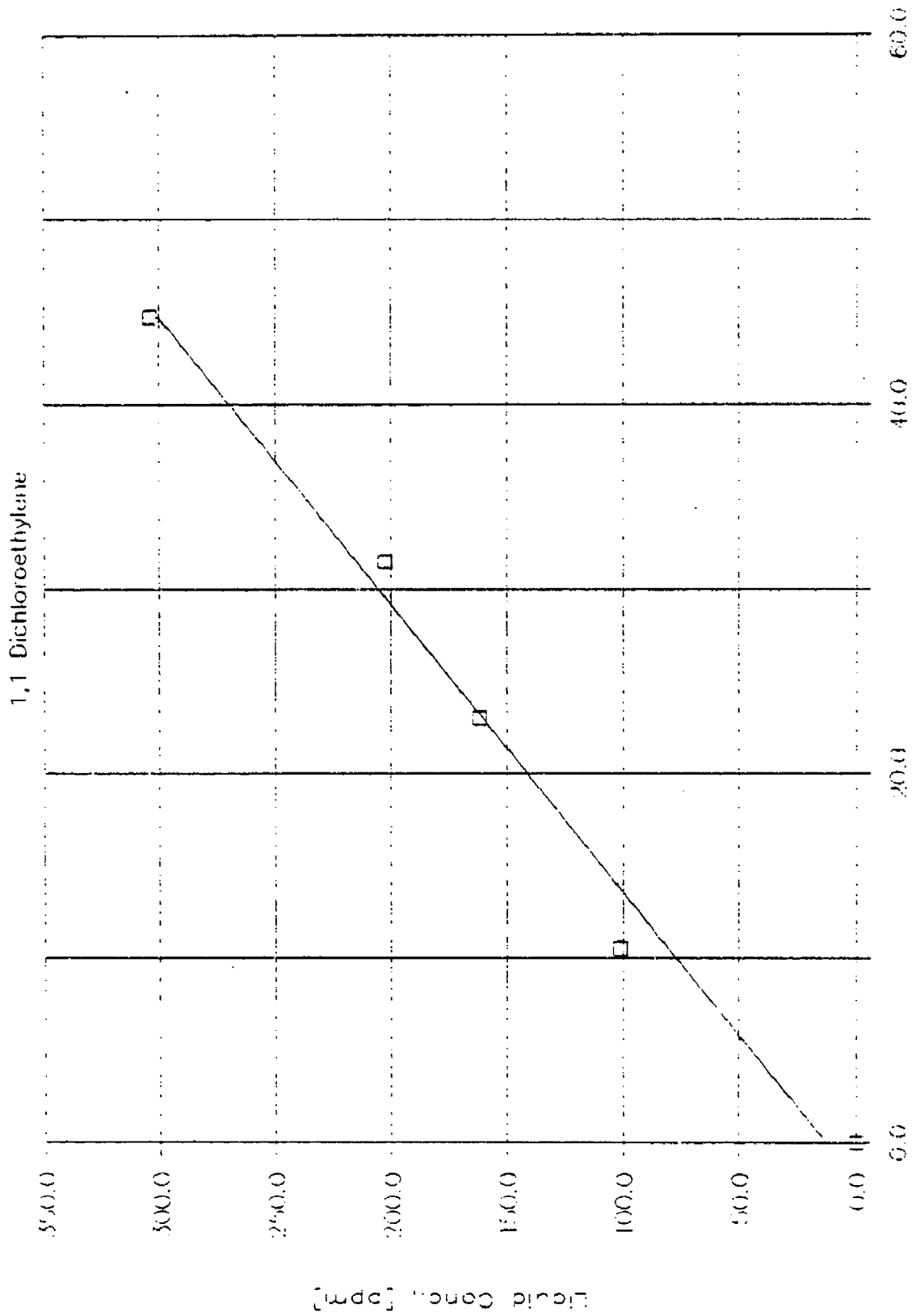
AQUEOUS SOLUBILITY CALIBRATION



Liquid Concentration (ppm)

1,1 Dichloroethylene

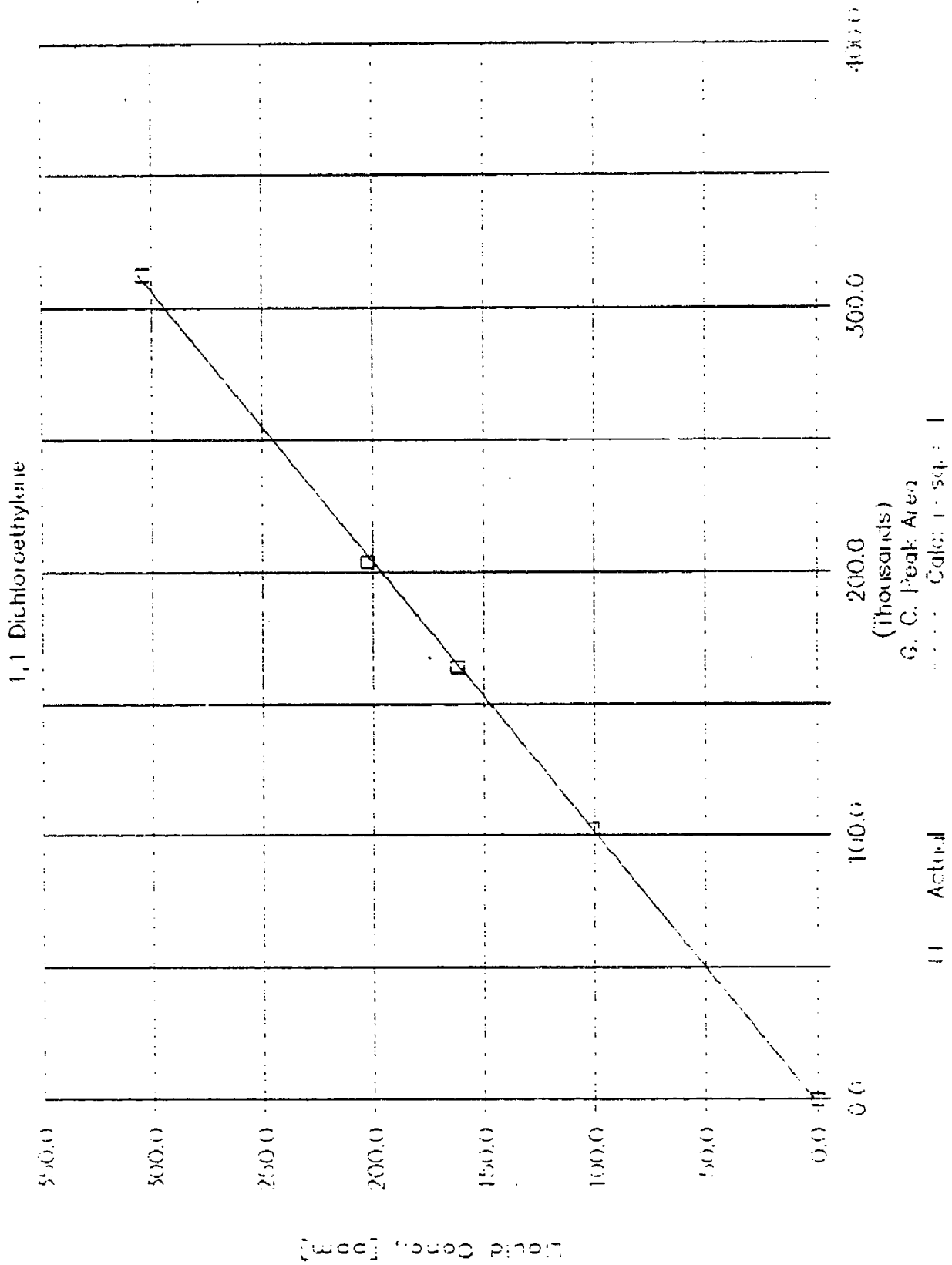
AQUEOUS SOLUBILITY CALIBRATION



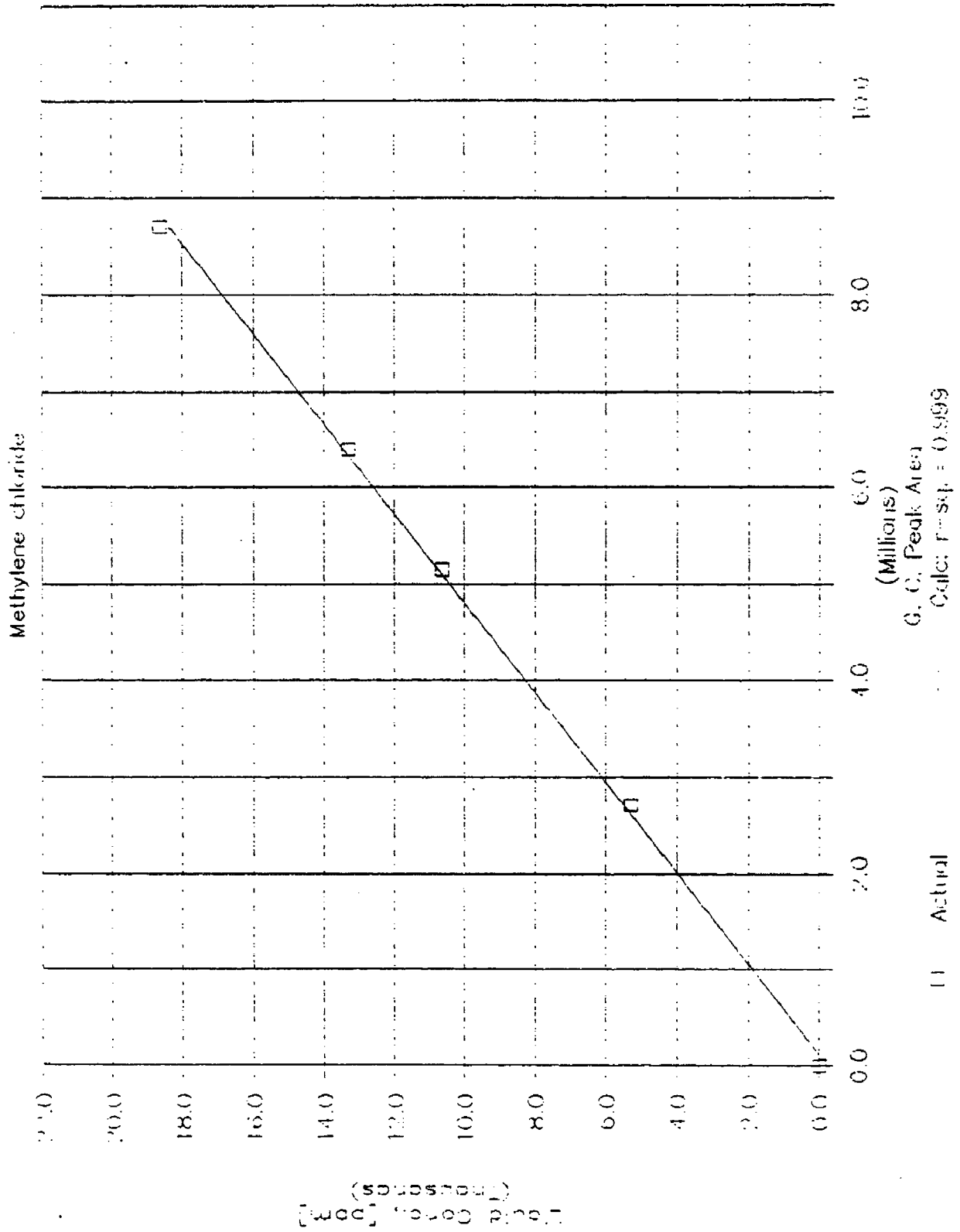
G.C. Peak Area
Calc. r-sq. = 0.985

() Actual

AQUEOUS SOLUBILITY CALIBRATION

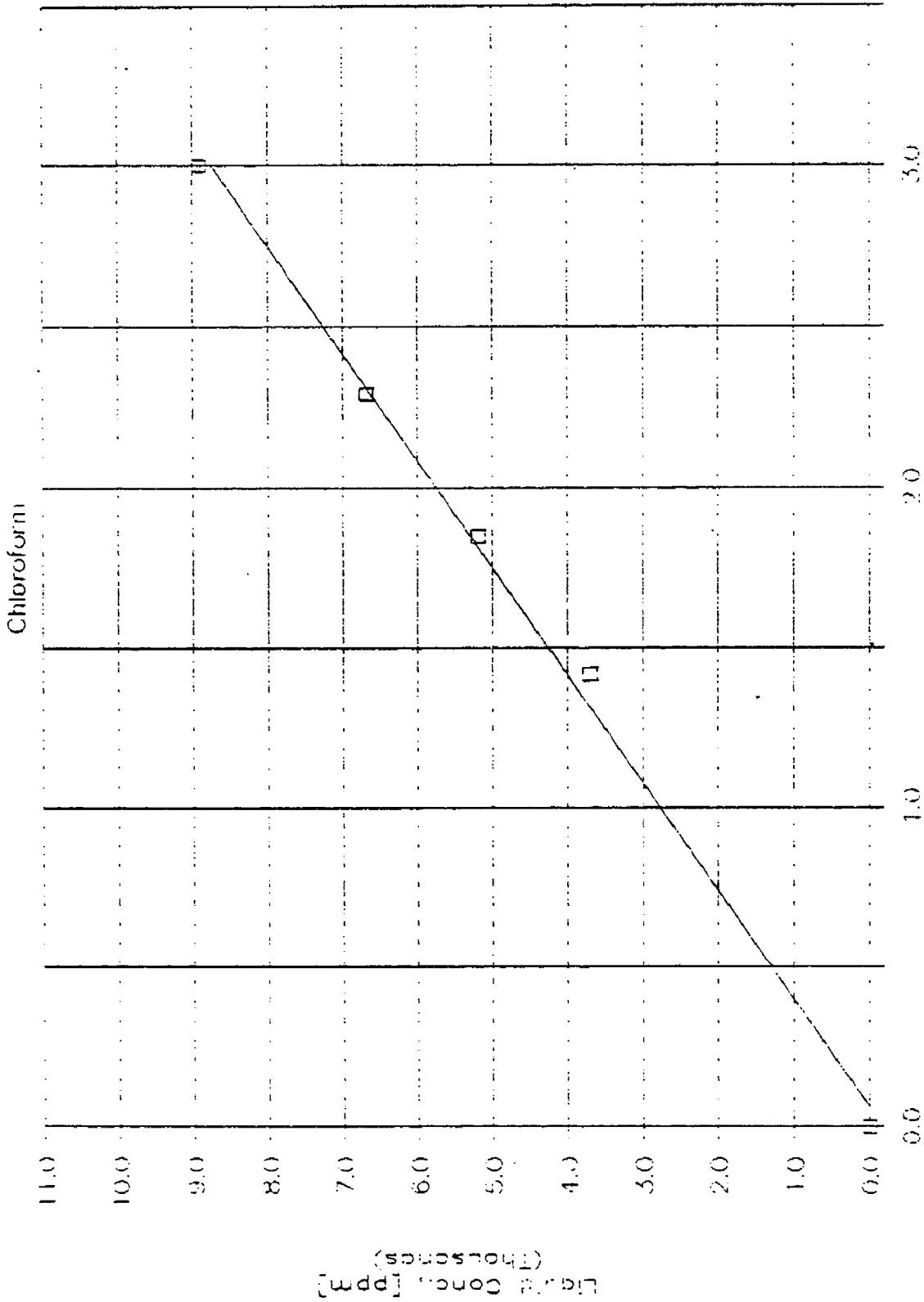


AQUEOUS SOLUBILITY CALIBRATION



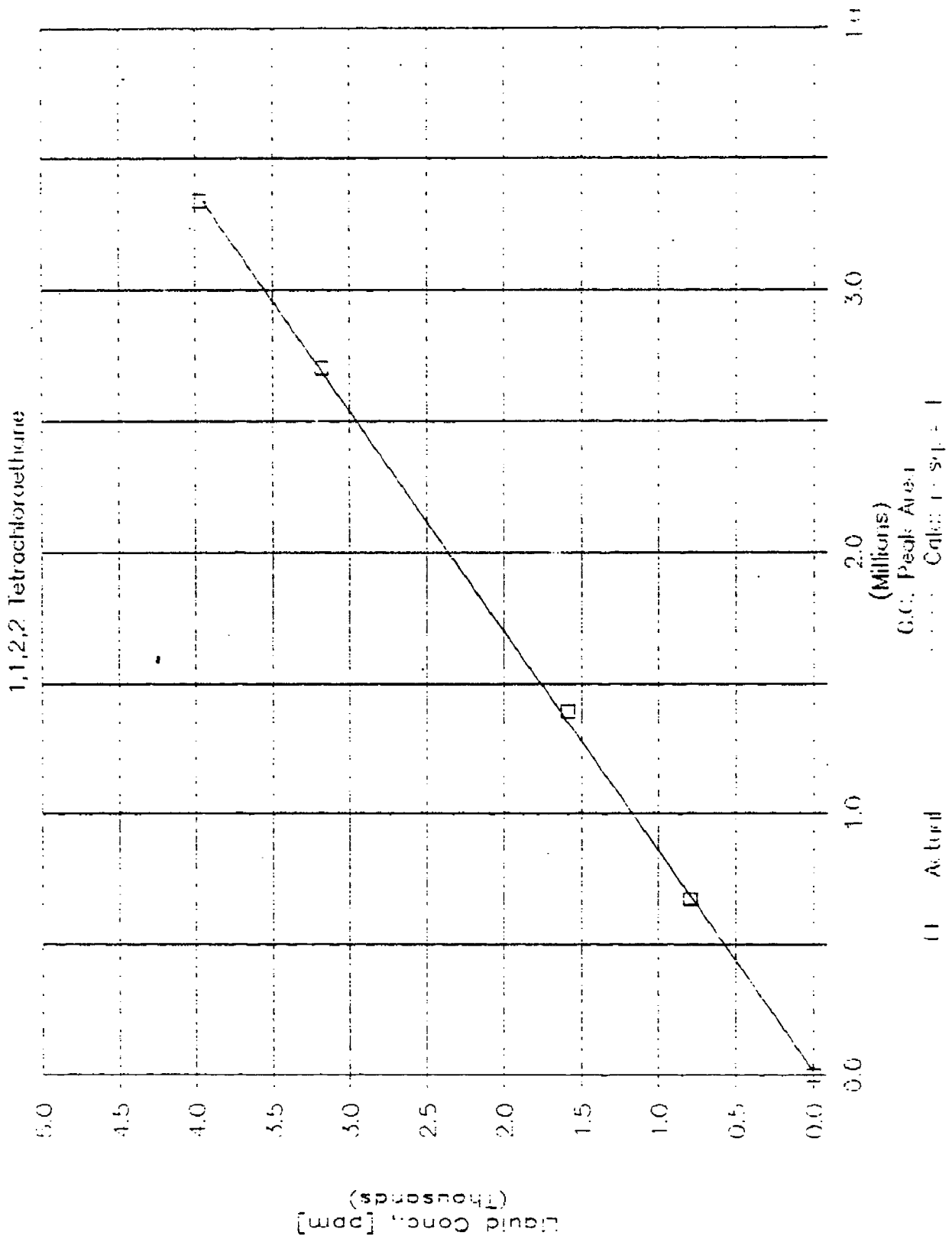
(continued)
[unclear] [unclear]

