UNITED STATES DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

HEWLETT PACKARD 9845A COMPUTER PROGRAM TO INVERT COMPLEX RESISTIVITY DATA

James C. Washburne

Open-File Report 83-198
1983

TABLE OF CONTENTS

Introduction..1

Program description

Input..3

Calculations..6

Output..8

Program limitations..10

Procedural guidelines..11

General program conventions and notes..12

User instructions: Program I/O..15

References cited..18

Figure 1a. Cole-Cole equivalent circuit model.
1b. A possible rock pore system..2

Figure 2. The effect of parametric variations on a single Cole-Cole model..4

Figure 3. Cole-Cole model parameter correlations..5

Table 1. List of symbols..iii

Table 2. List of major variables..14

Appendix A. Example program listing..19

Appendix B. Source program listing..23

Table 1--List of symbols

$Z(\omega)$	-Cole-Cole impedance
$\omega=2\pi f$	-angular frequency
$i=\sqrt{-1}$	-imaginary number
N	-number of data points
M	-number of dispersions
n	-number of parameters
$\nu=N-n-1$	-degrees of freedom
p_j	-Cole-Cole parameter (j=1,M*n)
R_o	-DC resistance or apparent resistivity
m_j	-chargeability
τ_j	-time constant
c_j	-frequency dependence
x_i	-measured sampling point (i=1,N)
y_i	-measured value
$\hat{y}_i$	-mean measurement
$y(x_i)$	-calculated value
$A_{ij}=\partial y_i/\partial p_j$	-derivative matrix
$\alpha= A^T A$	-curvature matrix
$\Delta\bar{p}$, Δp_j	-incremental parameter change
$\bar{d} = y_i-y(x_i)$	-difference vector
I	-identity matrix
λ	-stabilization factor
w_i	-weighting factor
s^2	-functional variance

σ_i	-sample variance
σ_j	-parameter variance
$\chi_\nu{}^2$	-reduced chi-square
$cov(p_{jk})$	-covariance matrix (j,k=1,M*n)
$cor(p_{jk})$	-correlation matrix
*	-multiplication

Introduction

This report describes an interactive Hewlett Packard (HP) 9845A computer program that inverts spectral induced polarization (SIP) or complex resistivity (CR) data. A complex (that is, frequency dependent) impedance is defined in terms of a multiplicative Cole-Cole model (Cole and Cole, 1941; Pelton, 1977; Washburne, 1982). Either amplitude, phase or amplitude-phase data can be inverted. This program utilizes Marquardt's (1963) maximum neighborhood method to stabilize the inverted derivative matrix. The reduced chi-square (Bevington, 1969) determines convergence. Both 'normal' and reversed (or 'negative') dispersions can be modeled. This allows first-order EM coupling effects to be removed from field data. Program operation is demonstrated using a calculated dispersion.

Below, SIP measurements, Cole-Cole models and inversion techniques are introduced briefly. Relevant computational procedures are reviewed in three sections under "Program description": 1) input, 2) calculations and 3) output. After noting a few program limitations, user-orientated procedural guidelines, program notes and program input/output (I/O) are reviewed.

An SIP measurement consists of recording a complex impedance (usually resolved into amplitude-phase or real-imaginary components) across a wide frequency range (.01-1000 Hz, for instance). The objective of a SIP measurement is to quantify charge polarization effects that are induced in the earth by an alternating electric field. In most rocks, this polarization can be characterized by a simple dispersion or by several dispersions. Specific mechanisms or polarizable sources cannot be attributed to these dispersions, however, because a wide range of mechanisms and overlap of active zones exist among several possible sources. Until more of the physical processes governing natural rock polarization are understood, Cole-Cole parameterization is limited to simply quantifing the spectrum of a given measurement.

Cole-Cole parameters can be defined in terms of a simple equivalent circuit (fig. 1a) or related to a hypothetical rock pore system (fig. 1b). More realistic situations consisting of a random network of these basic components also can be expressed in terms of a simple distributed circuit (Zonge, 1972). It is important to realize, however, that this solution is not unique in two

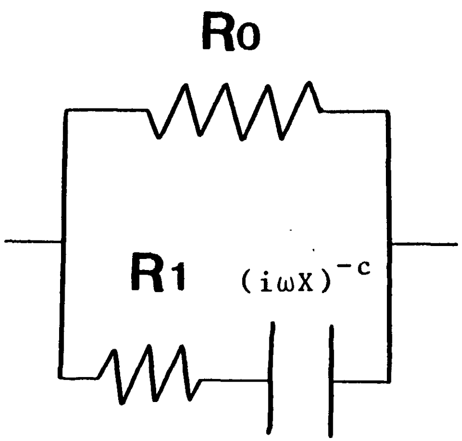


Figure la--Cole-Cole equivalent circuit model. The Cole-Cole parameters (R_0,m,τ,c) can be expressed in terms of a lossy circuit's impedance.

$$Z(\omega) = R_0\left(1 - m\left(1 - \frac{1}{1+(i\omega\tau)^c}\right)\right)$$

$$m = \frac{R_0}{R_0+R_1} \qquad \tau = X\left(R_0+R_1\right)^{1/c}$$

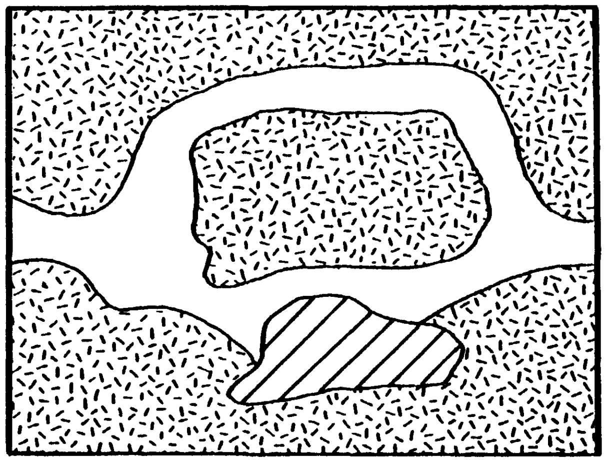


Figure lb--A possible rock pore system approximatingthe equivalent circuit on the left. Two possible electrolytic paths are open; the path below interacts with a polarizable particle (diagonal shading).

important respects: 1) the Cole-Cole model is only one of many possible mathematical representations and 2) inversion of Cole-Cole models involves a linearization of a nonlinear problem. Model parameters derived from noisy data are particularly nonunique. In general, an estimate of the parameters' nonuniqueness can be obtained by carefully examining the inversion's output, which usually includes: 1) statistical measures, such as parameter standard deviation and correlation matrix and 2) intermediate parameterizations (large fluctuations are a sign of poor resolution).

Program description: Input

The program requires as input sets of frequency (Hz), amplitude (ohm-m) and phase (mrad) measurements in order of ascending frequency. Subroutine Test (line 6130) demonstrates how the data arrays are filled. Four methods of interactively weighting the data are possible (see "Program operation"). Amplitude data are scaled using a simple logarithmic transformation; phase data are scaled using a hyperbolic sine transformation. The second transformation scales small ($-1<x<1$ mrad) phases linearly while larger positive and negative phases are scaled logarithmically.

Initial Cole-Cole parameters (m, τ and c) must be supplied interactively. Several families of curves (fig. 2) demonstrate the effect each parameter has on a dispersion's shape. Any parameter can be held constant to constrain the final solution. A logarithmic transformation of τ effectively limits its range to between 4 and -7. The ranges of m and c are physically constrained between 0 and 1 by the equivalent circuit representation of a Cole-Cole dispersion (Pelton etal., 1978). Constraining these two parameters with a simple trigonometric or hyperbolic transformation is ineffective because poorly resolvable parameters become fixed at unrealistic values. Therefore, it is best that these parameters be fixed as the need arises. Both m and c must be positive. Occasionally, a modified parameter becomes negative (it is displayed as a negative); however, its absolute value is used for subsequent calculations.

A weakness of the Cole-Cole model is the high inherent correlation between parameters. This is illustrated by inverting the systematically varying spectra in figure 2 and graphically summarizing the resulting parameter correlations (fig. 3). These results indicate that parametric ambiguity is greater when a dispersion is: strong (m), incomplete (τ) and/or broad (c).