AQUEOUS SOLUBILITY CALIBRATION

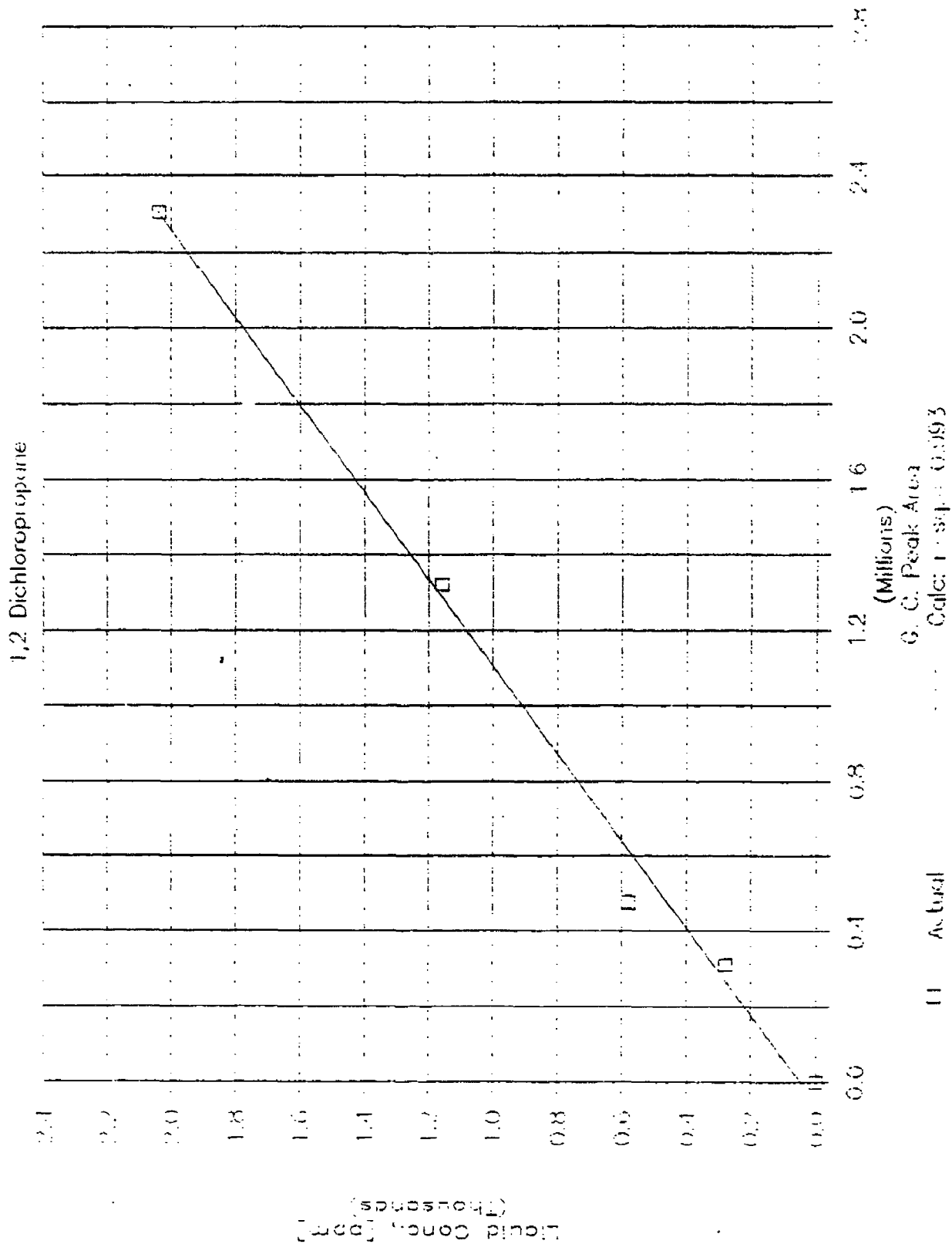


□ Actual
○ C. Peak Area
Calcs: $r = 0.996$

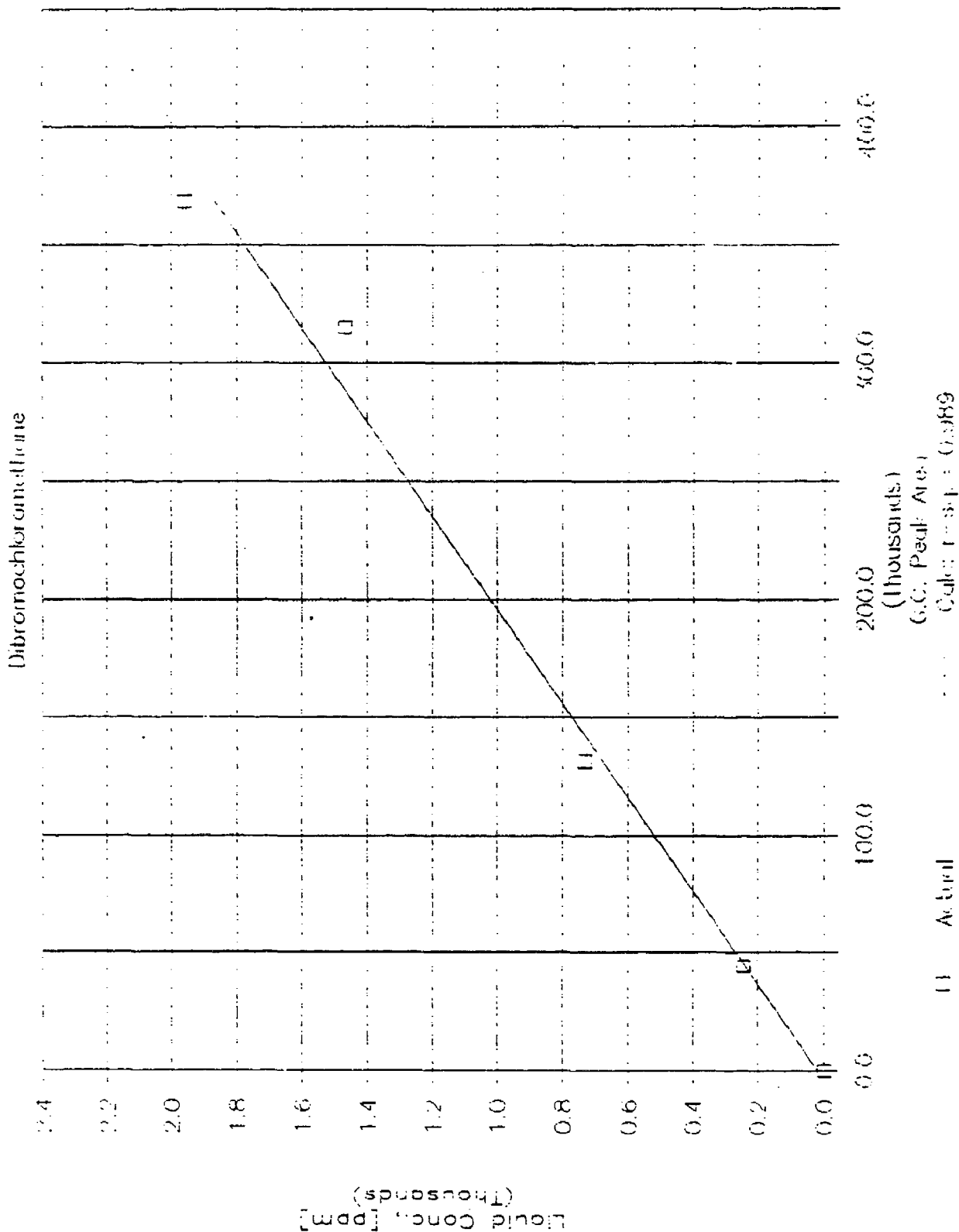
AQUEOUS SOLUBILITY CALIBRATION



AQUEOUS SOLUBILITY CALIBRATION

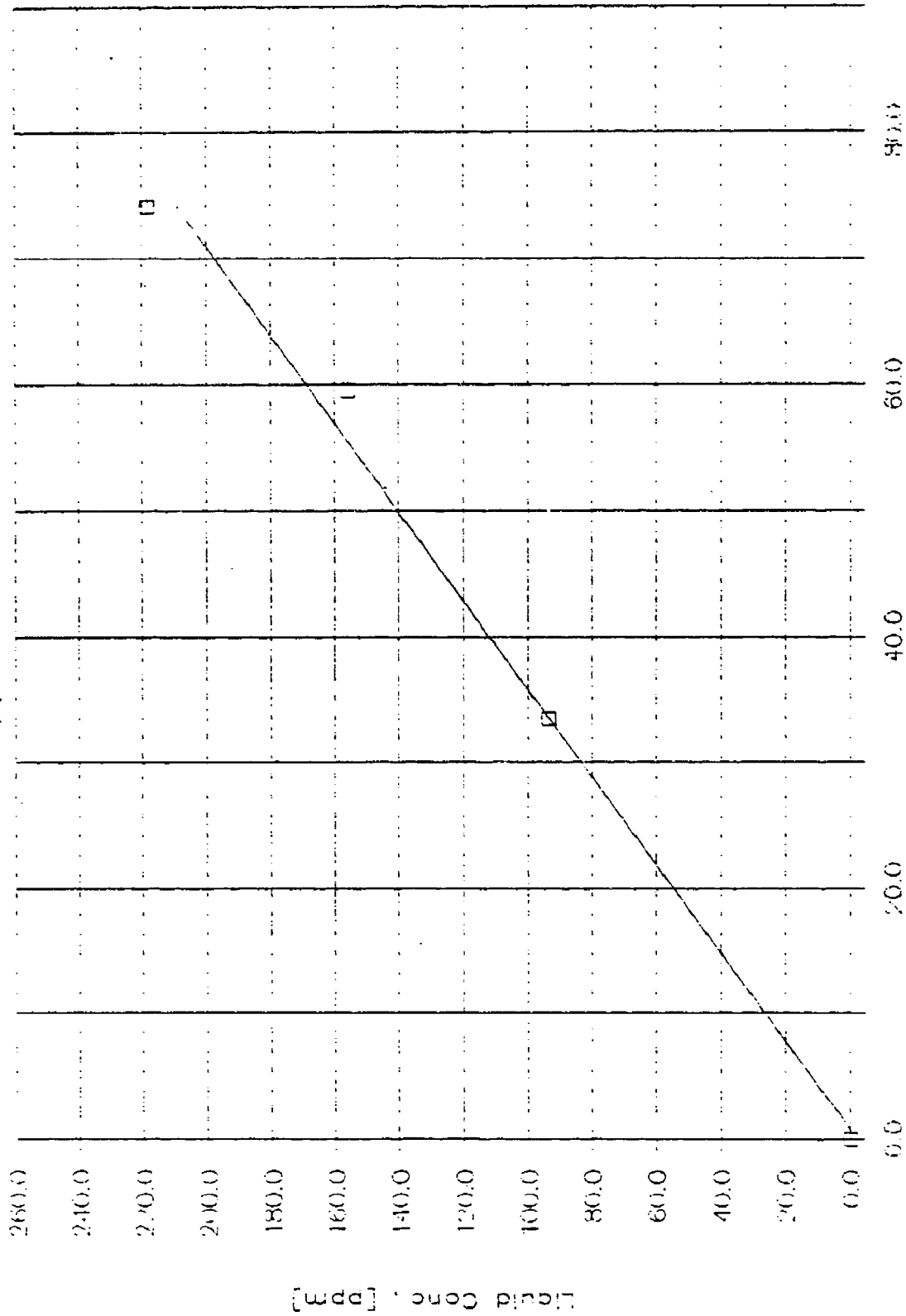


aqueous solubility calibration



AQUEOUS SOLUBILITY CALIBRATION

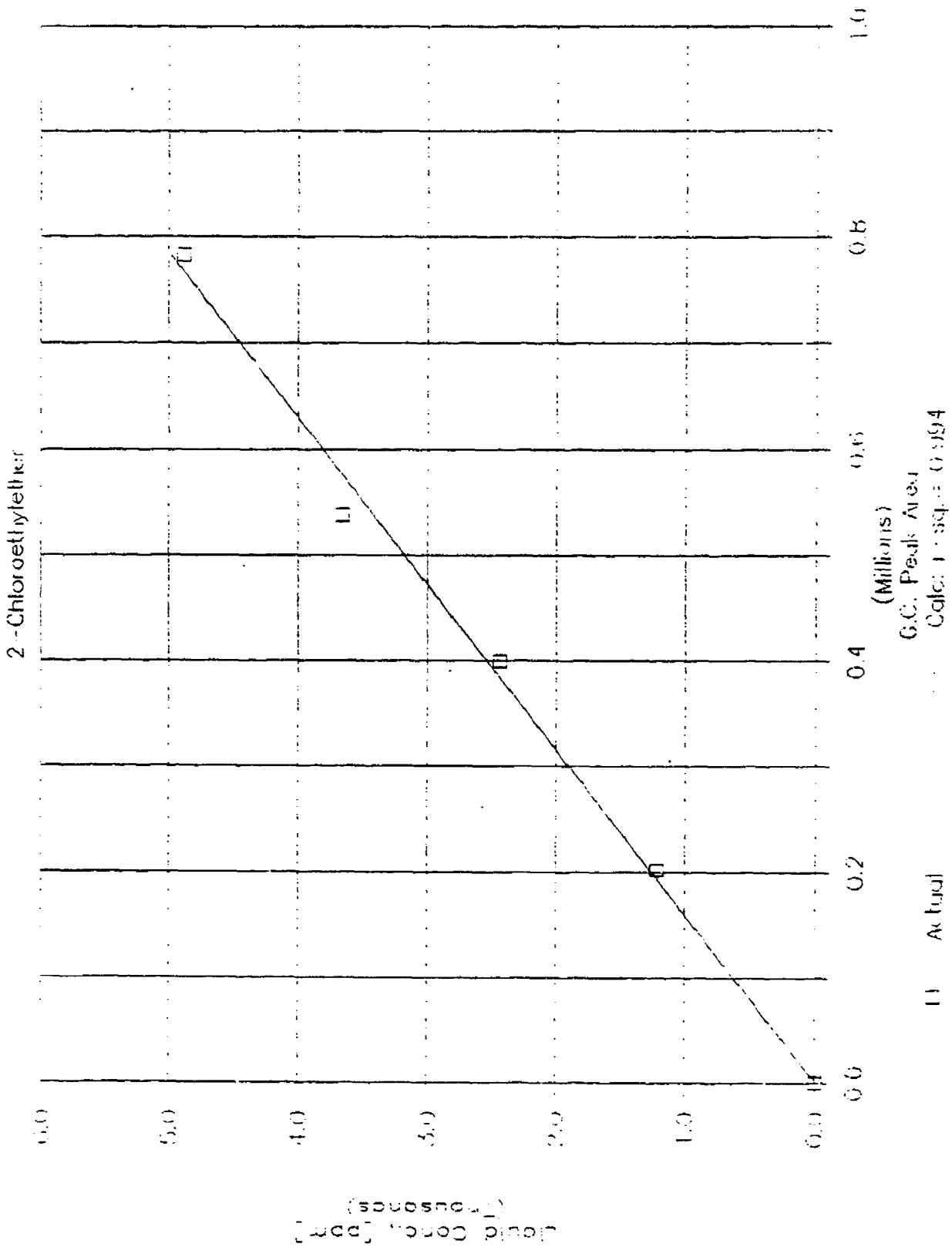
1,1,2 trichlorotrifluoroethane



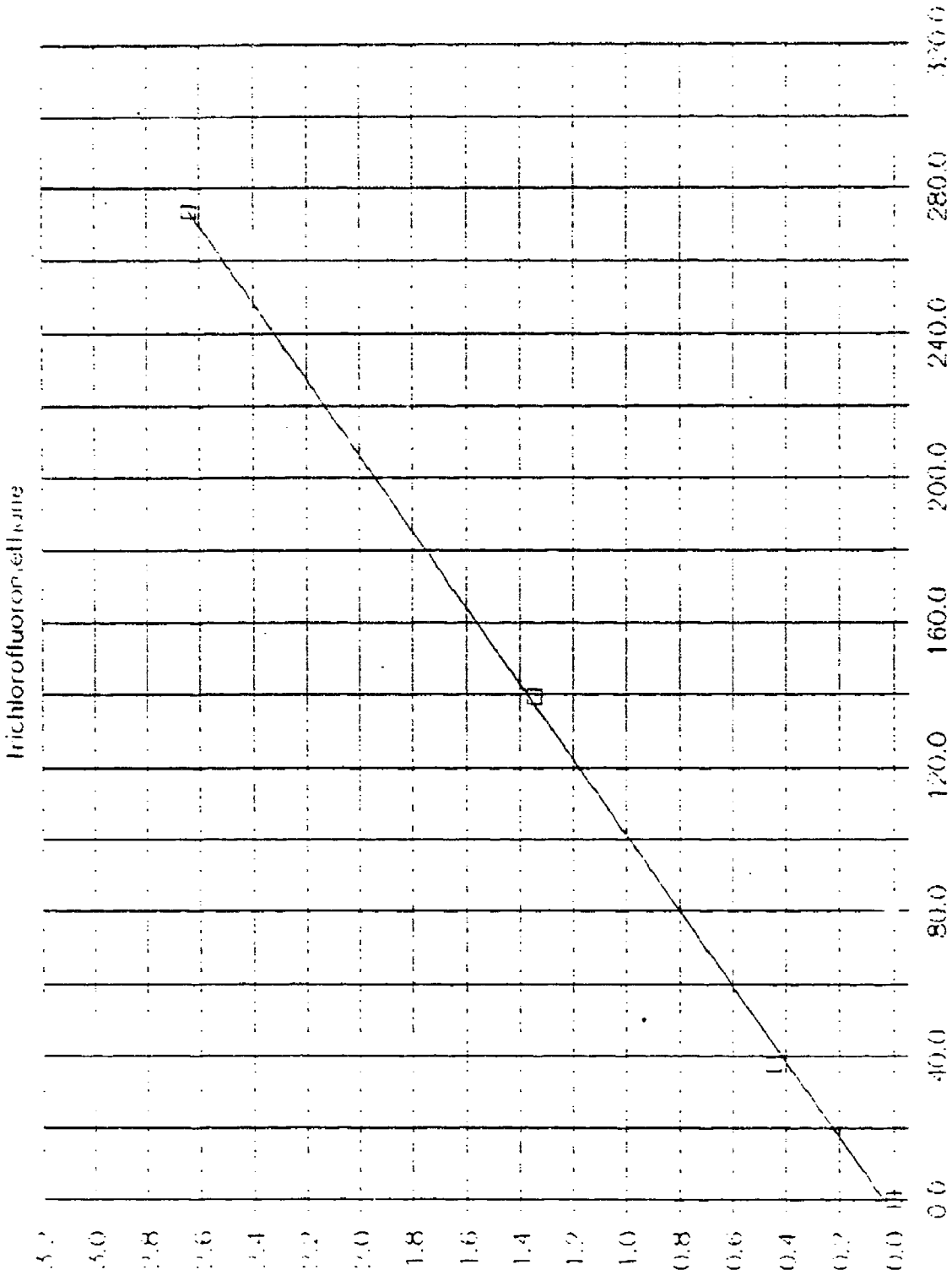
Calc: r = .991

|| Actual

AQUEOUS SOLUBILITY CALIBRATION



AQUEOUS SOLUBILITY CALIBRATION



G. C. Peak Area
Calc: $r^2 = 0.999$

|| Actual

APPENDIX D
SOFTWARE DOCUMENTATION

SOFTWARE DOCUMENTATION

This Appendix contains the Fortran source code for the simplex UNIFAC parameter fitting routine and the interactive program for calculating Henry's law constants and aqueous solubilities. The output from an example simplex fitting run is included for the interested reader. A complete Fortran listing of the UNIFAC VLE and LLE binary interaction databases is also presented in this Appendix to complete the software documentation.