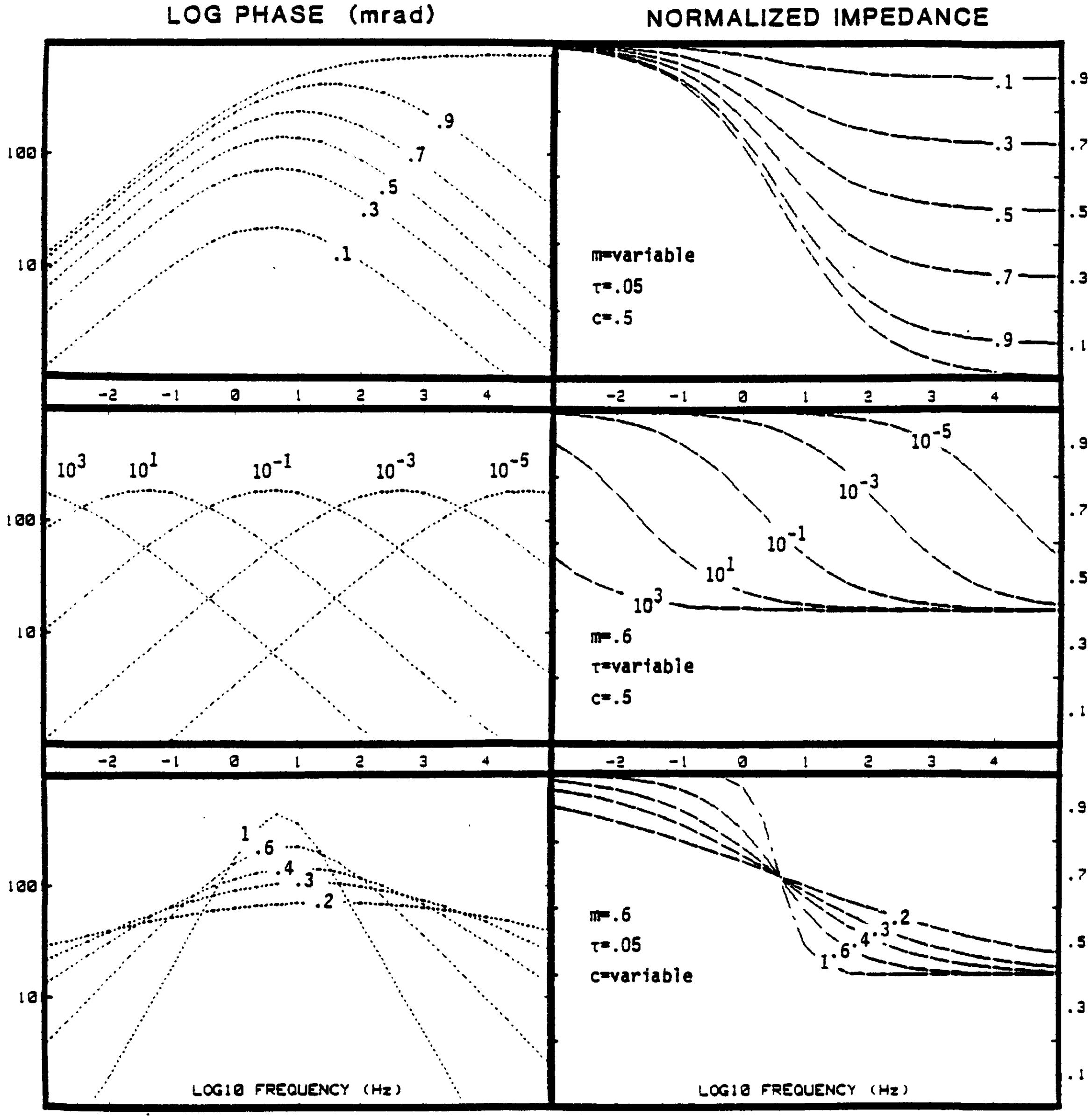


Figure 2--The effect of parametric variations on a single Cole-Cole model. Each family of curves was generated by holding two parameters constant and letting the third vary as indicated.