EXAMPLE PARAMETER FIT WITH SIMPLEX ALGORITHM

```
*****
*
* CH2CO/CCOH *
*
*****
```

ACETONE(11)-ETHANOL(20) AT 305.15, CAN.J.RES. 24B,254, (1946)
 ETHANOL(20)-2,BUTANONE(12) AT 1 ATM, ZH.PRIKL.KHIM. 41,589, (1968)
 ISOPROPANOL(1)-2,PENTANONE(16) AT 1 ATM, IEC 45,1803, (1953)

COMPONENT /	GROUPS				
	1	2	17	15	22
11	1	0	0	0	1
20	0	0	1	0	0
12	1	1	0	0	1
16	1	2	0	0	1
1	1	0	0	1	0

GROUP NO	GROUP R	GROUP Q
1	.9011E+00	.8480E+00
2	.6744E+00	.5400E+00
17	.2106E+01	.1972E+01
15	.1878E+01	.1660E+01
22	.1672E+01	.1488E+01

GROUP INTERACTION PARAMETERS

.0000E+00	.0000E+00	.7375E+03	.7375E+03	.4764E+03
.0000E+00	.0000E+00	.7375E+03	.7375E+03	.4764E+03
-.8793E+02	-.8793E+02	.0000E+00	.0000E+00	.1000E+01
-.8793E+02	-.8793E+02	.0000E+00	.0000E+00	.1000E+01
.2676E+02	.2676E+02	.1000E+01	.1000E+01	.0000E+00

INITIAL PARAMETERS 3 5

100.000	100.000
110.000	100.000
100.000	110.000

SIMPLEX ALGORITHM RESULTS

FINAL PARAMETERS
 117.240 36.630
 FMIN= .54424E-01 SD= .70283E-06

	TEMP	NUMBER	X	GAMEXP	GAMCAL	DEV
1	305.15	11	.0750	2.0516	1.8340	-10.6
	305.15	20	.9250	1.0055	1.0038	-.2
2	305.15	11	.5000	1.2187	1.2088	-.8
	305.15	20	.5000	1.2304	1.1903	-3.3
3	305.15	11	.8000	1.0356	1.0330	-.3
	305.15	20	.2000	1.6686	1.5984	-4.2
4	305.15	11	1.0000	1.0000	1.0000	.0
	305.15	20	.0000	2.3166	2.1507	-7.2
5	348.15	20	.2380	1.4546	1.5372	5.7
	348.15	12	.7620	1.0506	1.0463	-.4
6	347.55	20	.3970	1.2497	1.3013	4.1
	347.55	12	.6030	1.1154	1.1306	1.4
7	347.55	20	.6220	1.0965	1.1057	.8
	347.55	12	.3780	1.2825	1.3382	4.3
8	348.45	20	.7860	1.0234	1.0321	.8
	348.45	12	.2140	1.5091	1.5754	4.4
9	350.15	20	.9340	1.0046	1.0030	-.2
	350.15	12	.0660	1.7549	1.8757	6.9
10	371.37	1	.0755	1.5221	1.5656	2.9
	371.37	16	.9245	.9678	1.0028	3.6
11	361.71	1	.3720	1.2395	1.2481	.7
	361.71	16	.6280	1.0060	1.0742	6.8
12	358.59	1	.5845	1.0974	1.1070	.9
	358.59	16	.4155	1.1203	1.2018	7.3
13	356.76	1	.7815	1.0257	1.0298	.4
	356.76	16	.2185	1.2816	1.4073	9.8

\$DEBUG

C
C
C
C
C
C
C
C
C
C
C
C

```
*****  
*  
* UNIFAC PARAMETER ESTIMATION BY MEANS OF  
* NELDER-MEAD'S EXTENDED SIMPLEX MINIMIZATION METHOD  
*  
*****
```

```
IMPLICIT REAL*8(A-H,O-Z)  
DIMENSION R(10),RS(10),QS(10),XL(10),KG(10),NTEXT(40),NUM(10),  
&IPAR(4),PAR(4),DEV(100,2),JENS(10),NU(100,2)  
DIMENSION RI(76),QI(76),AI(40,40),MAIN(76)  
DIMENSION X(5,4),F(5),XB(4),XS(4),XM(4),XE(4),XX(4),XR(4),XK(4)  
COMMON /U/ T(100),NM(100,2),XXX(100,2),GME(100,2),GMC(100,2),  
&GMR(100,2),NNY(10,10),Q(10),A(10,10)  
DATA NC,NP/5,6/
```

C

```
OPEN(6,FILE='SIMPLEX.DAT',STATUS='OLD',ACCESS='SEQUENTIAL',  
+FORM='FORMATTED')  
OPEN(7,FILE='SIMPLEX.OUT',STATUS='NEW',ACCESS='SEQUENTIAL',  
+FORM='FORMATTED')
```

C

```
READ(6,16) NCOMP,NG,NOBS,LAAF,NOIT,MDA  
16 FORMAT(20I3)
```

C
C
C
C
C
C
C
C
C
C
C
C
C
C
C

```
NCOMP= NUMBER OF COMPONENTS  
NG = NUMBER OF DIFFERENT GROUPS  
NOBS = NUMBER OF DATA POINTS  
LAAF, LAAF=1 THE EXPERIMENTAL ACTIVITY COEFFICIENTS  
ARE READ  
LAAF=2 THE LOGARITHMS TO THE EXPERIMENTAL  
ACTIVITY COEFFICIENTS ARE READ  
NOIT, NOIT=1 PARAMETER ESTIMATION IS PERFORMED  
NOIT=2 NO PARAMETER ESTIMATION. THE ACTIVITY  
COEFFICIENTS ARE CALCULATED BASED ON  
THE GIVEN R,Q AND A MATRIX  
MDA, MDA=1 BUILT-IN VLE PARAMETERS ARE LOADED  
MDA=2 BUILT-IN LLE PARAMETERS ARE LOADED
```

```
IF(NOIT.EQ.2) GOTO 1301  
WRITE(7,15)  
WRITE(*,15)  
15 FORMAT(1H , ' PARAMETER ESTIMATION')  
GOTO 1302  
1301 WRITE(7,1303)  
WRITE(*,1303)  
1303 FORMAT(1H , ' CALCULATION OF ACTIVITY COEFFICIENTS BASED ON THE  
&GIVEN R, Q, AND A MATRIX')  
1302 CONTINUE  
DO 14 I=1,16
```



```

READ(6,13) NTEXT
WRITE(*,13) NTEXT
14 WRITE(7,13) NTEXT
13 FORMAT(40A2)
READ(6,16) (NUM(I), I=1, NCOMP)
C
C     NUM GIVES THE NUMBERS ATTACHED TO
C     THE DIFFERENT COMPONENTS
C
DO 1304 I=1, NCOMP
1304 READ(6,2) (NNY(I,K), K=1, NG)
2 FORMAT(40I2)
C
C     NNY(I,K) IS THE MATRIX GIVING THE NUMBER OF GROUPS OF
C     KIND K IN MOLECULE I
C
READ(6,16) (KG(K), K=1, NG)
C
C     KG(K) IS THE NUMBER ATTACHED TO GROUP K
C
DO 100 N=1, NOBS
100 READ(6,101) T(N), (NU(N,I), XXX(N,I), GME(N,I), I=1,2)
101 FORMAT(F7.2, 3(I4, 2F7.4))
C
C     N           = DATA POINT NUMBER
C     T(N)        = TEMPERATURE IN K
C     NU(N,I)     = NUMBER ATTACHED TO COMPONENT I
C                 (NU(N,I)=NUM(I))
C     XXX(N,I)    = LIQUID MOLE FRACTION
C     GME(N,I),   LAAF=1 EXPERIMENTAL ACTIVITY COEFFICIENT
C                 LAAF=2 LOGARITHM TO THE EXPERIMENTAL ACTIVITY
C                 COEFFICIENT
C
IF(LAAF-1)820, 820, 821
821 CONTINUE
DO 822 N=1, NOBS
DO 822 I=1, 2
GGG=GME(N,I)
GME(N,I)=DEXP(GGG)
822 CONTINUE
820 CONTINUE
IF(MDA.EQ.1) CALL UVLE(RI, QI, AI, MAIN)
IF(MDA.EQ.2) CALL ULLE(RI, QI, AI, MAIN)
DO 705 I=1, NG
R(I)=RI(KG(I))
Q(I)=QI(KG(I))
DO 705 IU=1, NG
A(I, IU)=AI(MAIN(KG(IU)), MAIN(KG(I)))
705 CONTINUE
READ(6, * ) IOWN
IF(IOWN.EQ.0) GOTO 707
DO 704 IU=1, IOWN
READ(6, * ) KOWNI

```

```

      READ(6,706) R(KOWNI),Q(KOWNI),(A(KOWNI,J),J=1,NG)
706  FORMAT(8F10.4)
704  CONTINUE
707  CONTINUE

```

C
C
C
C
C
C
C
C
C
C

```

      R(I) IS THE GROUP VOLUME OF GROUP I
      Q(I) IS THE GROUP AREA OF GROUP I
      A(I,J) IS THE GROUP INTERACTION PARAMETER
      BETWEEN GROUPS I AND J
      IOWN IS THE NUMBER OF SETS OF USER-SUPPLIED
      R, Q, AND A(I,J) DATA
      KOWNI IS THE GROUP ID NUMBER FOR A SET OF
      USER-SUPPLIED DATA

      DO 7 I=1,NCOMP
      RS(I)=0.DO
      QS(I)=0.DO
      DO 8 J=1,NG
      RS(I)=RS(I)+NNY(I,J)*R(J)
      QS(I)=QS(I)+NNY(I,J)*Q(J)
      8  CONTINUE
      7  XL(I)=5.DO*(RS(I)-QS(I))-RS(I)+1.DO
      WRITE(7,5)
      WRITE(*,5)
      5  FORMAT (1H , ' COMPONENT / GROUPS',/)
      WRITE(7,12) (KG(I),I=1,NG)
      WRITE(*,12) (KG(I),I=1,NG)
      12  FORMAT (17X,10I3)
      DO 10 I=1,NCOMP
      WRITE(*,6) NUM(I),(NNY(I,J),J=1,NG)
      10  WRITE(7,6) NUM(I),(NNY(I,J),J=1,NG)
      6  FORMAT(10X,I3,4X,10I3)
      WRITE(7,3)
      WRITE(*,3)
      3  FORMAT(//, ' GROUP NO GROUP R GROUP Q',/)
      DO 11 I=1,NG
      WRITE(*,4) KG(I),R(I),Q(I)
      11  WRITE(7,4) KG(I),R(I),Q(I)
      4  FORMAT(4X,I3,5X,2E12.4)
      WRITE(7,1400)
      WRITE(*,1400)
      1400  FORMAT(//, ' GROUP INTERACTION PARAMETERS',/)
      DO 1401 I=1,NG
      WRITE(*,1402) (A(I,J),J=1,NG)
      1401  WRITE(7,1402) (A(I,J),J=1,NG)
      1402  FORMAT(10E12.4)
      DO 102 N=1,NOBS
      DO 102 I=1,2
      DO 104 J=1,NCOMP
      IF(NU(N,I)-NUM(J))104,103,104
      103  NM(N,I)=J
      GOTO 102
      104  CONTINUE

```

```

102 CONTINUE
    CALL PFAC3(RS, QS, XL, NOBS)
    IF(NOIT-1)830, 830, 831
831 CONTINUE
    CALL PFAC4(NG, NOBS)
    DO 832 NR=1, NOBS
    DO 832 I=1, 2
    GMR(NR, I)=GMC(NR, I)+GMR(NR, I)
    GMR(NR, I)=DEXP(GMR(NR, I))
832 CONTINUE
    GOTO 833
830 CONTINUE
    READ(6, 16) NPAR, KRIT, IDEN
C
C     NPAR IS THE NUMBER OF PARAMETERS TO
C     BE ESTIMATED
C
C     KRIT DETERMINES THE OBJECTIVE FUNCTION
C     IF KRIT=1, THE SUM OF THE SQUARED DIFFERENCES BETWEEN
C     THE EXPERIMENTAL AND CALCULATED ACTIVITY COEFFICIENTS
C     IS MINIMIZED
C     IF KRIT=2, THE LOGARITHMS OF THE ACTIVITY COEFFICIENTS
C     ARE USED
C
C     IDEN IS THE NUMBER OF IDENTICAL PAIRS
C     OF INTERACTION PARAMETERS
C
    READ(6, 16) (IPAR(I), I=1, NPAR)
C
C     IPAR IS THE VECTOR INDICATING THE
C     PARAMETERS TO BE ESTIMATED IN THE
C     A MATRIX
C
    IF(IDEN)950, 950, 951
951 IDEN=2*IDEN
    READ(6, 16) (JENS(J), J=1, IDEN)
950 CONTINUE
C
C     JENS IS THE VECTOR INDICATING THE IDENTICAL
C     PAIRS OF PARAMETERS
C
    NN=NPAR+1
    N=NPAR
    READ(6, 400) (X(1, I), I=1, NPAR)
C
C     X(1, I)= ROW OF INITIAL PARAMETERS
C     NO INITIAL PARAMETER MUST BE ZERO
C
    SA=1.D-6
C
C     SA IS THE STANDARD ERROR AS DEFINED BY
C     NELDER-MEAD
C

```

```

DO 201 J=2, NN
DO 201 I=1, N
IF(J-I-1) 202, 203, 202
203 X(J, I)=1.1DO*X(1, I)
GOTO 201
202 X(J, I)=X(1, I)
201 CONTINUE
WRITE(7, 300) (IPAR(I), I=1, N)
WRITE(*, 300) (IPAR(I), I=1, N)
300 FORMAT(/, ' INITIAL PARAMETERS', 4I3/)
DO 204 J=1, NN
WRITE(*, 400) (X(J, I), I=1, N)
204 WRITE(7, 400) (X(J, I), I=1, N)
DO 1 J=1, NN
DO 21 I=1, N
21 XX(I)=X(J, I)
CALL FMIN(NPAR, IPAR, PAR, NOBS, NG, XX, FF, KRIT, JENS, IDEN)
1 F(J)=FF
NF=NN

```

C
C
C

NF IS THE NUMBER OF CALCULATIONS OF F

```

ALFA=1. DO
BETA=0. 5DO
GAMMA=2. DO
ITER=0
JPR=0
400 FORMAT(8F10. 3)

```

C
C
C

ESTIMATION OF THE LOWEST VALUE OF F=FB

```

25 FB=F(1)
DO 98 I=1, N
98 XB(I)=X(1, I)
JB=1
DO 31 J=2, NN
IF(FB-F(J)) 31, 31, 108
108 FB=F(J)
JB=J
DO 41 I=1, N
41 XB(I)=X(J, I)
31 CONTINUE

```

C
C
C

ESTIMATION OF THE HIGHEST VALUE OF F=FS

```

FS=F(1)
DO 51 I=1, N
51 XS(I)=X(1, I)
JS=1
DO 61 J=2, NN
IF(FS-F(J)) 111, 61, 61
111 FS=F(J)
JS=J

```

```

DO 71 I=1,N
71 XS(I)=X(J,I)
61 CONTINUE
C
C      CALCULATION OF THE CENTROID XM(I) OF POINTS
C      EXCLUDING XS(I)
C
DO 81 I=1,N
81 XM(I)=-XS(I)
DO 9 J=1,NN
DO 122 I=1,N
122 XM(I)=XM(I)+X(J,I)
9 CONTINUE
DO 121 I=1,N
121 XM(I)=XM(I)/DBLE(N)
C
C      REFLECTION
C
DO 131 I=1,N
131 XR(I)=XM(I)+ALFA*(XM(I)-XS(I))
CALL FMIN(NPAR,IPAR,PAR,NOBS,NG,XR,FR,KRIT,JENS,IDEN)
NF=NF+1
C
C      EXPANSION
C
IF(FR-FB)141,151,151
141 DO 161 I=1,N
161 XE(I)=XM(I)+GAMMA*(XR(I)-XM(I))
CALL FMIN(NPAR,IPAR,PAR,NOBS,NG,XE,FE,KRIT,JENS,IDEN)
NF=NF+1
IF(FE-FB)17,18,18
17 DO 19 I=1,N
X(JS,I)=XE(I)
19 XS(I)=XE(I)
F(JS)=FE
C
C      CALCULATION OF THE HALTING CRITERION
C
27 FM=0.DO
DO 20 J=1,NN
20 FM=FM+F(J)
FM=FM/DBLE(NN)
FRMS=0.DO
DO 22 J=1,NN
22 FRMS=(F(J)-FM)**2+FRMS
RMS=DSQRT(FRMS/DBLE(N))
ITER=ITER+1
JPR=JPR+1
IF(ITER-200)500,500,23
500 CONTINUE
IF(JPR-1)902,902,903
903 CONTINUE
IF(JPR-6)901,904,904

```

```

904 JPR=1
902 CONTINUE
    WRITE(7,107) ITER,NF
    WRITE(*,107) ITER,NF
107 FORMAT(/,,'  ITERATION',I4,'  NUMBER OF CALLS FOR THE SUBROUTIN
    &E',I5)
    WRITE(7,109)
    WRITE(*,109)
109 FORMAT('  PARAMETERS')
    WRITE(7,400) (X(JS,I),I=1,N)
    WRITE(*,400) (X(JS,I),I=1,N)
    WRITE(7,106) F(JS),RMS
    WRITE(*,106) F(JS),RMS
106 FORMAT(1H,'  FMIN=',E14.5,'  SD=',E14.5)
901 CONTINUE
    IF(RMS-SA)23,23,25

C
C      NEW SIMPLEX
C      FE GREATER THAN FB
C
18 DO 26 I=1,N
    X(JS,I)=XR(I)
26 XS(I)=XR(I)
    F(JS)=FR
    FS=FR
    GOTO 27

C
C      NEW SIMPLEX
C      FR GREATER THAN FB
C
151 DO 30 J=1,NN
    IF(J-JS)28,30,28
28 IF(FR-F(J))18,18,30
30 CONTINUE
    IF(FR-FS)91,91,32
91 DO 33 I=1,N
    X(JS,I)=XR(I)
33 XS(I)=XR(I)
    F(JS)=FR
    FS=FR
32 DO 34 I=1,N
34 XK(I)=XM(I)+BETA*(XS(I)-XM(I))
    CALL FMIN(NPAR,IPAR,PAR,NOBS,NG,XK,FK,KRIT,JENS,IDEN)
    NF=NF+1

C
C      NEW SIMPLEX
C      AFTER CONTRACTION
C
    IF(FK-FS)35,35,36
35 DO 37 I=1,N
    X(JS,I)=XK(I)
37 XS(I)=XK(I)
    F(JS)=FK

```

```

FS=FK
GOTO 27
36 DO 38 J=1, NN
DO 39 I=1, N
39 X(J, I)=(X(J, I)+XB(I))/2. DO
38 CONTINUE
GOTO 27
23 WRITE(*, 905)
WRITE(7, 905)
905 FORMAT(//, '      FINAL PARAMETERS')
WRITE(7, 400) (X(JS, I), I=1, N)
WRITE(*, 400) (X(JS, I), I=1, N)
WRITE(7, 106) F(JS), RMS
WRITE(*, 106) F(JS), RMS
833 CONTINUE
DO 906 N=1, NOBS
DO 906 I=1, 2
DEV(N, I)=(GMR(N, I)-GME(N, I))*100. DO/GME(N, I)
906 CONTINUE
WRITE(7, 207)
WRITE(*, 207)
207 FORMAT(//, '      TEMP NUMBER      X      GAMEXP      GAMCAL      DEV', /)
DO 222 N=1, NOBS
WRITE(7, 16) N
WRITE(*, 16) N
DO 223 I=1, 2
WRITE(*, 96) T(N), NU(N, I), XXX(N, I), GME(N, I), GMR(N, I), DEV(N, I)
223 WRITE(7, 96) T(N), NU(N, I), XXX(N, I), GME(N, I), GMR(N, I), DEV(N, I)
96 FORMAT(F8. 2, I5, F8. 4, 2F14. 4, F6. 1)
222 CONTINUE
IF(NOIT-1)835, 835, 836
835 CONTINUE
WRITE(7, 1400)
WRITE(*, 1400)
DO 112 I=1, NG
WRITE(*, 1402) (A(I, J), J=1, NG)
112 WRITE(7, 1402) (A(I, J), J=1, NG)
WRITE(7, 996)
WRITE(*, 996)
996 FORMAT(//, '      THE SUM OF THE SQUARED DIFFERENCES BETWEEN THE')
IF(KRIT-1)997, 997, 998
998 WRITE(*, 995)
WRITE(7, 995)
995 FORMAT('      LOGARITHMS TO THE')
997 WRITE(*, 999)
WRITE(7, 999)
999 FORMAT('      EXPERIMENTAL AND CALCULATED ACTIVITY COEFFICIENTS IS
& MINIMIZED')
836 CONTINUE
STOP
END

```

C
C

```

C .....
C *
C *   SUBROUTINE PFAC3
C *
C *
C *   SUBROUTINE PFAC3 CALCULATES THE COMBINATORIAL PART
C *   OF THE ACTIVITY COEFFICIENTS
C *
C .....

```

```

C
C   SUBROUTINE PFAC3(RS, QS, XL, NOBS)
C
C   IMPLICIT REAL*8(A-H, O-Z)
C   DIMENSION THETA(2), PHI(2), RS(10), QS(10), XL(10)
C   COMMON /U/ T(100), NM(100, 2), XXX(100, 2), GME(100, 2), GMC(100, 2),
C   &GMR(100, 2), NNY(10, 10), Q(10), A(10, 10)
C   DO 3 N=1, NOBS
C     SQ=0. DO
C     SR=0. DO
C     SXL=0. DO
C     DO 2 I=1, 2
C       J=NM(N, I)
C       SXL=SXL+XL(J)*XXX(N, I)
C       SQ=SQ+QS(J)*XXX(N, I)
C 2    SR=SR+RS(J)*XXX(N, I)
C     DO 3 I=1, 2
C       J=NM(N, I)
C       THETA(I)=QS(J)/SQ
C       PHI(I)=RS(J)/SR
C       GMC(N, I)=DLOG(PHI(I))+5. DO*QS(J)*DLOG(THETA(I)/PHI(I))+XL(J)
C     &-PHI(I)*SXL
C 3    CONTINUE
C     RETURN
C     END

```

```

C .....
C *
C *   SUBROUTINE PFAC4
C *
C *
C *   SUBROUTINE PFAC4 CALCULATES THE RESIDUAL PART
C *   OF THE ACTIVITY COEFFICIENTS
C *
C .....

```

```

C
C   SUBROUTINE PFAC4(NG, NOBS)
C
C   IMPLICIT REAL*8(A-H, O-Z)
C   DIMENSION GMOL(10), ATET(10), ANYK(10), BNYK(10), GK(10, 3), P(10, 10)
C   COMMON /U/ T(100), NM(100, 2), XXX(100, 2), GME(100, 2), GMC(100, 2),

```



```

&GMR(100, 2), NNY(10, 10), Q(10), A(10, 10)
C
C      CALCULATION OF THE PSI MATRIX
C
DO 250 NR=1, NOBS
DO 7 I=1, NG
DO 7 J=1, NG
7 P(I, J)=DEXP(-A(I, J)/T(NR))
C
C      CALCULATION OF GROUP MOLE FRACTIONS
C
DO 105 II=1, 3
IF(II-2)100, 100, 101
100 SNYK=0. DO
C
C      PURE COMPONENT
C
J=NM(NR, II)
DO 12 K=1, NG
12 SNYK=SNYK+DBLE(NNY(J, K))
DO 13 K=1, NG
13 GMOL(K)=DBLE(NNY(J, K))/SNYK
GOTO 102
101 SNYK=0. DO
C
C      MIXTURE
C
DO 2 I=1, 2
J=NM(NR, I)
DO 2 K=1, NG
2 SNYK=SNYK+DBLE(NNY(J, K))*XXX(NR, I)
DO 3 K=1, NG
GNYK=0. DO
DO 4 I=1, 2
J=NM(NR, I)
4 GNYK=GNYK+DBLE(NNY(J, K))*XXX(NR, I)
3 GMOL(K)=GNYK/SNYK
C
C      CALCULATION OF GROUP AREA FRACTIONS
C
102 SNYK=0. DO
DO 5 K=1, NG
5 SNYK=SNYK+Q(K)*GMOL(K)
DO 6 K=1, NG
6 ATET(K)=Q(K)*GMOL(K)/SNYK
C
C      CALCULATION OF GAMMA K
C
DO 9 K=1, NG
ANYK(K)=0. DO
DO 10 M=1, NG
SNYK=0. DO
DO 8 N=1, NG

```



```

C   56 CH2NO2 ... 57 CHNO2 ... 58 ACNO2 ... 59 CS2 ..... 60 CH3SH ....
C   61 CH2SH ..... 62 FURFURAL 63 (CH2OH)2 64 I ..... 65 BR .....
C   66 CH=-C ..... 67 C=-C ..... 68 DMSO ..... 69 ACRY ..... 70 CL(C=-C) .
C   71 ACF ..... 72 DMF-1 ... 73 DMF-2 ... 74 CF3 ..... 75 CF2 .....
C   76 CF .....

```

```

C*****
C

```

```

IMPLICIT REAL*8(A-H,O-Z)
DIMENSION AI(40,40),RR(76),QQ(76),RI(76),QI(76),MAINSG(76),
*      MAIN(76)
DIMENSION A1(40),A2(40),A3(40),A4(40),A5(40),A6(40),A7(40),A8(40),
* A9(40),A10(40),A11(40),A12(40),A13(40),A14(40),A15(40),A16(40),
* A17(40),A18(40),A19(40),A20(40),A21(40),A22(40),A23(40),A24(40),
* A25(40),A26(40),A27(40),A28(40),A29(40),A30(40),A31(40),A32(40),
* A33(40),A34(40),A35(40),A36(40),A37(40),A38(40),A39(40),A40(40)
DATA MAINSG/ 4*1, 5*2, 2*3, 3*4, 1*5, 1*6, 1*7, 1*8, 2*9, 1*10,
*      2*11, 1*12, 4*13, 3*14, 3*15, 2*16, 1*17, 3*18, 2*19, 2*20,
*      3*21, 3*22, 2*23, 1*24, 1*25, 3*26, 1*27, 1*28, 2*29, 1*30,
*      1*31, 1*32, 1*33, 2*34, 1*35, 1*36, 1*37, 1*38, 2*39, 3*40/
DATA RR/0.9011,0.6744,0.4469,0.2195,1.3454,1.1167,1.1173,0.8886,
*0.6605,0.5313,0.3652,1.2663,1.0396,0.8121,1.0,1.4311,0.92,0.8952,
*1.6724,1.4457,0.9980,1.9031,1.6764,1.2420,1.1450,0.9183,0.6908,
*0.9183,1.5959,1.3692,1.1417,1.4337,1.2070,0.9795,1.1865,0.9597,
*1.0600,2.9993,2.8332,2.6670,1.8701,1.6434,1.3013,1.5280,1.4654,
*1.2380,1.0060,2.2564,2.0606,1.8016,2.8700,2.6401,3.3900,1.1562,
*2.0086,1.7818,1.5544,1.4199,2.0570,1.8770,1.6510,3.1680,2.4088,
*1.2640,0.9492,1.2920,1.0613,2.8266,2.3144,0.7910,0.6948,3.0856,
*2.6322,1.4060,1.0105,0.6150/
DATA QQ/0.848,0.540,0.228,0.000,1.176,0.867,0.988,0.676,0.485,
*0.400,0.120,0.968,0.660,0.348,1.200,1.432,1.400,0.680,1.488,
*1.180,0.948,1.728,1.420,1.188,1.088,0.780,0.468,1.100,1.544,
*1.236,0.924,1.244,0.936,0.624,0.940,0.632,0.816,2.113,1.833,
*1.553,1.724,1.416,1.224,1.532,1.264,0.952,0.724,1.988,1.684,
*1.448,2.410,2.184,2.910,0.844,1.868,1.560,1.248,1.104,1.650,
*1.676,1.368,2.481,2.248,0.992,0.832,1.088,0.784,2.472,2.052,
*0.724,0.524,2.736,2.120,1.380,0.920,0.460/
DATA A1/ 0.000, -200.000, 61.130, 76.500, 986.500,
* 697.200, 1318.000, 1333.000, 476.400, 677.000,
* 232.100, 741.400, 251.500, 391.500, 255.700,
* 206.600, 1245.000, 287.770, 597.000, 663.500,
* 35.930, 53.760, 24.900, 104.300, 321.500,
* 661.500, 543.000, 153.600, 184.400, 354.550,
* 3025.000, 335.800, 479.500, 298.900, 526.500,
* 689.000, -0.505, 125.800, 485.300, -2.859/
DATA A2/ 2520.000, 0.000, 340.700, 4102.000, 693.900,
* 1509.000, 634.000, 547.400, 524.500, 9000.000,
* 71.230, 468.700, 289.300, 396.000, 273.600,
* 658.800, 9000.000, 9000.000, 405.900, 730.400,
* 99.610, 337.100, 4584.000, 5831.000, 959.700,
* 542.100, 9000.000, 76.302, 9000.000, 9000.000,
* 9000.000, 9000.000, 9000.000, 523.600, 9000.000,
* 9000.000, 237.300, 9000.000, 320.400, 9000.000/

```

DATA A3/	-11.120,	-94.780,	0.000,	167.000,	636.100,
* 637.350,	903.800,	1329.000,	25.770,	9000.000,	
* 5.994,	9000.000,	32.140,	161.700,	122.800,	
* 90.490,	668.200,	-4.449,	212.500,	537.400,	
* -18.810,	-144.400,	-231.900,	3.000,	538.200,	
* 168.000,	194.900,	52.070,	-10.430,	-64.690,	
* 210.400,	113.300,	-13.590,	9000.000,	169.900,	
* 9000.000,	69.110,	389.300,	245.600,	9000.000/	
DATA A4/	-69.700,	-269.700,	-146.800,	0.000,	803.200,
* 603.250,	5695.000,	884.900,	-52.100,	9000.000,	
* 5688.000,	9000.000,	213.100,	9000.000,	-49.290,	
* 23.500,	764.700,	52.800,	6096.000,	603.800,	
* -114.100,	9000.000,	-12.140,	-141.300,	-126.900,	
* 3629.000,	4448.000,	-9.451,	9000.000,	-20.360,	
* 4975.000,	9000.000,	-171.300,	9000.000,	4284.000,	
* 9000.000,	9000.000,	101.400,	5629.000,	9000.000/	
DATA A5/	156.400,	8694.000,	89.600,	25.820,	0.000,
* -137.100,	353.500,	-259.700,	84.000,	441.800,	
* 101.000,	193.100,	28.060,	83.020,	42.700,	
* -323.000,	-348.200,	170.000,	6.712,	199.000,	
* 75.620,	-112.100,	-98.120,	143.100,	287.800,	
* 61.110,	157.100,	477.000,	147.500,	-120.500,	
* -318.900,	313.500,	133.400,	9000.000,	-202.100,	
* 9000.000,	253.900,	44.780,	-143.900,	9000.000/	
DATA A6/	16.510,	-52.390,	-50.000,	-44.500,	249.100,
* 0.000,	-181.000,	-101.700,	23.390,	306.400,	
* -10.720,	193.400,	-180.600,	359.300,	266.000,	
* 53.900,	335.500,	580.500,	36.230,	-289.500,	
* -38.320,	-102.500,	-139.400,	-67.800,	17.120,	
* 75.140,	9000.000,	-31.090,	37.840,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	-399.300,	
* 9000.000,	-21.220,	-48.250,	-172.400,	9000.000/	
DATA A7/	300.000,	692.700,	362.300,	377.600,	-229.100,
* 289.600,	0.000,	324.500,	-195.400,	-257.300,	
* 14.420,	9000.000,	540.500,	48.890,	168.000,	
* 304.000,	213.000,	459.000,	112.600,	-14.090,	
* 325.400,	37.400,	353.700,	497.500,	678.200,	
* 220.600,	399.500,	887.100,	9000.000,	188.000,	
* 0.000,	9000.000,	9000.000,	9000.000,	-139.000,	
* 160.800,	9000.000,	9000.000,	319.000,	9000.000/	
DATA A8/	275.900,	1665.000,	25.340,	244.200,	-451.600,
* -265.200,	-601.800,	0.000,	-356.100,	9000.000,	
* -449.400,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	-305.500,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	1827.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* -687.100,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
DATA A9/	26.760,	-82.920,	140.100,	365.800,	164.500,
* 108.700,	472.500,	-133.100,	0.000,	-37.360,	
* -213.700,	9000.000,	5.202,	9000.000,	9000.000,	
* 9000.000,	937.900,	165.100,	481.700,	669.400,	
* -191.700,	-284.000,	-354.600,	-39.200,	174.500,	

*	137.500,	9000.000,	216.100,	-46.280,	-163.700,	
*	9000.000,	53.590,	245.200,	-246.600,	-44.580,	
*	9000.000,	-44.420,	9000.000,	-61.700,	9000.000/	
	DATA A10/	505.700,	9000.000,	9000.000,	9000.000,	-404.800,
*	-340.200,	232.700,	9000.000,	128.000,	0.000,	
*	9000.000,	9000.000,	304.100,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	751.900,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
	DATA A11/	114.800,	269.300,	85.840,	-170.000,	245.400,
*	249.630,	10000.000,	-36.720,	372.200,	9000.000,	
*	0.000,	372.900,	-235.700,	9000.000,	-73.500,	
*	9000.000,	9000.000,	9000.000,	494.600,	660.200,	
*	9000.000,	108.900,	-209.700,	54.570,	629.000,	
*	9000.000,	9000.000,	183.000,	9000.000,	202.300,	
*	-101.700,	148.300,	9000.000,	9000.000,	52.080,	
*	9000.000,	-23.300,	9000.000,	9000.000,	9000.000/	
	DATA A12/	90.490,	91.650,	9000.000,	9000.000,	191.200,
*	155.700,	9000.000,	9000.000,	9000.000,	9000.000,	
*	-261.100,	0.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	-356.300,	
*	9000.000,	9000.000,	-287.200,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	4.339,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
	DATA A13/	83.360,	76.440,	52.130,	65.690,	237.700,
*	339.720,	-314.700,	9000.000,	52.380,	-7.838,	
*	461.300,	9000.000,	0.000,	9000.000,	141.700,	
*	9000.000,	9000.000,	9000.000,	9000.000,	664.600,	
*	301.100,	137.800,	-154.300,	47.670,	9000.000,	
*	95.180,	9000.000,	140.900,	-8.538,	9000.000,	
*	-20.110,	-149.500,	-202.300,	9000.000,	172.100,	
*	9000.000,	145.600,	9000.000,	254.800,	9000.000/	
	DATA A14/	-30.480,	79.400,	-44.850,	9000.000,	-164.000,
*	-481.700,	-330.400,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	0.000,	63.720,	
*	-41.110,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	-99.810,	68.810,	
*	9000.000,	9000.000,	9000.000,	-70.140,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
	DATA A15/	65.330,	-41.320,	-22.310,	223.000,	-150.000,
*	-500.400,	-448.200,	9000.000,	9000.000,	9000.000,	
*	136.000,	9000.000,	-49.300,	108.800,	0.000,	
*	-189.200,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	71.230,	4350.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
	DATA A16/	-83.980,	-188.000,	-223.900,	109.900,	28.600,
*	-406.800,	-598.800,	9000.000,	9000.000,	9000.000,	

* 9000.000,	9000.000,	9000.000,	38.890,	865.900,	
* 0.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	-73.850,	-352.900,	-8.283,	-86.360,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
DATA A17/	5339.000,	9000.000,	650.400,	979.800,	529.000,
* 5.182,	-339.500,	9000.000,	-399.100,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	0.000,	9000.000,	-216.800,	9000.000,	
* 9000.000,	9000.000,	9000.000,	8455.000,	699.100,	
* 9000.000,	-62.730,	9000.000,	9000.000,	9000.000,	
* 125.300,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	-293.100,	9000.000/	
DATA A18/	-101.600,	9000.000,	31.870,	49.800,	-132.300,
* -378.200,	-332.900,	-341.600,	-51.540,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	0.000,	-169.700,	-153.700,	
* 9000.000,	-351.600,	-114.700,	-165.100,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
DATA A19/	24.820,	34.780,	-22.970,	-138.400,	185.400,
* 157.800,	242.800,	9000.000,	-287.500,	9000.000,	
* -266.600,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	617.100,	134.300,	0.000,	9000.000,	
* 9000.000,	9000.000,	-15.620,	-54.860,	52.310,	
* 9000.000,	9000.000,	230.900,	21.370,	9000.000,	
* 9000.000,	9000.000,	9000.000,	-203.000,	9000.000,	
* 81.570,	-19.140,	9000.000,	9000.000,	9000.000/	
DATA A20/	315.300,	349.200,	62.320,	268.200,	-151.000,
* 1020.000,	-66.170,	9000.000,	-297.800,	9000.000,	
* -256.300,	312.500,	-338.500,	9000.000,	9000.000,	
* 9000.000,	9000.000,	-313.500,	9000.000,	0.000,	
* 44.420,	-183.400,	76.750,	212.700,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	-90.870,	9000.000,	9000.000,	9000.000/	
DATA A21/	91.460,	-24.360,	4.680,	122.900,	562.200,
* 529.000,	698.200,	9000.000,	286.300,	-47.510,	
* 9000.000,	9000.000,	225.400,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	326.400,	
* 0.000,	108.300,	249.200,	62.420,	464.400,	
* 9000.000,	9000.000,	450.100,	59.020,	9000.000,	
* 9000.000,	9000.000,	-125.900,	9000.000,	9000.000,	
* 9000.000,	-58.770,	9000.000,	9000.000,	9000.000/	
DATA A22/	34.010,	-52.710,	121.300,	9000.000,	747.700,
* 669.900,	708.700,	9000.000,	423.200,	9000.000,	
* -132.900,	9000.000,	-197.700,	9000.000,	9000.000,	
* -141.400,	9000.000,	587.300,	9000.000,	1821.000,	
* -84.530,	0.000,	0.000,	56.330,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	177.600,	9000.000,	9000.000,	215.000,	

* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
DATA A23/	36.700,	-185.100,	288.500,	33.610,	742.100,
* 649.100,	826.760,	9000.000,	552.100,	9000.000,	
* 176.500,	488.900,	-20.930,	9000.000,	9000.000,	
* -293.700,	9000.000,	18.980,	74.040,	1346.000,	
* -157.100,	0.000,	0.000,	-30.100,	9000.000,	
* 9000.000,	9000.000,	116.600,	9000.000,	-64.380,	
* 9000.000,	86.400,	9000.000,	9000.000,	363.700,	
* 9000.000,	-79.540,	9000.000,	9000.000,	9000.000/	
DATA A24/	-78.450,	-293.700,	-4.700,	134.700,	856.300,
* 860.100,	1201.000,	10000.000,	372.000,	9000.000,	
* 129.500,	9000.000,	113.900,	261.100,	91.130,	
* -126.000,	1301.000,	309.200,	492.000,	689.000,	
* 11.800,	17.970,	51.900,	0.000,	475.800,	
* 490.900,	534.700,	132.200,	9000.000,	546.700,	
* 9000.000,	247.800,	41.940,	9000.000,	337.700,	
* 9000.000,	-86.850,	215.200,	498.600,	9000.000/	
DATA A25/	-141.300,	-203.200,	-237.700,	375.500,	246.900,
* 661.600,	920.400,	9000.000,	128.100,	9000.000,	
* -246.300,	9000.000,	9000.000,	203.500,	-108.400,	
* 1088.000,	323.300,	9000.000,	356.900,	9000.000,	
* -314.900,	9000.000,	9000.000,	-255.400,	0.000,	
* -154.500,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	-60.700,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
DATA A26/	-32.690,	-49.920,	10.380,	-97.050,	341.700,
* 252.600,	417.900,	9000.000,	-142.600,	9000.000,	
* 9000.000,	9000.000,	-94.490,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	-34.680,	794.400,	
* 0.000,	533.200,	9000.000,	9000.000,	9000.000,	
* 139.800,	304.300,	10.170,	-27.700,	9000.000,	
* 9000.000,	48.400,	9000.000,	9000.000,	9000.000/	
DATA A27/	5541.000,	9000.000,	1824.000,	-127.800,	561.600,
* 9000.000,	360.700,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	5250.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	514.600,	9000.000,	
* -85.120,	0.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
DATA A28/	-52.650,	16.620,	21.500,	40.680,	823.500,
* 914.200,	1081.000,	9000.000,	303.700,	9000.000,	
* 243.800,	9000.000,	112.400,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	335.700,	9000.000,	
* -73.090,	9000.000,	-26.060,	-60.710,	9000.000,	
* 9000.000,	9000.000,	0.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	-47.370,	9000.000,	9000.000,	9000.000/	
DATA A29/	-7.481,	9000.000,	28.410,	9000.000,	461.600,
* 382.800,	9000.000,	9000.000,	160.600,	9000.000,	
* 9000.000,	239.800,	63.710,	106.700,	9000.000,	
* 9000.000,	9000.000,	9000.000,	125.700,	9000.000,	