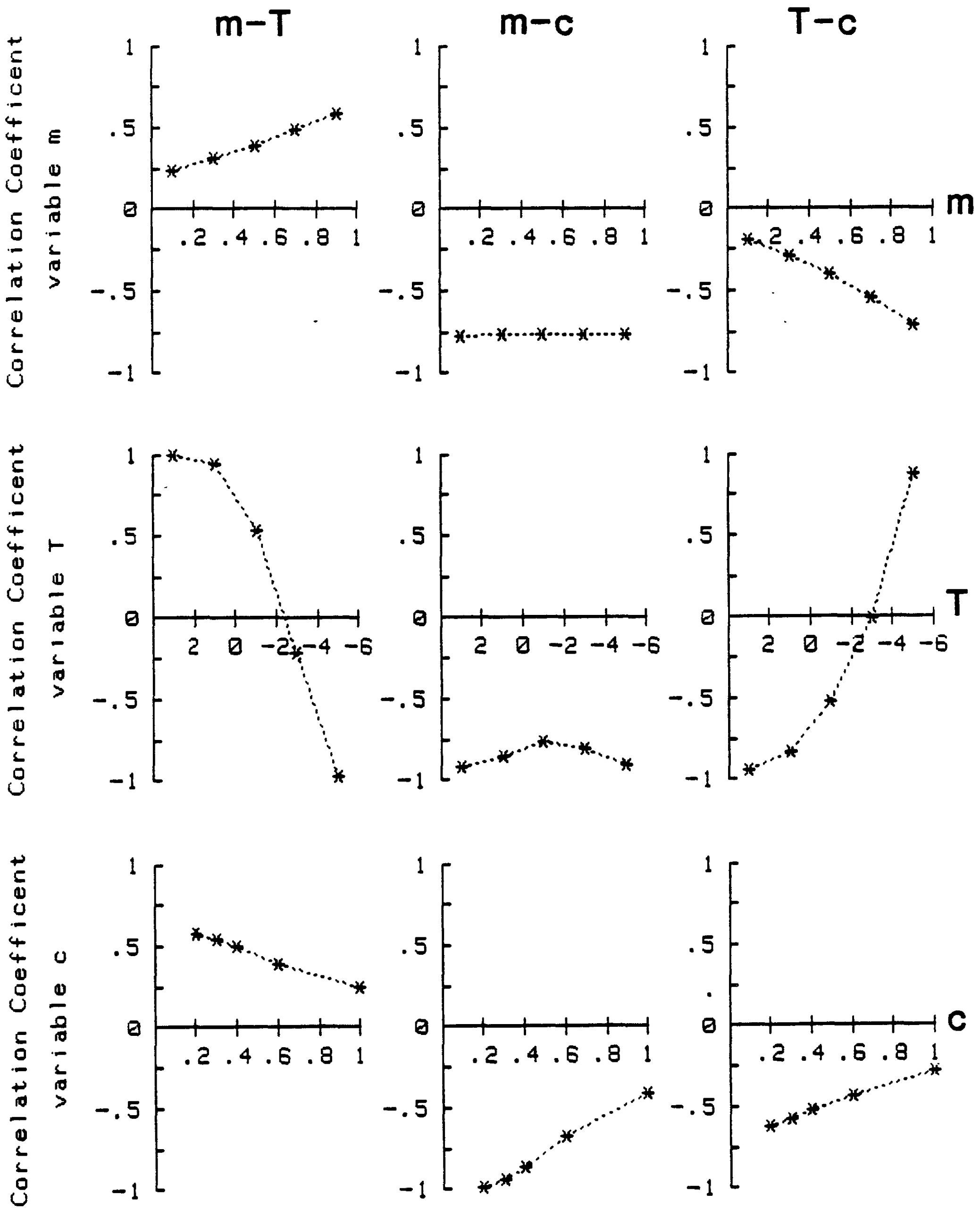


Figure 3--Cole-Cole model parameter correlations. Each dispersion in figure 2 has been inverted to form a measure of inherent parameter correlation. Constant parameters are m,τ,c=.6,.05,.5. Variable parameters are plotted along the horizontal axis.