*	-27.940,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	0.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	31.660,	
*	9000.000,	9000.000,	9000.000,	78.920,	9000.000/	
	DATA A30/	-25.310,	9000.000,	157.300,	404.300,	521.600,
*	9000.000,	23.480,	9000.000,	317.500,	9000.000,	
*	-146.300,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	48.480,	-133.160,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	0.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
	DATA A31/	140.000,	9000.000,	221.400,	150.600,	267.600,
*	9000.000,	0.000,	838.400,	9000.000,	9000.000,	
*	152.000,	9000.000,	9.207,	9000.000,	9000.000,	
*	9000.000,	164.400,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	481.300,	9000.000,	9000.000,	9000.000,	9000.000,	
*	0.000,	9000.000,	9000.000,	9000.000,	-417.200,	
*	9000.000,	9000.000,	9000.000,	302.200,	9000.000/	
	DATA A32/	128.000,	9000.000,	58.680,	9000.000,	501.300,
*	9000.000,	9000.000,	9000.000,	138.000,	9000.000,	
*	21.920,	9000.000,	476.600,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	-40.820,	21.760,	48.490,	9000.000,	
*	64.280,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	0.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
	DATA A33/	-31.520,	9000.000,	155.600,	291.100,	721.900,
*	9000.000,	9000.000,	9000.000,	-142.600,	9000.000,	
*	9000.000,	9000.000,	736.400,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	1169.000,	9000.000,	9000.000,	225.800,	224.000,	
*	125.300,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	0.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000/	
	DATA A34/	-72.880,	-184.400,	9000.000,	9000.000,	9000.000,
*	9000.000,	9000.000,	9000.000,	443.600,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	329.100,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	174.400,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	0.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	-119.800,	9000.000/	
	DATA A35/	50.490,	9000.000,	-2.504,	-143.200,	-25.870,
*	695.000,	-240.000,	9000.000,	110.400,	9000.000,	
*	41.570,	9000.000,	-122.100,	9000.000,	9000.000,	
*	9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
*	9000.000,	-215.000,	-343.600,	-58.430,	9000.000,	
*	9000.000,	9000.000,	9000.000,	85.700,	9000.000,	
*	535.800,	9000.000,	9000.000,	9000.000,	0.000,	
*	9000.000,	9000.000,	9000.000,	-97.710,	9000.000/	
	DATA A36/	-165.900,	9000.000,	9000.000,	9000.000,	9000.000,

* 9000.000,	386.600,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	-42.310,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 0.000,	9000.000,	9000.000,	9000.000,	9000.000/	
DATA A37/	41.900,	-3.167,	-75.670,	9000.000,	640.900,
* 726.700,	9000.000,	9000.000,	-8.671,	9000.000,	
* -34.140,	9000.000,	-209.300,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	298.400,	2344.000,	
* 201.700,	9000.000,	85.320,	143.200,	9000.000,	
* 313.800,	9000.000,	167.900,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	0.000,	9000.000,	9000.000,	9000.000/	
DATA A38/	-5.132,	9000.000,	-237.200,	-157.300,	649.700,
* 645.900,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	-124.600,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	0.000,	9000.000,	9000.000/	
DATA A39/	-31.950,	37.700,	-133.900,	-240.200,	64.160,
* 172.200,	-287.100,	9000.000,	97.040,	9000.000,	
* 9000.000,	9000.000,	-158.200,	9000.000,	9000.000,	
* 9000.000,	335.600,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	-186.700,	9000.000,	
* 9000.000,	9000.000,	9000.000,	-71.000,	9000.000,	
* -191.700,	9000.000,	9000.000,	6.699,	136.600,	
* 9000.000,	9000.000,	9000.000,	0.000,	9000.000/	
DATA A40/	147.300,	9000.000,	9000.000,	9000.000,	9000.000,
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	0.000/	

```

DO 5 I=1,40
AI(I,1)=A1(I)
AI(I,2)=A2(I)
AI(I,3)=A3(I)
AI(I,4)=A4(I)
AI(I,5)=A5(I)
AI(I,6)=A6(I)
AI(I,7)=A7(I)
AI(I,8)=A8(I)
AI(I,9)=A9(I)
AI(I,10)=A10(I)
AI(I,11)=A11(I)
AI(I,12)=A12(I)
AI(I,13)=A13(I)

```

```

AI(I, 14)=A14(I)
AI(I, 15)=A15(I)
AI(I, 16)=A16(I)
AI(I, 17)=A17(I)
AI(I, 18)=A18(I)
AI(I, 19)=A19(I)
AI(I, 20)=A20(I)
AI(I, 21)=A21(I)
AI(I, 22)=A22(I)
AI(I, 23)=A23(I)
AI(I, 24)=A24(I)
AI(I, 25)=A25(I)
AI(I, 26)=A26(I)
AI(I, 27)=A27(I)
AI(I, 28)=A28(I)
AI(I, 29)=A29(I)
AI(I, 30)=A30(I)
AI(I, 31)=A31(I)
AI(I, 32)=A32(I)
AI(I, 33)=A33(I)
AI(I, 34)=A34(I)
AI(I, 35)=A35(I)
AI(I, 36)=A36(I)
AI(I, 37)=A37(I)
AI(I, 38)=A38(I)
AI(I, 39)=A39(I)
AI(I, 40)=A40(I)

```

```

5 CONTINUE
DO 10 I=1, 76
RI(I)=RR(I)
QI(I)=QQ(I)
MAIN(I)=MAINSG(I)
10 CONTINUE
RETURN
END

```

```

C
C
C .....
C *
C *
C *          ULLE CONTAINS BUILT-IN UNIFAC LLE-PARAMETERS
C *
C *
C .....
C *
C .....
C SUBROUTINE ULLE(RI, QI, AI, MAIN)
C .....
C
C THE MAIN GROUPS ARE:
C   1 CH2 ..... 2 C=C ..... 3 ACH ..... 4 ACCH2 ..... 5 OH .....
C   6 P1 ..... 7 P2 ..... 8 H2O ..... 9 ACOH ..... 10 CH2CO .....
C   11 CHO ..... 12 FURFURAL 13 COOH ..... 14 COOC ..... 15 CH2O .....

```

```

C   16 CCL ..... 17 CCL2 .... 18 CCL3 .... 19 CCL4 .... 20 ACCL .....
C   21 CCN ..... 22 ACNH2 ... 23 CNO2 .... 24 ACNO2 ... 25 DOH .....
C   26 DEOH ..... 27 PYRIDINE 28 TCE ..... 29 MFA ..... 30 DMFA .....
C   31 TMS ..... 32 DMSO ....

```

THE SUB GROUPS ARE:

```

C   1 CH3 ..... 2 CH2 ..... 3 CH ..... 4 C ..... 5 CH2=CH ...
C   6 CH=CH ..... 7 CH=C ..... 8 CH2=C .... 9 ACH ..... 10 AC .....
C   11 ACCH3 .... 12 ACCH2 ... 13 ACCH .... 14 OH ..... 15 P1 .....
C   16 P2 ..... 17 H2O ..... 18 ACOH .... 19 CH3CO ... 20 CH2CO ....
C   21 CHO ..... 22 FURFURAL 23 COOH .... 24 HCOOH ... 25 CH3COO ...
C   26 CH2COO ... 27 CH3O .... 28 CH2O .... 29 CH-O .... 30 FCH2O ....
C   31 CH2CL .... 32 CHCL .... 33 CCL ..... 34 CH2CL2 .. 35 CHCL2 ....
C   36 CCL2 ..... 37 CHCL3 ... 38 CCL3 .... 39 CCL4 .... 40 ACCL .....
C   41 CH3CN .... 42 CH2CN ... 43 ACNH2 ... 44 CH3NO2 .. 45 CH2NO2 ...
C   46 CHNO2 .... 47 ACNO2 ... 48 (CH2OH)2 49 (HOMM)20 50 C5H5N ....
C   51 C5H4N .... 52 C5H3N ... 53 CCL2=CHCL 54 HCONHCH3 55 HCON(CH3)2
C   56 (CH2)4SO2 57 (CH2)2SO

```

IMPLICIT REAL*8(A-H,O-Z)

DIMENSION AI(40,40),RR(57),QQ(57),RI(76),QI(76),MAINSG(57),
 * MAIN(76)

DIMENSION A1(32),A2(32),A3(32),A4(32),A5(32),A6(32),A7(32),A8(32),
 * A9(32),A10(32),A11(32),A12(32),A13(32),A14(32),A15(32),A16(32),
 * A17(32),A18(32),A19(32),A20(32),A21(32),A22(32),A23(32),A24(32),
 * A25(32),A26(32),A27(32),A28(32),A29(32),A30(32),A31(32),A32(32)

DATA MAINSG/ 4*1, 4*2, 2*3, 3*4, 1*5, 1*6, 1*7, 1*8, 1*9, 2*10,
 * 1*11, 1*12, 2*13, 2*14, 4*15, 3*16, 3*17, 2*18, 1*19, 1*20,
 * 2*21, 1*22, 3*23, 1*24, 1*25, 1*26, 3*27, 1*28, 1*29, 1*30,
 * 1*31, 1*32/

DATA RR/0.9011, 0.6744, 0.4469, 0.2195, 1.3454, 1.1167, 0.8886, 1.1173,
 * 0.5313, 0.3652, 1.2663, 1.0396, 0.8121, 1.0000, 3.2499, 3.2491, 0.9200,
 * 0.8952, 1.6724, 1.4457, 0.9980, 3.1680, 1.3013, 1.5280, 1.9031, 1.6764,
 * 1.1450, 0.9183, 0.6908, 0.9183, 1.4654, 1.2380, 1.0060, 2.2564, 2.0606,
 * 1.8016, 2.8700, 2.6401, 3.3900, 1.1562, 1.8701, 1.6434, 1.0600, 2.0086,
 * 1.7818, 1.5544, 1.4199, 2.4088, 4.0013, 2.9993, 2.8332, 2.6670, 3.3092,
 * 2.4317, 3.0856, 4.0358, 2.8266/

DATA QQ/0.848, 0.540, 0.228, 0.000, 1.176, 0.867, 0.676, 0.988, 0.400,
 * 0.120, 0.968, 0.660, 0.348, 1.200, 3.128, 3.124, 1.400, 0.680, 1.488,
 * 1.180, 0.948, 2.484, 1.224, 1.532, 1.728, 1.420, 1.088, 0.780, 0.468,
 * 1.100, 1.264, 0.952, 0.724, 1.988, 1.684, 1.448, 2.410, 2.184, 2.910,
 * 0.844, 1.724, 1.416, 0.816, 1.868, 1.560, 1.248, 1.104, 2.248, 3.568,
 * 2.113, 1.833, 1.553, 2.860, 2.192, 2.736, 3.200, 2.472/

DATA A1/ 0.000, 74.540, -114.800, -115.700, 644.600,
 * 329.600, 310.700, 1300.000, 2255.000, 472.600,
 * 158.100, 383.000, 139.400, 972.400, 662.100,
 * 42.140, -243.900, 7.500, -5.550, 924.800,
 * 696.800, 902.200, 556.700, 575.700, 527.500,
 * 269.200, -300.000, -63.600, 928.300, 331.000,
 * 561.400, 956.500/

DATA A2/	292.300,	0.000,	340.700,	4102.000,	724.400,
* 1731.000,	1731.000,	896.000,	9000.000,	343.700,	
* -214.700,	9000.000,	1647.000,	-577.500,	289.300,	
* 99.610,	337.100,	4583.000,	5831.000,	9000.000,	
* 405.900,	9000.000,	425.700,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	500.700,	115.400,	
* 784.400,	265.400/				
DATA A3/	156.500,	-94.780,	0.000,	167.000,	703.900,
* 511.500,	577.300,	859.400,	1649.000,	593.700,	
* 362.300,	31.140,	461.800,	6.000,	32.140,	
* -18.810,	9000.000,	-231.900,	3000.000,	-878.100,	
* 29.130,	1.640,	-1.770,	-11.190,	358.900,	
* 363.500,	-578.200,	9000.000,	364.200,	-58.100,	
* 21.970,	84.160/				
DATA A4/	104.400,	-269.700,	-146.800,	0.000,	4000.000,
* 136.600,	906.800,	5695.000,	292.600,	916.700,	
* 1218.000,	715.600,	339.100,	5688.000,	213.100,	
* -114.100,	9000.000,	-12.140,	-141.300,	-107.300,	
* 1208.000,	689.600,	3629.000,	-175.600,	337.700,	
* 1023.000,	-390.700,	9000.000,	9000.000,	9000.000,	
* 238.000,	132.200/				
DATA A5/	328.200,	470.700,	-9.210,	1.270,	0.000,
* 937.300,	991.300,	28.730,	-195.500,	67.070,	
* 1409.000,	-140.300,	-104.000,	195.600,	262.500,	
* 62.050,	272.200,	-61.570,	-41.750,	-597.100,	
* -189.300,	-348.200,	-30.700,	-159.000,	536.600,	
* 53.370,	183.300,	-44.440,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A6/	-136.700,	-135.700,	-223.000,	-162.600,	-281.100,
* 0.000,	0.000,	-61.290,	-153.200,	-47.410,	
* -344.100,	299.300,	244.400,	19.570,	1970.000,	
* -166.400,	128.600,	1544.000,	224.600,	9000.000,	
* 9000.000,	9000.000,	150.800,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A7/	-131.900,	-135.700,	-252.000,	-273.600,	-268.800,
* 0.000,	0.000,	5.890,	-153.200,	353.800,	
* -338.600,	-241.800,	-57.980,	487.100,	1970.000,	
* -166.400,	507.800,	1544.000,	-207.000,	9000.000,	
* 9000.000,	9000.000,	150.800,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A8/	342.400,	220.600,	372.800,	203.700,	-122.400,
* 247.000,	104.900,	0.000,	344.500,	-171.800,	
* -349.900,	66.950,	-465.700,	-6.320,	64.420,	
* 315.900,	370.700,	356.800,	502.900,	-97.270,	
* 198.300,	-109.800,	1539.000,	32.920,	-269.200,	
* 9000.000,	-873.600,	1429.000,	-364.200,	-117.400,	
* 18.410,	9000.000/				
DATA A9/	-159.800,	9000.000,	-473.200,	-470.400,	-63.150,
* -547.200,	-547.200,	-595.900,	0.000,	-825.700,	
* 9000.000,	9000.000,	9000.000,	-898.300,	9000.000,	
* 9000.000,	9000.000,	9000.000,	4894.000,	9000.000,	

* 9000.000,	-851.600,	9000.000,	-16.130,	-538.600,	
* 9000.000,	-637.300,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A10/	66.560,	306.100,	-78.310,	-73.870,	216.000,
* 401.700,	-127.600,	634.800,	-568.000,	0.000,	
* -37.360,	120.300,	1247.000,	258.700,	5.202,	
* 1000.000,	-301.000,	12.010,	-10.880,	902.600,	
* 430.600,	1010.000,	400.000,	-328.600,	211.600,	
* 9000.000,	9000.000,	148.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A11/	146.100,	517.000,	-75.300,	223.200,	-431.300,
* 643.400,	231.400,	623.700,	9000.000,	128.000,	
* 0.000,	1724.000,	0.750,	-245.800,	9000.000,	
* 751.800,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A12/	14.780,	9000.000,	-10.440,	-184.900,	444.700,
* -94.640,	732.300,	211.600,	9000.000,	48.930,	
* -311.600,	0.000,	1919.000,	57.700,	9000.000,	
* 9000.000,	-347.900,	-249.300,	61.590,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	-278.200,	
* 9000.000,	-208.400,	-13.910,	9000.000,	173.800,	
* 9000.000,	9000.000/				
DATA A13/	1744.000,	-48.520,	75.490,	147.300,	118.400,
* 728.700,	349.100,	652.300,	9000.000,	-101.300,	
* 1051.000,	-115.700,	0.000,	-117.600,	-96.620,	
* 19.770,	1670.000,	48.150,	43.830,	874.300,	
* 9000.000,	942.200,	446.300,	9000.000,	572.700,	
* 9000.000,	9000.000,	-2.160,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A14/	-320.100,	485.600,	114.800,	-170.000,	180.600,
* -76.640,	-152.800,	385.900,	-337.300,	58.840,	
* 1090.000,	-46.130,	1417.000,	0.000,	-235.700,	
* 9000.000,	108.900,	-209.700,	54.570,	629.000,	
* -149.200,	9000.000,	9000.000,	9000.000,	343.100,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A15/	1571.000,	76.440,	52.130,	65.690,	137.100,
* -218.100,	-218.100,	212.800,	9000.000,	52.380,	
* 9000.000,	9000.000,	1402.000,	461.300,	0.000,	
* 301.100,	137.800,	-154.300,	47.670,	9000.000,	
* 9000.000,	9000.000,	95.180,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A16/	73.800,	-24.360,	4.680,	122.900,	455.100,
* 351.500,	351.500,	770.000,	9000.000,	483.900,	
* -47.510,	9000.000,	337.100,	9000.000,	225.400,	
* 0.000,	110.500,	249.200,	62.420,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A17/	27.900,	-52.710,	9000.000,	9000.000,	669.200,

* -186.100,	-401.600,	740.400,	9000.000,	550.600,	
* 9000.000,	808.800,	437.700,	-132.900,	-197.700,	
* -21.350,	0.000,	9000.000,	56.330,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A18/	21.230,	-185.100,	288.500,	33.610,	418.400,
* -465.700,	-465.700,	793.200,	9000.000,	342.200,	
* 9000.000,	203.100,	370.400,	176.500,	-20.930,	
* -157.100,	9000.000,	0.000,	-30.100,	9000.000,	
* 70.040,	-75.500,	9000.000,	9000.000,	9000.000,	
* 9000.000,	18.980,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A19/	89.970,	-293.700,	-4.700,	134.700,	713.500,
* -260.300,	512.200,	1205.000,	1616.000,	550.000,	
* 9000.000,	70.140,	438.100,	129.500,	113.900,	
* 11.800,	17.970,	51.900,	0.000,	475.800,	
* 492.000,	1302.000,	490.900,	534.700,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A20/	-59.060,	9000.000,	777.800,	-47.130,	1989.000,
* 9000.000,	9000.000,	390.700,	9000.000,	190.500,	
* 9000.000,	9000.000,	1349.000,	-246.300,	9000.000,	
* 9000.000,	9000.000,	9000.000,	-255.400,	0.000,	
* 346.200,	9000.000,	-154.500,	9000.000,	124.800,	
* 9000.000,	-387.700,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A21/	29.080,	34.780,	56.410,	-53.290,	2011.000,
* 9000.000,	9000.000,	63.480,	9000.000,	-349.200,	
* 9000.000,	9000.000,	9000.000,	2.410,	9000.000,	
* 9000.000,	9000.000,	-15.620,	-54.860,	-465.200,	
* 0.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	134.300,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A22/	175.800,	9000.000,	-218.900,	-15.410,	529.000,
* 9000.000,	9000.000,	-239.800,	-860.300,	857.700,	
* 9000.000,	9000.000,	681.400,	9000.000,	9000.000,	
* 9000.000,	9000.000,	-216.300,	8455.000,	9000.000,	
* 9000.000,	0.000,	9000.000,	179.900,	125.300,	
* 9000.000,	924.500,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A23/	94.340,	375.400,	113.600,	-97.050,	483.800,
* 264.700,	264.700,	13.320,	9000.000,	377.000,	
* 9000.000,	9000.000,	152.400,	9000.000,	-94.490,	
* 9000.000,	9000.000,	9000.000,	-34.680,	794.400,	
* 9000.000,	9000.000,	0.000,	9000.000,	139.800,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A24/	193.600,	9000.000,	7.180,	-127.100,	332.600,
* 9000.000,	9000.000,	439.900,	-230.400,	211.600,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	514.600,	9000.000,	
* 9000.000,	175.800,	9000.000,	0.000,	963.000,	

* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A25/	108.500,	9000.000,	247.300,	453.400,	-289.300,
* 9000.000,	9000.000,	-424.300,	523.000,	82.770,	
* 9000.000,	-75.230,	-1707.000,	29.860,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	-241.700,	
* 9000.000,	164.400,	481.300,	-246.000,	0.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A26/	81.490,	9000.000,	-50.710,	-30.280,	-99.560,
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 0.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A27/	-128.800,	9000.000,	-225.300,	-124.600,	-319.200,
* 9000.000,	9000.000,	203.000,	-222.700,	9000.000,	
* 9000.000,	-201.900,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	-114.700,	9000.000,	-906.500,	
* -169.700,	-944.900,	9000.000,	9000.000,	9000.000,	
* 9000.000,	0.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A28/	147.300,	9000.000,	9000.000,	9000.000,	837.900,
* 9000.000,	9000.000,	1153.000,	9000.000,	417.400,	
* 9000.000,	123.200,	639.700,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	0.000,	9000.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A29/	-11.910,	176.700,	-80.480,	9000.000,	9000.000,
* 9000.000,	9000.000,	-311.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	0.000,	9000.000,	
* 9000.000,	9000.000/				
DATA A30/	14.910,	132.100,	-17.780,	9000.000,	9000.000,
* 9000.000,	9000.000,	-262.600,	9000.000,	9000.000,	
* 9000.000,	-281.900,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	0.000,	
* 9000.000,	9000.000/				
DATA A31/	67.840,	42.730,	59.160,	26.590,	9000.000,
* 9000.000,	9000.000,	1.110,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	
* 0.000,	9000.000/				
DATA A32/	36.420,	60.820,	29.770,	55.970,	9000.000,
* 9000.000,	9000.000,	9000.000,	9000.000,	9000.000,	


```

* 9000.000, 9000.000, 9000.000, 9000.000, 9000.000,
* 9000.000, 9000.000, 9000.000, 9000.000, 9000.000,
* 9000.000, 9000.000, 9000.000, 9000.000, 9000.000,
* 9000.000, 9000.000, 9000.000, 9000.000, 9000.000,
* 9000.000, 0.000/
DO 5 I=1,32
AI(I,1)=A1(I)
AI(I,2)=A2(I)
AI(I,3)=A3(I)
AI(I,4)=A4(I)
AI(I,5)=A5(I)
AI(I,6)=A6(I)
AI(I,7)=A7(I)
AI(I,8)=A8(I)
AI(I,9)=A9(I)
AI(I,10)=A10(I)
AI(I,11)=A11(I)
AI(I,12)=A12(I)
AI(I,13)=A13(I)
AI(I,14)=A14(I)
AI(I,15)=A15(I)
AI(I,16)=A16(I)
AI(I,17)=A17(I)
AI(I,18)=A18(I)
AI(I,19)=A19(I)
AI(I,20)=A20(I)
AI(I,21)=A21(I)
AI(I,22)=A22(I)
AI(I,23)=A23(I)
AI(I,24)=A24(I)
AI(I,25)=A25(I)
AI(I,26)=A26(I)
AI(I,27)=A27(I)
AI(I,28)=A28(I)
AI(I,29)=A29(I)
AI(I,30)=A30(I)
AI(I,31)=A31(I)
AI(I,32)=A32(I)
5 CONTINUE
DO 10 I=1,57
RI(I)=RR(I)
QI(I)=QQ(I)
MAIN(I)=MAINS(I)
10 CONTINUE
RETURN
END

```

```

C
C .....
C *
C *      <<< MAIN PROGRAM TO CALL UNIFAC SUBROUTINES >>>
C *
C *
C *      RTI INTERACTIVE VERSION: 10-8-86 (WRITTEN BY TONY ROGERS)
C *
C .....
C
C      IMPLICIT REAL*8(A-H,O-Z)
C      CHARACTER*25 NCOMP
C      DIMENSION X(10),ACT(10),DACT(10,10),TACT(10),DLACT(10),PVAP(10),
C      *          AI(40,40),RI(76),OI(76),MAIN(76),NCOMP(10),HENRY(10)
C      DATA MDL /3/
C      DATA LIQ /0/
C      DATA NVAP /1/
C      DATA NC /2/
C      DATA IPASS /0/
C      DATA IFLAG /0/
C      DATA NH /0/
C
C      --- SET UP I/O FILE NAMES ---
C
C      OPEN(6,FILE='UNIFAC.DAT',STATUS='OLD',ACCESS='SEQUENTIAL',
C      *FORM='FORMATTED')
C      OPEN(7,FILE='UNIFAC.OUT',STATUS='NEW',ACCESS='SEQUENTIAL',
C      *FORM='FORMATTED')
C      OPEN(8,FILE='IOWN.DAT',STATUS='OLD',ACCESS='SEQUENTIAL',
C      *FORM='FORMATTED')
C
C      1 READ(6,*) MODEL,IOUT,NDIF,NACT,LIQ
C      IF(IPASS.GT.0) GOTO 2
C
C      IF(IOUT.GT.0) OPEN(IOUT,FILE='BACKUP.OUT',STATUS='NEW',
C      *ACCESS='SEQUENTIAL',FORM='FORMATTED')
C
C      --- PRINT "START-UP" MESSAGES ---
C
C      2 IF(LIQ.GT.0) GOTO 5
C      WRITE(7,901)
C      WRITE(*,901)
C      GOTO 10
C      5 WRITE(7,902)
C      WRITE(*,902)
C
C      --- INTERACTIVE READ STATEMENTS ---
C
C      10 WRITE(*,'(1X,A\)' ) 'Enter database ID: [1] for VLE, [2] for LLE,
C      &r [3] for "Environmental" ==> '
C      READ(*,'(BN,I3)') MDL
C      IF((MDL.NE.1).AND.(MDL.NE.2).AND.(MDL.NE.3)) GOTO 10

```

```

WRITE(*, '(/)')
C
IF(LIQ.GT.0) GOTO 100
C
20 WRITE(*, '(1X,A\)' ) 'Enter total number of components ==> '
READ(*, '(BN,I3)') NC
IF((NC.LE.0).OR.(NC.GT.10)) GOTO 20
WRITE(*, '(/)')
IF(NC.GE.2) GOTO 30
C
C
C      --- ERROR MESSAGE FOR NUMBER OF COMPONENTS ---
C
WRITE(*, '(5X,A)') '* Number of components must be 2 or greater! *'
WRITE(*, '(//)')
GOTO 130
C
30 WRITE(*, '(1X,A\)' ) 'Enter absolute temperature in [K] ==> '
READ(*,*) T
IF(T.LE.0.DO) GOTO 30
WRITE(*, '(/)')
C
DO 35 L=1,NC
WRITE(*, '(5X,A,I2,A\)' ) 'Name (<25 chars.) for Component',L,
&' ==> '
READ(*, '(BN,A)') NCOMP(L)
WRITE(*, '(5X,A,I2,A\)' ) 'Liquid mole fraction for Component',L,
&' ==> '
READ(*,*) X(L)
WRITE(*, '(/)')
35 CONTINUE
WRITE(*, '(/)')
C
40 WRITE(*, '(1X,A\)' ) 'Calculate K-factors (y/x) for components? [1=Y
&, 0=N] ==> '
READ(*, '(BN,I3)') NH
IF((NH.NE.0).AND.(NH.NE.1)) GOTO 40
WRITE(*, '(/)')
IF(NH.EQ.0) GOTO 100
C
50 WRITE(*, '(1X,A\)' ) 'Enter [1] for vapor pressure entry, [2] for Ar
&toine coefficients ==> '
READ(*, '(BN,I3)') NVAP
IF((NVAP.NE.1).AND.(NVAP.NE.2)) GOTO 50
WRITE(*, '(/)')
IF(NVAP.EQ.2) GOTO 60
C
DO 55 K=1,NC
WRITE(*, '(5X,A,I2,A\)' ) 'Enter Component',K,' v.p. in mm Hg ==> '
READ(*,*) PVAP(K)
55 CONTINUE
C
WRITE(*, '(/)')
GOTO 100

```

```

C
60 DO 70 K=1, NC
   WRITE(*, '(5X, A, I2, A\)' ) ' Antoine Coefficient A (v.p. in mm Hg) for
   & Comp. ', K, ' ==> '
   READ(*, *) ANTA
   WRITE(*, '(5X, A, I2, A\)' ) ' Antoine Coefficient B (v.p. in mm Hg) for
   & Comp. ', K, ' ==> '
   READ(*, *) ANTB
   WRITE(*, '(5X, A, I2, A\)' ) ' Antoine Coefficient C (v.p. in mm Hg) for
   & Comp. ', K, ' ==> '
   READ(*, *) ANTC
   PVAP(K)=DEXP(ANTA-ANTB/(T+ANTC))/760.DO
   WRITE(*, '(/)')
70 CONTINUE

C
100 IF(MDL.EQ.1) CALL UVLE(RI, QI, AI, MAIN)
    IF(MDL.EQ.2) CALL ULLE(RI, QI, AI, MAIN)
    IF(MDL.EQ.3) CALL ENVIRON(RI, QI, AI, MAIN)
    IF(LIQ.GT.0) GOTO 120
    CALL FINOUT(NC, RI, QI, AI, MAIN, NG)
    REWIND 8

C
    WRITE(*, '(//)')
    PAUSE
    WRITE(*, '(//)')

C
    CALL PARAM(NC, NG, T)
    CALL UNIFA(NDIF, NACT, NC, NG, T, X, ACT, DACT, TACT)

C
C
C    --- CALCULATE K-VALUES [Y/X] FOR ALL COMPONENTS (NH=1) ---
C
    WRITE(7, 903) T
    WRITE(*, 903) T
    DO 110 K=1, NC
      HENRY(K)=-999.DO
      IF(NH.GT.0) HENRY(K)=ACT(K)*PVAP(K)
      WRITE(7, 904) K, NCOMP(K), X(K), ACT(K), HENRY(K)
      WRITE(*, 904) K, NCOMP(K), X(K), ACT(K), HENRY(K)
      WRITE(7, 905)
      WRITE(*, 905)
110 CONTINUE

C
    GOTO 130

C
C
C    --- CALCULATION OF AQUEOUS SOLUBILITIES (LIQ=1) ---
C
120 CALL XLIQUID(NC, MDL, MODEL, IOUT, NDIF, NACT, RI, QI, AI, MAIN)
C
130 CONTINUE

C
C
C    --- USER PROMPT TO RE-START PROGRAM OR END ---
C
    WRITE(*, '(//1X, A\)' ) 'Enter [1] to re-start program or [0] to end

```

```

&=> '
  READ(*, '(BN, I3)') IFLAG
  IF((IFLAG.NE.0).AND.(IFLAG.NE.1)) GOTO 130
  WRITE(*, '(/////)' )
  IF(IFLAG.EQ.0) GOTO 999
  REWIND 6
  IFLAG=0
  IPASS=1
  GOTO 1
999 CONTINUE
C
  STOP
C
  --- FORMAT STATEMENTS ---
C
901 FORMAT(20X, '*** ACTIVITY COEFFICIENT CALCULATION ***', ///,
  &10X, 'NOTE: Vapor pressure data (or Antoine coefficients) must be s
  &supplied', /, 16X, 'for all components to calculate their Henry's Con
  &stants', ///)
902 FORMAT(15X, '*** BINARY LIQUID-LIQUID EQUILIBRIUM ROUTINE ***', ///,
  &10X, 'NOTE: Enter the most dilute chemical as Component #1.', ///)
903 FORMAT(25X, 'TEMP. =', F8.2, ' [K]', //, 5X, 'ID #', 5X, 'COMPONENT NAME',
  &4X, 'LIG. M-FRAC', 5X, 'ACT. COEFF.', 5X, 'H (TU)', //)
904 FORMAT(6X, I2, 6X, A15, 2X, E13.5, 3X, F12.5, 3X, F8.1)
905 FORMAT(///)
  END
C
C
C *****
C *
C *
C * <<< SUBROUTINE TO HANDLE INPUT/OUTPUT OF DATA >>>
C *
C *
C *****
C SUBROUTINE FINOUT(NC, RI, QI, AI, MAIN, NG)
C *****
C IMPLICIT REAL*8(A-H, O-Z)
C COMMON/UNIF/RT(10, 10), QT(10, 10), TAU(10, 10), S(10, 10), F(10), Q(10),
C * R(10), P(10, 10)
C DIMENSION AI(40, 40), RI(76), QI(76), MAIN(76),
C * NGM(10), MS(10, 10, 2), NY(10, 20), JH(76), IH(20)
C
C IF(IOUT.EQ.0) IOUT=7
C NK=NC
C
C DO 10 I=1, 10
C DO 10 J=1, NK
C QT(I, J)=0. DO
C RT(I, J)=0. DO
10 CONTINUE
C
C IF(MODEL.NE.1) GO TO 30

```

```

      NG=NK
C
      DO 20 I=1, NK
        READ(8, *) PT(I, I), QT(I, I), (P(I, J), J=1, NK)
20 CONTINUE
30 CONTINUE
C
      IF(MODEL.EQ.1) GO TO 290
      READ(8, *) IOWNRQ, IOWNP
      IF(IOWNRQ.EQ.0) GO TO 50
C
      DO 40 I=1, IOWNRQ
        READ(8, *) K, RI(K), QI(K)
40 CONTINUE
C
      50 IF(IOWNP.EQ.0) GO TO 70
C
      DO 60 I=1, IOWNP
        READ(8, *) J, K, AI(J, K)
60 CONTINUE
70 CONTINUE
C
      DO 80 I=1, NK
        DO 90 J=1, 10
          DO 80 K=1, 2
            MS(I, J, K)=0
80 CONTINUE
C
      DO 90 I=1, 76
        JH(I)=0
90 CONTINUE
C
      NMAX=0
      DO 100 I=1, NK
        WRITE(*, '(1X, A, I2, A\)' ) 'Enter number of types of subGroups in
&component', I, ' ==> '
        READ(*, '(BN, I3)' ) NGRP
        WRITE(*, '(/)')
C
        DO 95 KR=1, NGRP
          WRITE(*, '(5X, A, I2, A\)' ) 'Enter ID# of subGroup', KR, ' ==> '
          READ(*, '(BN, I3)' ) MS(I, KR, 1)
          WRITE(*, '(5X, A, I3, A, I2, A\)' ) 'Enter number of subGroup', KR,
& ' in component', I, ' ==> '
          READ(*, '(BN, I3)' ) MS(I, KR, 2)
          WRITE(*, '(/)')
95 CONTINUE
        WRITE(*, '(/)')
C
        IF(NGRP.GT.NMAX) NMAX=NGRP
100 CONTINUE
C
      IC=1

```

```

C      DO 160 I=1, NK
C
C      DO 150 J=1, NMAX
        IF (MS(I, J, 1).EQ.0) GO TO 160
        IH(IC)=MS(I, J, 1)
        IF (IC.EQ.1) GO TO 140
        IF (IH(IC).EQ.IH(IC-1)) GO TO 150
        IF (IH(IC).GT.IH(IC-1)) GO TO 140
        IF (IC.GT.2) GO TO 110
        IHH=IH(1)
        IH(1)=IH(2)
        IH(2)=IHH
        GO TO 140
110     I1=IC-1
C
C      DO 130 I2=1, I1
        IF (IH(IC).GT.IH(I2)) GO TO 130
        IF (IH(IC).EQ.IH(I2)) GO TO 150
        I4=IC-I2
C
C      DO 120 I3=1, I4
        IH(IC+1-I3)=IH(IC-I3)
120     CONTINUE
C
C      IH(I2)=MS(I, J, 1)
130     CONTINUE
C
140     IC=IC+1
        IF (IC.GT.20) WRITE(7, 905)
        IF (IC.GT.20) WRITE(*, 905)
        IF (IOUT.NE.7.AND.IC.GT.20) WRITE(IOUT, 905)
150     CONTINUE
C
160     CONTINUE
C
C      IC=IC-1
C
C      DO 170 I=1, IC
        JH(IH(I))=I
170     CONTINUE
C
C      DO 180 I=1, 10
        DO 180 J=1, 20
            NY(I, J)=0
180     CONTINUE
C
C      DO 200 I=1, NK
C
C      DO 190 J=1, 10
        IF (MS(I, J, 1).EQ.0) GO TO 200
        N1=MS(I, J, 1)
        N2=MS(I, J, 2)

```

```

                IF(N1.EQ.0) GO TO 200
                N3=JH(N1)
                NY(I,N3)=N2
190      CONTINUE
C
200      CONTINUE
C
        I=0
        NGMGL=0
C
        DO 210 K=1, IC
            NSG=IH(K)
            NGMNY=MAIN(NSG)
            IF(NGMNY.NE.NGMGL) I=I+1
            NGM(I)=NGMNY
            NGMGL=NGMNY
        DO 210 J=1, NK
            RT(I,J)=RT(I,J)+NY(J,K)*RI(NSG)
            QT(I,J)=QT(I,J)+NY(J,K)*QI(NSG)
210      CONTINUE
C
        NG=I
        WRITE(7,906) (IH(K),K=1,IC)
        WRITE(*,906) (IH(K),K=1,IC)
        WRITE(7,907) (MAIN(IH(K)),K=1,IC)
        WRITE(*,907) (MAIN(IH(K)),K=1,IC)
        WRITE(7,908)
        WRITE(*,908)
C
        DO 220 I=1, NK
            WRITE(*,909) I, (NY(I,K),K=1,IC)
            WRITE(7,909) I, (NY(I,K),K=1,IC)
220      CONTINUE
C
        WRITE(7,913)
        WRITE(*,913)
        IF(IOUT.EQ.7) GO TO 240
        WRITE(IOUT,906) (IH(K),K=1,IC)
        WRITE(IOUT,907) (MAIN(IH(K)),K=1,IC)
        WRITE(IOUT,908)
C
        DO 230 I=1, NK
            WRITE(IOUT,909) I, (NY(I,K),K=1,IC)
230      CONTINUE
C
        WRITE(IOUT,913)
240      CONTINUE
C
        DO 250 I=1, NG
        DO 250 J=1, NG
            NI=NGM(I)
            NJ=NGM(J)
            P(I,J)=AI(NJ,NI)

```



```

250 CONTINUE
C
WRITE(7,910)
WRITE(*,910)
C
DO 260 K=1,IC
  NN=IH(K)
  WRITE(*,911) NN,RI(NN),QI(NN)
  WRITE(7,911) NN,RI(NN),QI(NN)
260 CONTINUE
C
WRITE(7,913)
WRITE(*,913)
IF(IOUT.EQ.7) GO TO 280
WRITE(IOUT,910)
C
DO 270 K=1,IC
  NN=IH(K)
  WRITE(IOUT,911) NN,RI(NN),QI(NN)
270 CONTINUE
C
WRITE(IOUT,913)
C
280 CONTINUE
290 CONTINUE
C
WRITE(7,902)
WRITE(*,902)
C
DO 300 I=1,NG
  WRITE(*,901) (P(I,J),J=1,NG)
  WRITE(7,901) (P(I,J),J=1,NG)
300 CONTINUE
C
WRITE(7,913)
WRITE(*,913)
IF(MODEL.EQ.0) WRITE(7,903)
IF(MODEL.EQ.0) WRITE(*,903)
IF(MODEL.EQ.1) WRITE(7,912)
IF(MODEL.EQ.1) WRITE(*,912)
IF(IOUT.EQ.7) GO TO 320
WRITE(IOUT,902)
C
DO 310 I=1,NG
  WRITE(IOUT,901) (P(I,J),J=1,NG)
310 CONTINUE
C
WRITE(IOUT,913)
IF(MODEL.EQ.0) WRITE(IOUT,903)
IF(MODEL.EQ.1) WRITE(IOUT,912)
320 CONTINUE
C
DO 330 I=1,NK

```

```

      Q(I)=0.DO
      R(I)=0.DO
DO 330 K=1,NG
      Q(I)=Q(I)+QT(K,I)
      R(I)=R(I)+RT(K,I)
330 CONTINUE
C
      DO 340 I=1,NK
        WRITE(*,904) I,R(I),Q(I)
        WRITE(7,904) I,R(I),Q(I)
340 CONTINUE
C
      IF(IOUT.EQ.7) GO TO 360
C
      DO 350 I=1,NK
        WRITE(IOUT,904) I,R(I),Q(I)
350 CONTINUE
360 CONTINUE
C
C      --- FORMAT STATEMENTS ---
C
901 FORMAT(1X,10F12.3)
902 FORMAT(2X,'INTERACTION PARAMETERS',/)
903 FORMAT(1X,'UNIFAC MOLECULAR R AND Q',/)
904 FORMAT(15,2F15.4)
905 FORMAT(1X,'** WARNING: NO. OF SUBGROUPS MUST NOT EXCEED 20 **')
906 FORMAT(/,1X,'SUB GROUPS :',20I3)
907 FORMAT(1X,'MAIN GROUPS:',20I3)
908 FORMAT(1X,'COMPONENT')
909 FORMAT(6X,I2,5X,20I3)
910 FORMAT(1X,'GROUP R- AND Q-VALUES',/)
911 FORMAT(1X,I3,2F10.4)
912 FORMAT(1X,'SPECIFIED UNIFAC R AND Q',/)
913 FORMAT(//)
      RETURN
      END
C
C
C*****
C
C      <<< UNIFAC BINARY LIQUID-LIQUID FLASH ROUTINE >>>
C
C      NEWTON-RAPHSON ALGORITHM (GAUSS-JORDAN MAXIMUM PIVOT STRATEGY)
C-----
C      THIS IS A SUBROUTINE TO IMPLEMENT THE NEWTON-RAPHSON ALGORITHM FOR
C      SOLVING SYSTEMS OF NONLINEAR ALGEBRAIC EQUATIONS. A VARIATION OF
C      THE GAUSS-JORDAN MAXIMUM PIVOT STRATEGY IS EMPLOYED TO DETERMINE
C      THE INVERSE OF THE JACOBIAN MATRIX. THE CORRECTION FACTORS ARE
C      CALCULATED IN AN ITERATIVE MANNER TO BRING THE ADJUSTABLE VARIABLES
C      WITHIN A SPECIFIED TOLERANCE.
C
C      M = NUMBER OF COLUMNS IN MATRIX C

```

```

C           N = NUMBER OF ROWS IN MATRIX C
C
C
C*****
SUBROUTINE XLIQUID(NC, MDL, MODEL, IOU, NDIF, NACT, RI, QI, AI, MAIN)
C*****
IMPLICIT REAL*8(A-H, O-Z)
CHARACTER*25 NAME
DIMENSION C(10, 11), XNEW(10), DX(10), IR(10), F(10), XMW(10),
&          X1(10), X2(10), XOLD(10), FOLD(10), TEST(10), IE(10),
&          ACT1(10), DACT1(10, 10), TACT1(10), DLACT1(10),
&          ACT2(10), DACT2(10, 10), TACT2(10), DLACT2(10),
&          XE(10, 2), TEMP(10), FMIN(10, 2), NAME(10),
&          AI(40, 40), RI(76), QI(76), MAIN(76)
C
C      --- SET LENGTH AND WIDTH OF JACOBIAN MATRIX ---
C
      N=NC
      M=N+1
      INDEX=0
C
C      --- INITIALIZE ARRAYS, VECTORS ---
C
      DO 5 I=1, 10
          DO 6 J=1, N
              XE(I, J)=0. DO
              FMIN(I, J)=0. DO
6          CONTINUE
C
          TEMP(I)=0. DO
          NAME(I)=' '
5 CONTINUE
C
C      --- BEGIN OUTER LOOP TO READ MULTIPLE DATA SETS ---
C
      READ(6, *) IMAX, TOL
C
      WRITE(*, '(1X, A\)' ) 'Enter number of data sets (binary pairs) ==> '
      READ(*, '(BN, I3)' ) NDS
      WRITE(*, '( / )' )
C
      DO 777 KN=1, NDS
C
C      --- READ DEFAULT PARAMETERS ---
C
          WRITE(*, '(1X, A, I2, A\)' ) 'Give name (<25 chars.) of binary system',
&KN, ' ==> '
          READ(*, '(BN, A)' ) NAME(KN)
          WRITE(*, '( / )' )
C
          WRITE(*, '(1X, A\)' ) 'Enter absolute temperature in [K] ==> '
          READ(*, *) TEMP(KN)

```

```

      T=TEMP(KN)
      WRITE(*, '(/)')
C
      WRITE(7, 902) KN, NAME(KN), TEMP(KN)
      WRITE(*, 902) KN, NAME(KN), TEMP(KN)
      WRITE(7, 903)
      WRITE(*, 903)
C
      DO 7 LP=1, N
      WRITE(*, '(5X, A, I2, A\)' ) 'Enter molecular weight for component', LP,
&' ==> '
      READ(*, *) XMW(LP)
C
      WRITE(*, '(5X, A, I2, A\)' ) 'Enter composition guess for component',
&LP, ' ==> '
      READ(*, *) XNEW(LP)
      WRITE(*, '(/)')
7 CONTINUE
      WRITE(*, '(/)')
C
      CALL FINOUT(NC, RI, QI, AI, MAIN, NG)
      REWIND 8
C
C      --- INITIALIZE COEFFICIENT MATRICES (C, FOLD, XOLD, DX, IE) ---
C
      DO 10 L=1, N
      FOLD(L)=0. DO
      XOLD(L)=XNEW(L)
      DX(L)=0. DO
      IE(L)=0
C
      DO 20 LL=1, M
      C(L, LL)=0. DO
20 CONTINUE
C
10 CONTINUE
C
C      --- WRITE "WAIT" MESSAGE TO SCREEN ---
C
      WRITE(7, 904)
      WRITE(*, 904)
C
C
C      <<< START OF NEWTON-RAPHSON ITERATION LOOP >>>
C
      DO 999 ITER=1, IMAX
      INDEX=ITER-1
C
      WRITE(7, 905) INDEX
      WRITE(*, 905) INDEX
C
C      --- EVALUATE FUNCTION VECTOR (F) --

```

```

C
X1(1)=XNEW(1)
X1(2)=1.DO-X1(1)
X2(1)=XNEW(2)
X2(2)=1.DO-X2(1)
C
CALL PARAM(N, NG, T)
CALL UNIFA(NDIF, NACT, N, NG, T, X1, ACT1, DACT1, TACT1)
CALL UNIFA(NDIF, NACT, N, NG, T, X2, ACT2, DACT2, TACT2)
C
DO 21 J=1, N
    F(J)=X1(J)*ACT1(J)-X2(J)*ACT2(J)
    TEST(J)=DSQRT((FOLD(J))**2+(F(J))**2)
21 CONTINUE
C
C
C    --- TEST FOR CONVERGENCE OF SOLUTION ---
C
LOGIC=-1
DIFF=DSQRT((XNEW(1)-XOLD(1))**2+(XNEW(2)-XOLD(2))**2)
IF(DIFF.GT.TOL) LOGIC=1
C
DO 22 I=1, N
    IF(TEST(I).GT.TOL) LOGIC=1
    IF(DX(I).GT.TOL) LOGIC=1
22 CONTINUE
C
IF(LOGIC.LT.0) GOTO 8
C
DO 25 I=1, N
    FOLD(I)=F(I)
    XOLD(I)=XNEW(I)
25 CONTINUE
C
C
C    --- CALCULATE PARTIAL DERIVATIVES IN JACOBIAN ---
C
C(1,1)= ACT1(1)+XNEW(1)*DACT1(1,1)
C(1,2)=-ACT2(1)-XNEW(2)*DACT2(1,1)
C(2,1)=-ACT1(2)+(1.DO-XNEW(1))*DACT1(2,1)
C(2,2)= ACT2(2)-(1.DO-XNEW(2))*DACT2(2,1)
C
C
C    --- FINISH LOADING C MATRIX WITH F VECTOR ---
C
DO 30 I=1, N
    C(I,M)=-F(I)
30 CONTINUE
C
C
C    *** GAUSS-JORDAN ALGORITHM ***
C
C
C    --- INITIALIZE ALL VECTORS AND MATRICES ---
C
DO 40 I=1, N
    IR(I)=0

```

```

      DX(I)=0. DO
      JJ=G
      JM=0
40    CONTINUE
C
      DO 50 K=1,N
      PK=0. DO
C
C    --- LOCATE PIVOT ELEMENT ---
C
      DO 60 I=1,N
      IF(I.EQ.IR(I)) GOTO 60
C
      DO 70 IK=1,N
      P=DABS(C(I,IK))
      IF(P.LT.PK) GOTO 70
      PK=P
      JJ=I
      JM=IK
70    CONTINUE
C
60    CONTINUE
C
      IR(JJ)=JJ
C
C    --- NORMALIZATION STEP ---
C
      DO 80 JR=1,M
      IF(JM.EQ.JR) GOTO 80
      C(JJ, JR)=C(JJ, JR)/C(JJ, JM)
80    CONTINUE
C
      C(JJ, JM)=1. DO
C
C    --- REDUCTION STEP ---
C
      DO 90 I=1,N
      IF(I.EQ.JJ) GOTO 90
C
      DO 100 JR=1,M
      IF(JR.EQ.JM) GOTO 100
      C(I, JR)=C(I, JR)-C(I, JM)*C(JJ, JR)
100   CONTINUE
C
      C(I, JM)=0. DO
90    CONTINUE
C
50    CONTINUE
C
C    *** END GAUSS-JORDAN MAXIMUM PIVOT ROUTINE ***
C
C    --- RECOVER SOLUTION VECTOR --

```

```

C          DO 110 I=1,N
C
C              DO 120 J=1,N
C                  IF((C(I,J).LT.1.DO).OR.(C(I,J).GT.1.DO)) GOTO 120
C                  DX(J)=C(I,M)
120          CONTINUE
C
C 110          CONTINUE
C
C          --- CORRECT ELEMENTS OF X VECTOR ---
C
C          DO 130 I=1,N
C              XNEW(I)=XNEW(I)+DX(I)
C              IF(XNEW(I).LT.0.DO) XNEW(I)=0.DO
C              IF(XNEW(I).GT.1.DO) XNEW(I)=1.DO
130          CONTINUE
C
C 999 CONTINUE
C
C          GOTO 9
C
C          <<< END OF ITERATION LOOP >>>
C
C          --- PRINT FINAL MATRIX (C) AND SOLUTION VECTOR ---
C
C      8 WRITE(*,906) INDEX
C        WRITE(7,906) INDEX
C
C          DO 140 I=1,N
C              WRITE(7,907) (C(I,J),J=1,N),C(I,M)
C              WRITE(*,907) (C(I,J),J=1,N),C(I,M)
140          CONTINUE
C
C          WRITE(7,908)
C          WRITE(*,908)
C
C          DO 150 I=1,N
C              WRITE(7,909) I,XNEW(I),I,F(I)
C              WRITE(*,909) I,XNEW(I),I,F(I)
150          CONTINUE
C
C          WRITE(7,910)
C          WRITE(*,910)
C
C          --- LOAD SUMMARY VECTORS FOR FINAL RESULTS TABLE ---
C
C          DO 160 I=1,N
C              DO 155 J=1,N
C                  IF(J.NE.I) GOTO 152
C                  IE(J)=IE(J)+1
152          IF(XNEW(J).GT.XNEW(I)) IE(J)=IE(J)+1

```

```

155     CONTINUE
C
      FMIN(KN, I)=F(I)
160     CONTINUE
C
      --- SORT COMPOSITION SUMMARY VECTOR ---
C
      DO 165 I=1, N
        XE(KN, I)=XNEW(IE(I))
165     CONTINUE
C
      --- CONVERT AQUEOUS CONCENTRATION TO PPMW UNITS ---
C
      XE(KN, 1)=1000000. DO/(1. DO+((1. DO/XE(KN, 1))-1. DO)*XMW(2)/XMW(1))
C
      GOTO 777
C
      --- CONVERGENCE FAILURE MESSAGE ---
C
9     WRITE(*, 911) IMAX
      WRITE(7, 911) IMAX
      WRITE(7, 910)
      WRITE(*, 910)
C
      --- LOAD SUMMARY VECTORS WITH DEFAULT VALUES ---
C
      DO 170 I=1, N
        XE(KN, I)=-999. DO
        FMIN(KN, I)=-999. DO
        IF(XNEW(1).LT. (1. DO-TOL)) GOTO 170
        XE(KN, I)=XNEW(I)
        FMIN(KN, I)=F(I)
170     CONTINUE
C
777     CONTINUE
C
      --- PRINT TABLE OF FINAL MISCIBILITY VALUES ---
C
      WRITE(7, 912)
      WRITE(*, 912)
      WRITE(7, 913)
      WRITE(*, 913)
C
      DO 180 I=1, NDS
        WRITE(7, 914) NAME(I), TEMP(I), (XE(I, J), J=1, N)
        WRITE(*, 914) NAME(I), TEMP(I), (XE(I, J), J=1, N)
180     CONTINUE
C
      --- FORMAT STATEMENTS ---
C
901     FORMAT(A25)
902     FORMAT(//, 5X, 'DATA SET', I3, ' : ', A25, ' AT', F7.2, ' K')
903     FORMAT(5X, 8(' - '), 6X, 37(' - '), //)

```



```

904 FORMAT(/, 20X, 'PLEASE WAIT... CALCULATIONS PROCEEDING... ^, /)
905 FORMAT(30X, 'ITERATION ', I3)
906 FORMAT(/, 15X, 'FINAL SOLUTION MATRIX: (AFTER', I3, ' ITERATIONS)',
&/)
907 FORMAT(15X, 7E15.5)
908 FORMAT(/, 5X, 'SOLUTION VECTOR:', 15X, 'FUNCTION VECTOR:', /)
909 FORMAT(5X, 'XNEW(', I2, ') =', E11.5, 10X, 'F(', I2, ') =', E11.5)
910 FORMAT(///)
911 FORMAT(/, 15X, 'ALGORITHM DID NOT CONVERGE AFTER', I4, ' ITERATIONS' /
912 FORMAT(/, 8X, 'SYSTEM', 7X, 'TEMP (K)', 5X, 'X1 (PPMW, I)', 5X,
&'X1 (M-F, II)')
913 FORMAT(8X, 6(' - '), 7X, 8(' - '), 5X, 12(' - '), 5X, 12(' - '), /)
914 FORMAT(1X, A15, 5X, F8.2, 4X, F8.1, 5X, 1E17.5)
RETURN
END

```

```

C
C
C .....
C *
C *
C * <<< SUBROUTINE TO CALCULATE THE ACTIVITY COEFFICIENTS >>>
C *
C *
C .....
C SUBROUTINE UNIFA(NDIF, NACT, NC, NG, T, X, ACT, DACT, TACT)
C .....
C IMPLICIT REAL*8(A-H, O-Z)
C COMMON/UNIF/RT(10, 10), QT(10, 10), TAU(10, 10), S(10, 10), F(10), Q(10),
C * R(10), P(10, 10)
C DIMENSION X(10), GAM(10), ACT(10), DACT(10, 10), THETA(10), PHI(10),
C * RI(10), QI(10), QIL(10), RIL(10), QID(10), ETAL(10), TACT(10), U(10, 10),
C * V(10, 10), DETA(10), DS(10, 10), ETA(10), TETAR(10), H3(10, 1
C
C THETS=0. DO
C PHS=0. DO
C
C DO 10 I=1, NC
C THETA(I)=X(I)*Q(I)
C PHI(I)=R(I)*X(I)
C THETS=THETS+THETA(I)
C PHS=PHS+PHI(I)
C 10 CONTINUE
C
C DO 20 I=1, NC
C THETA(I)=THETA(I)/THETS
C PHI(I)=PHI(I)/PHS
C RI(I)=R(I)/PHS
C RIL(I)=DLOG(RI(I))
C QI(I)=Q(I)/THETS
C QID(I)=1. DO-R(I)/QI(I)
C QIL(I)=DLOG(QI(I))
C 20 CONTINUE
C

```

```

DO 30 I=1, NC
  XX=F(I)+Q(I)*(1.DO-QIL(I))-RI(I)+RIL(I)
  XX=XX-5.DO*Q(I)*(QID(I)+RIL(I)-QIL(I))
  GAM(I)=XX
30 CONTINUE
C
DO 50 I=1, NG
  TETAR(I)=O.DO
  ETA(I)=O.DO
C
DO 40 J=1, NC
  ETA(I)=ETA(I)+S(I, J)*X(J)
  TETAR(I)=TETAR(I)+QT(I, J)*X(J)
40 CONTINUE
C
  ETAL(I)=DLOG(ETA(I))
50 CONTINUE
C
DO 70 I=1, NC
C
DO 60 J=1, NG
  U(J, I)=S(J, I)/ETA(J)
  V(J, I)=U(J, I)*TETAR(J)
  GAM(I)=GAM(I)-V(J, I)-QT(J, I)*ETAL(J)
60 CONTINUE
C
  ACT(I)=DEXP(GAM(I))
  IF(NACT.EQ.1) ACT(I)=ACT(I)*X(I)
70 CONTINUE
C
  IF(NDIF.EQ.0) GO TO 160
  IF(NDIF.EQ.2) GO TO 110
C
DO 90 I=1, NC
DO 90 J=1, NC
  XX=Q(I)*QI(J)*(1.DO-5.DO*QID(I)+QID(J))*(1.DO-RI(I))*
&    (1.DO-RI(J))
C
DO 80 K=1, NG
  XX=XX+U(K, I)*(V(K, J)-QT(K, J))-U(K, J)*QT(K, I)
80 CONTINUE
C
  DACT(I, J)=XX
  DACT(J, I)=XX
  IF(NACT.EQ.1) GO TO 90
  DACT(I, J)=DACT(I, J)*ACT(I)
  IF(J.EQ.I) GO TO 90
  DACT(J, I)=DACT(J, I)*ACT(J)
90 CONTINUE
C
  IF(NACT.EQ.0) GO TO 110
C
DO 100 I=1, NC

```

```

DO 100 J=1,NC
  DACT(I,J)=ACT(I)*(DACT(I,J)-1.DO)
  IF(J.EQ.I) DACT(I,J)=DACT(I,J)+DEXP(GAM(I))
100 CONTINUE
110 CONTINUE
C
  IF(NDIF.EQ.1) GO TO 160
C
  DO 130 K=1,NG
    DETA(K)=0.DO
  DO 130 I=1,NC
    DS(K,I)=0.DO
C
    DO 120 M=1,NG
      IF(QT(M,I).EQ.0.DO) GO TO 120
      DS(K,I)=DS(K,I)-QT(M,I)*DLOG(TAU(M,K))*TAU(M,K)/T
120 CONTINUE
C
    DETA(K)=DETA(K)+DS(K,I)*X(I)
130 CONTINUE
C
  DO 150 I=1,NC
    TACT(I)=0.DO
C
    DO 140 K=1,NG
      H3(K,I)=(-S(K,I)*DETA(K)/ETA(K)+DS(K,I))/ETA(K)
      HH=H3(K,I)*(TETAR(K)-QT(K,I)*ETA(K)/S(K,I))
      TACT(I)=TACT(I)-HH
140 CONTINUE
C
    TACT(I)=TACT(I)*ACT(I)
150 CONTINUE
160 CONTINUE
  RETURN
  END
C
C
C .....
C *
C *
C *   PARAM CALCULATES SOME COMPOSITION-INDEPENDENT QUANTITIES:
C *   TAU, S, AND F, TO BE USED IN UNIFA.  PARAM MUST BE CALLED
C *   PRIOR TO UNIFA.
C *
C *
C .....
C
  SUBROUTINE PARAM(NC,NG,T)
C .....
  IMPLICIT REAL*8(A-H,O-Z)
  COMMON/UNIF/RT(10,10),QT(10,10),TAU(10,10),S(10,10),F(10),Q(10),
  * R(10),P(10,10)
C
  DO 10 I=1,NG

```

```

DO 10 J=1,NG
  TAU(I,J)=DEXP(-P(I,J)/T)
10 CONTINUE
C
DO 20 I=1,NC
DO 20 K=1,NG
  S(K,I)=0.DO
DO 20 M=1,NG
  S(K,I)=S(K,I)+QT(M,I)*TAU(M,K)
20 CONTINUE
C
DO 30 I=1,NC
  F(I)=1.DO
DO 30 J=1,NG
  F(I)=F(I)+QT(J,I)*DLOG(S(J,I))
30 CONTINUE
C
RETURN
END

```

*
*
* UVLE CONTAINS BUILT-IN UNIFAC VLE-PARAMETERS *
*
*

SUBROUTINE UVLE(RI, QI, AI, MAIN)

THE MAIN GROUPS ARE:

- | | | | | |
|----------|----------|-------------|----------|-------------|
| 1 CH2 | 2 C=C | 3 ACH | 4 ACCH2 | 5 OH |
| 6 CH3OH | 7 H2O | 8 ACOH | 9 CH2CO | 10 CHO |
| 11 CCOO | 12 HCOO | 13 CH2O | 14 CNH2 | 15 CNH |
| 16 (C)3N | 17 ACNH2 | 18 PYRIDINE | 19 CCN | 20 COOH |
| 21 CCL | 22 CCL2 | 23 CCL3 | 24 CCL4 | 25 ACCL |
| 26 CNO2 | 27 ACNO2 | 28 CS2 | 29 CH3SH | 30 FURFURAL |
| 31 DOH | 32 I | 33 BR | 34 C=-C | 35 DMSO |
| 36 ACRY | 37 CLCC | 38 ACF | 39 DMF | 40 CF2 |

THE SUB GROUPS ARE:

- | | | | | |
|----------|-----------|-----------|-----------|-----------|
| 1 CH3 | 2 CH2 | 3 CH | 4 C | 5 CH2=CH |
| 6 CH=CH | 7 CH2=C | 8 CH=C | 9 C=C | 10 ACH |
| 11 AC | 12 ACCH3 | 13 ACCH2 | 14 ACCH | 15 OH |
| 16 CH3OH | 17 H2O | 18 ACOH | 19 CH3CO | 20 CH2CO |
| 21 CHO | 22 CH3COO | 23 CH2COO | 24 HCOO | 25 CH3O |
| 26 CH2O | 27 CH-O | 28 FCH2O | 29 CH3NH2 | 30 CH2NH2 |
| 31 CHNH2 | 32 CH3NH | 33 CH2NH | 34 CHNH | 35 CH3N |
| 36 CH2N | 37 ACNH2 | 38 C5H5N | 39 C5H4N | 40 C5H3N |
| 41 CH3CN | 42 CH2CN | 43 COOH | 44 HCOOH | 45 CH2CL |
| 46 CHCL | 47 CCL | 48 CH2CL2 | 49 CHCL2 | 50 CCL2 |

```

C      51 CHCL3 .... 52 CCL3 .... 53 CCL4 .... 54 ACCL .... 55 CH3NO2 ...
C      56 CH2NO2 ... 57 CHNO2 ... 58 ACNO2 ... 59 CS2 ..... 60 CH3SH ....
C      61 CH2SH .... 62 FURFURAL 63 (CH2OH)2 64 I ..... 65 BR .....
C      66 CH=-C .... 67 C=-C .... 68 DMSO .... 69 ACRY .... 70 CL(C=-C) .
C      71 ACF ..... 72 DMF-1 ... 73 DMF-2 ... 74 CF3 ..... 75 CF2 .....
C      76 CF .....
C
C

```

```

C*****
C

```

```

C      IMPLICIT REAL*8(A-H,O-Z)
C      COMMON/UNIF/RT(10,10),QT(10,10),TAU(10,10),S(10,10),F(10),Q(10),
C      *      R(10),P(10,10)
C      DIMENSION AI(40,40),RR(76),QQ(76),RI(76),QI(76),MAINSG(76),
C      *      MAIN(76)

```

```

C      DATA MAINSG/ 4*1, 5*2, 2*3, 3*4, 1*5, 1*6, 1*7, 1*8, 2*9,1*10,
C      *      2*11,1*12,4*13,3*14,3*15,2*16,1*17,3*18,2*19,2*20,
C      *      3*21,3*22,2*23,1*24,1*25,3*26,1*27,1*28,2*29,1*30,
C      *      1*31,1*32,1*33,2*34,1*35,1*36,1*37,1*38,2*39,3*40/

```

```

C      DATA RR/0.9011,0.6744,0.4469,0.2195,1.3454,1.1167,1.1173,0.8886,
C      *0.6605,0.5313,0.3652,1.2663,1.0396,0.8121,1.0,1.4311,0.92,0.8952,
C      *1.6724,1.4457,0.9980,1.9031,1.6764,1.2420,1.1450,0.9183,0.6908,
C      *0.9183,1.5959,1.3692,1.1417,1.4337,1.2070,0.9795,1.1865,0.9597,
C      *1.0600,2.9993,2.8332,2.6670,1.8701,1.6434,1.3013,1.5280,1.4654,
C      *1.2380,1.0060,2.2564,2.0606,1.8016,2.8700,2.6401,3.3900,1.1562,
C      *2.0086,1.7818,1.5544,1.4199,2.0570,1.8770,1.6510,3.1680,2.4088,
C      *1.2640,0.9492,1.2920,1.0613,2.8266,2.3144,0.7910,0.6948,3.0856,
C      *2.6322,1.4060,1.0105,0.6150/

```

```

C      DATA QQ/0.848,0.540,0.228,0.000,1.176,0.867,0.988,0.676,0.485,
C      *0.400,0.120,0.968,0.660,0.348,1.200,1.432,1.400,0.680,1.488,
C      *1.180,0.948,1.728,1.420,1.188,1.088,0.780,0.468,1.100,1.544,
C      *1.236,0.924,1.244,0.936,0.624,0.940,0.632,0.816,2.113,1.833,
C      *1.553,1.724,1.416,1.224,1.532,1.264,0.952,0.724,1.988,1.684,
C      *1.448,2.410,2.184,2.910,0.844,1.868,1.560,1.248,1.104,1.650,
C      *1.676,1.368,2.481,2.248,0.992,0.832,1.088,0.784,2.472,2.052,
C      *0.724,0.524,2.736,2.120,1.380,0.920,0.460/

```

```

C      OPEN(9,FILE='AVLE.DAT',STATUS='OLD',ACCESS='SEQUENTIAL',
C      *FORM='FORMATTED')
C      OPEN(10,FILE='AVLE.OUT',STATUS='NEW',ACCESS='SEQUENTIAL',
C      *FORM='FORMATTED')

```

```

C      DO 5 I=1,76
C          RI(I)=0.DO
C          QI(I)=0.DO
C          MAIN(I)=0.DO

```

```

C      S CONTINUE

```

```

C      DO 20 I=1,40
C          DO 10 J=1,40
C              AI(I,J)=0.DO

```

```

10     CONTINUE
20     CONTINUE
C
    DO 40 I=1,40
      DO 30 J=1,8
        READ(9,*,END=40) (AI((S*(J-1)+K),I), K=1,5)
30     CONTINUE
40     CONTINUE
C
    DO 50 I=1,76
      RI(I)=RR(I)
      QI(I)=QQ(I)
      MAIN(I)=MAINSG(I)
50     CONTINUE
C
    DO 70 I=1,40
      DO 60 J=1,8
        WRITE(10,999) (AI((5*(J-1)+K),I), K=1,5)
60     CONTINUE
70     CONTINUE
C
C     --- FORMAT STATEMENTS ---
C
999    FORMAT(SF10.3)
C
    RETURN
    END
C
C
C
C *****
C *
C *
C *             ULLE CONTAINS BUILT-IN UNIFAC LLE-PARAMETERS
C *
C *
C *****
C
C   SUBROUTINE ULLE(RI, QI, AI, MAIN)
C *****
C
C   THE MAIN GROUPS ARE:
C   1 CH2 ..... 2 C=C ..... 3 ACH ..... 4 ACCH2 ..... 5 OH .....
C   6 P1 ..... 7 P2 ..... 8 H2O ..... 9 ACOH ..... 10 CH2CO .....
C   11 CHO ..... 12 FURFURAL 13 COOH ..... 14 COOC ..... 15 CH2O .....
C   16 CCL ..... 17 CCL2 .... 18 CCL3 ..... 19 CCL4 ..... 20 ACCL .....
C   21 CCN ..... 22 ACNH2 ... 23 CNO2 ..... 24 ACNG2 ... 25 DOH .....
C   26 DEOH ..... 27 PYRIDINE 28 TCE ..... 29 MFA ..... 30 DMFA .....
C   31 TMS ..... 32 DMSO ....
C
C   THE SUB GROUPS ARE:
C   1 CH3 ..... 2 CH2 ..... 3 CH ..... 4 C ..... 5 CH2=CH ...
C   6 CH=CH ..... 7 CH=C ..... 8 CH2=C ..... 9 ACH ..... 10 AC .....
C   11 ACCH3 ..... 12 ACCH2 ... 13 ACCH ..... 14 OH ..... 15 P1 .....

```

```

C      16 P2 ..... 17 H2O ..... 18 ACOH ..... 19 CH3CO ... 20 CH2CO ...
C      21 CHO ..... 22 FURFURAL 23 COOH ..... 24 HCOOH ... 25 CH3COO ..
C      26 CH2COO ... 27 CH3O ..... 28 CH2O ..... 29 CH-O ..... 30 FCH2O ...
C      31 CH2CL ..... 32 CHCL ..... 33 CCL ..... 34 CH2CL2 .. 35 CHCL2 ...
C      36 CCL2 ..... 37 CHCL3 ... 38 CCL3 ..... 39 CCL4 ..... 40 ACCL .....
C      41 CH3CN ..... 42 CH2CN ... 43 ACNH2 ... 44 CH3NO2 .. 45 CH2NO2 ..
C      46 CHNO2 ..... 47 ACNO2 ... 48 (CH2OH)2 49 (HOMM)20 50 C5H5N ...
C      51 C5H4N ..... 52 C5H3N ... 53 CCL2=CHCL 54 HCONHCH3 55 HCON(CH3)
C      56 (CH2)4SO2 57 (CH2)2SO

```

```

C
C .....
C

```

```

      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/UNIF/RT(10,10),QT(10,10),TAU(10,10),S(10,10),F(10),Q(10),
      *          R(10),P(10,10)
      DIMENSION AI(40,40),RR(57),QQ(57),RI(76),QI(76),MAINSG(57),
      *          MAIN(76)

```

```

C
      DATA MAINSG/ 4*1, 4*2, 2*3, 3*4, 1*5, 1*6, 1*7, 1*8, 1*9, 2*10,
      *             1*11, 1*12, 2*13, 2*14, 4*15, 3*16, 3*17, 2*18, 1*19, 1*20,
      *             2*21, 1*22, 3*23, 1*24, 1*25, 1*26, 3*27, 1*28, 1*29, 1*30,
      *             1*31, 1*32/

```

```

C
      DATA RR/O.9011, 0.6744, 0.4469, 0.2195, 1.3454, 1.1167, 0.8886, 1.1173,
      *0.5313, 0.3652, 1.2663, 1.0396, 0.8121, 1.0000, 3.2499, 3.2491, 0.9200,
      *0.8952, 1.6724, 1.4457, 0.9980, 3.1680, 1.3013, 1.5280, 1.9031, 1.6764,
      *1.1450, 0.9183, 0.6908, 0.9183, 1.4654, 1.2380, 1.0060, 2.2564, 2.0606,
      *1.8016, 2.8700, 2.6401, 3.3900, 1.1562, 1.8701, 1.6434, 1.0600, 2.0086,
      *1.7818, 1.5544, 1.4199, 2.4088, 4.0013, 2.9993, 2.8332, 2.6670, 3.3092,
      *2.4317, 3.0856, 4.0358, 2.8266/

```

```

C
      DATA QQ/O.848, 0.540, 0.228, 0.000, 1.176, 0.867, 0.676, 0.988, 0.400,
      *0.120, 0.968, 0.660, 0.348, 1.200, 3.128, 3.124, 1.400, 0.680, 1.488,
      *1.180, 0.948, 2.484, 1.224, 1.532, 1.728, 1.420, 1.088, 0.780, 0.468,
      *1.100, 1.264, 0.952, 0.724, 1.988, 1.684, 1.448, 2.410, 2.184, 2.910,
      *0.844, 1.724, 1.416, 0.816, 1.868, 1.560, 1.248, 1.104, 2.248, 3.568,
      *2.113, 1.833, 1.553, 2.860, 2.192, 2.736, 3.200, 2.472/

```

```

C
      OPEN(11, FILE='ALLE.DAT', STATUS='OLD', ACCESS='SEQUENTIAL',
      *FORM='FORMATTED')
      OPEN(12, FILE='ALLE.OUT', STATUS='NEW', ACCESS='SEQUENTIAL',
      *FORM='FORMATTED')

```

```

C
      DO 5 I=1,76
          RI(I)=0.DO
          QI(I)=0.DO
          MAIN(I)=0.DO
5 CONTINUE

```

```

C
      DO 20 I=1,40
          DO 10 J=1,40
              AI(I,J)=0.DO
10 CONTINUE

```

```

C
20 CONTINUE
C
DO 40 I=1,32
DO 30 J=1,6
READ(11,*,END=40) (AI((5*(J-1)+K),I), K=1,5)
30 CONTINUE
READ(11,*,END=40) (AI(K,I), K=31,32)
40 CONTINUE

```

```

C
DO 50 I=1,57
RI(I)=RR(I)
QI(I)=QQ(I)
MAIN(I)=MAINSG(I)
50 CONTINUE

```

```

C
DO 70 I=1,32
DO 60 J=1,6
WRITE(12,999) (AI((5*(J-1)+K),I), K=1,5)
60 CONTINUE
WRITE(12,999) (AI(K,I), K=31,32)
70 CONTINUE

```

```

C
C
C --- FORMAT STATEMENTS ---

```

```

C
999 FORMAT(SF10.3)

```

```

C
RETURN
END

```

```

C
C
C .....
C *
C *
C * ENVIRON CONTAINS "ENVIRONMENTAL" UNIFAC VLE-PARAMETERS
C *
C *
C .....

```

```

C
SUBROUTINE ENVIRON(RI,QI,AI,MAIN)
C
C .....

```

```

C
THE MAIN GROUPS ARE:
C
1 CH2 ..... 2 C=C ..... 3 ACH ..... 4 ACCH2 ..... 5 OH .....
C
6 CH3OH ..... 7 H2O ..... 8 ACOH ..... 9 CH2CO ... 10 CHO .....
C
11 CCOO ..... 12 HCOO .... 13 CH2O ..... 14 CNH2 ..... 15 CNH .....
C
16 (C)3N ..... 17 ACNH2 ... 18 PYRIDINE 19 CCN ..... 20 COOH .....
C
21 CCL ..... 22 CCL2 .... 23 CCL3 ..... 24 CCL4 ..... 25 ACCL .....
C
26 CNO2 ..... 27 ACNO2 ... 28 CS2 ..... 29 CH3SH ... 30 FURFURAL .
C
31 DOH ..... 32 I ..... 33 BR ..... 34 C=-C ..... 35 DMSO .....
C
36 ACRY ..... 37 CLCC .... 38 ACF ..... 39 DMF ..... 40 CF2 .....
C

```

```

C
THE SUB GROUPS ARE:
C
1 CH3 ..... 2 CH2 ..... 3 CH ..... 4 C ..... 5 CH2=CH ...

```



```

C      6 CH=CH ..... 7 CH2=C ..... 8 CH=C ..... 9 C=C ..... 10 ACH .....
C     11 AC ..... 12 ACCH3 ... 13 ACCH2 ... 14 ACCH ..... 15 OH .....
C     16 CH3OH ..... 17 H2O ..... 18 ACOH ..... 19 CH3CO ... 20 CH2CO ...
C     21 CHO ..... 22 CH3COO .. 23 CH2COO .. 24 HCOO ..... 25 CH3O .....
C     26 CH2O ..... 27 CH-O ..... 28 FCH2O ... 29 CH3NH2 .. 30 CH2NH2 ...
C     31 CHNH2 ..... 32 CH3NH ... 33 CH2NH ... 34 CHNH ..... 35 CH3N .....
C     36 CH2N ..... 37 ACNH2 ... 38 C5H5N ... 39 C5H4N ... 40 C5H3N ...
C     41 CH3CN ..... 42 CH2CN ... 43 COOH ..... 44 HCOOH ... 45 CH2CL ...
C     46 CHCL ..... 47 CCL ..... 48 CH2CL2 .. 49 CHCL2 ... 50 CCL2 .....
C     51 CHCL3 ..... 52 CCL3 ... 53 CCL4 ..... 54 ACCL ..... 55 CH3NO2 ...
C     56 CH2NO2 ... 57 CHNO2 ... 58 ACNO2 ... 59 CS2 ..... 60 CH3SH .....
C     61 CH2SH ..... 62 FURFURAL 63 (CH2OH)2 64 I ..... 65 BR .....
C     66 CH=-C ..... 67 C=-C ..... 68 DMSO ..... 69 ACRY ..... 70 CL(C=-C) .
C     71 ACF ..... 72 DMF-1 ... 73 DMF-2 ... 74 CF3 ..... 75 CF2 .....
C     76 CF .....
C
C

```

```

C.....
C

```

```

      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/UNIF/RT(10,10),QT(10,10),TAU(10,10),S(10,10),F(10),Q(10),
      *          R(10),P(10,10)
      DIMENSION AI(40,40),RR(76),QQ(76),RI(76),QI(76),MAINSG(76),
      *          MAIN(76)

```

```

      DATA MAINSG/ 4*1, 5*2, 2*3, 3*4, 1*5, 1*6, 1*7, 1*8, 2*9,1*10,
      *          2*11,1*12,4*13,3*14,3*15,2*16,1*17,3*18,2*19,2*20,
      *          3*21,3*22,2*23,1*24,1*25,3*26,1*27,1*28,2*29,1*30,
      *          1*31,1*32,1*33,2*34,1*35,1*36,1*37,1*38,2*39,3*40/

```

```

      DATA RR/0.9011,0.6744,0.4469,0.2195,1.3454,1.1167,1.1173,0.8886,
      *0.6605,0.5313,0.3652,1.2663,1.0396,0.8121,1.0,1.4311,0.92,0.8952,
      *1.6724,1.4457,0.9980,1.9031,1.6764,1.2420,1.1450,0.9183,0.6908,
      *0.9183,1.5959,1.3692,1.1417,1.4337,1.2070,0.9795,1.1865,0.9597,
      *1.0600,2.9993,2.8332,2.6670,1.8701,1.6434,1.3013,1.5280,1.4654,
      *1.2380,1.0060,2.2564,2.0606,1.8016,2.8700,2.6401,3.3900,1.1562,
      *2.0086,1.7818,1.5544,1.4199,2.0570,1.8770,1.6510,3.1680,2.4088,
      *1.2640,0.9492,1.2920,1.0613,2.8266,2.3144,0.7910,0.6948,3.0856,
      *2.6322,1.4060,1.0105,0.6150/

```

```

      DATA QQ/0.848,0.540,0.228,0.000,1.176,0.867,0.988,0.676,0.485,
      *0.400,0.120,0.968,0.660,0.348,1.200,1.432,1.400,0.680,1.488,
      *1.180,0.948,1.728,1.420,1.188,1.088,0.780,0.468,1.100,1.544,
      *1.236,0.924,1.244,0.936,0.624,0.940,0.632,0.816,2.113,1.833,
      *1.553,1.724,1.416,1.224,1.532,1.264,0.952,0.724,1.988,1.684,
      *1.448,2.410,2.184,2.910,0.844,1.868,1.560,1.248,1.104,1.650,
      *1.676,1.368,2.481,2.248,0.992,0.832,1.088,0.784,2.472,2.052,
      *0.724,0.524,2.736,2.120,1.380,0.920,0.460/

```

```

      OPEN(13,FILE='AENV.DAT',STATUS='OLD',ACCESS='SEQUENTIAL',
      *FORM='FORMATTED')
      OPEN(14,FILE='AENV.OUT',STATUS='NEW',ACCESS='SEQUENTIAL',
      *FORM='FORMATTED')

```

```

DO 5 I=1,76
  RI(I)=0.DO
  QI(I)=0.DO
  MAIN(I)=0.DO
5 CONTINUE
C
DO 20 I=1,40
  DO 10 J=1,40
    AI(I,J)=0.DO
10 CONTINUE
20 CONTINUE
C
DO 40 I=1,40
  DO 30 J=1,8
    READ(13,*,END=40) (AI((5*(J-1)+K),I), K=1,5)
30 CONTINUE
40 CONTINUE
C
DO 50 I=1,76
  RI(I)=RR(I)
  QI(I)=QQ(I)
  MAIN(I)=MAINSG(I)
50 CONTINUE
C
DO 70 I=1,40
  DO 60 J=1,8
    WRITE(14,999) (AI((5*(J-1)+K),I), K=1,5)
60 CONTINUE
70 CONTINUE
C
C    --- FORMAT STATEMENTS ---
C
999 FORMAT(5F10.3)
C
RETURN
END

```