Program description: Calculations

The forward solution (subroutine Colem) is expressed in terms of the real and imaginary parts of a complex impedance (equations 1-2), which is modeled by a product of Cole-Cole dispersions (equation 3) (B.D. Smith, pers.commun., 1981; Washburne, 1982).

$$Z_{real} = 1 - \frac{m\left(a + (\omega\tau)^{2c}\right)}{D_1} \qquad (1)$$

$$Z_{imag} = - \frac{m\ b}{D_1} \qquad (2)$$

$$Z(\omega) = R_o\ (\ 1 - m_1 f_1\ )(\ 1 - m_2 f_2\ ) \qquad (3)$$

where $a = (\omega\tau)^c \cos\ (c\ \pi/2)$

$b = (\omega\tau)^c \sin\ (c\ \pi/2)$

$D_1 = 1 + 2a + (\omega\tau)^{2c}$

$$f = \left(\ 1 - \frac{1}{1 + (i\omega\tau)^c}\ \right)$$

(see table 1 for a definition of these and other variables). This parameterization is useful to quantify many spectral curves simply for more detailed interpretational analysis.

The inversion subroutine (Curfit) searches a nxn parameter space for a minimum reduced chi-square between the weighted raw and inverted data points (Bevington, 1969). The impedance function is linearized by considering only the first two terms in its Taylor-series expansion (equation 4), where $Z(\omega)/R_o$ has been transformed to $y(x)$ for simplicity.

$$y_i = y(x_i) + \partial y_i/\partial p_j\ \Delta p_j \qquad (4)$$

In matrix notation this becomes (equation 5):

$$\bar{d} = A\ \Delta p \qquad (5)$$

Multiplying each side by A^T gives (equation 6):

$$A^T\ \bar{d} = A^T A\ \Delta p \qquad (6)$$

A Marquardt-like stabilization factor (λ) is added to the main diagonal of the curvature or normalized derivative

A Marquardt-like stabilization factor (λ) is added to the main diagonal of the curvature or normalized derivative matrix to keep it from becoming singular (Marquardt, 1963). This equation can now be solved for Δp (equation 7).

$$\Delta p = (A^T A + \lambda I)^{-1} A^T \bar{d} \qquad (7)$$

At each iteration, λ is changed. Initially, and away from the zone of convergence, λ is large and increasing-- the search approximating the gradient method. As a minimum is approached, however, λ is decreased and the search approximates the linearized expansion method, which converges quadratically in this region. Convergence is reached when either the incremental or relative chi-square error becomes less than some tolerance. The tolerance defaults to 10^{-4} for phase and amplitude-phase inversion options and to 10^{-5} for the inversion of amplitude. These might need to be modified for exceedingly good or noisy data sets. Because these tolerances are derived from experience with theoretical, laboratory and field data sets, it appears as though the inversion of logarithmic amplitude data is less sensitive to parametric variations than is the inversion of logarithmic phase data.

Either numeric or analytic derivatives can be used in this program. The only significant difference between the two methods is the much greater speed at which analytic derivatives are calculated; the time savings is a function of the number and resolvability of the parameters. Numerical derivatives are calculated by perturbing each parameter, in turn, by 10% and finding the differential: $\Delta y/\Delta x$. Analytic derivatives are derived from the partial derivatives (equations 8-10) of a single Cole-Cole dispersion.

$$\partial y/\partial m = 1/D_1 \, [\, a + (\omega\tau)^{2c} - i \, b \,] \qquad (8)$$

$$\partial y/\partial \tau = mc/(D_1\tau) \, [\, R + I \,] \qquad (9)$$

$$\partial y/\partial c = m/{D_1}^2 [R \ln(\omega\tau) - I\pi/2 + i(R\pi/2 + I \ln(\omega\tau))] \qquad (10)$$

where $R = 2 \, (\omega\tau)^{2c} + a[1+(\omega\tau)^{2c}]$

$I = b \, [1-(\omega\tau)^{2c}]$.

Since amplitude and phase are defined as (equations 11-12),

$$A = \sqrt{ {Z_{real}}^2 + {Z_{imag}}^2 } \qquad (11)$$

$$\phi = \tan(Z_{imag}/Z_{real}), \quad (12)$$

analytic derivatives have the form of equations 13-14:

$$\frac{\partial A}{\partial p} = \frac{1}{A} \left(\frac{\partial Z_{real}}{\partial p} + \frac{Z_{imag}}{Z_{real}} \frac{\partial Z_{imag}}{\partial p}\right) \quad (13)$$

$$\frac{\partial \phi}{\partial p} = \frac{1}{A^2} \left(\frac{\partial Z_{imag}}{\partial p} - \frac{Z_{imag}}{Z_{real}} \frac{\partial Z_{real}}{\partial p}\right) . \quad (14)$$

Pelton etal. (1978) note that a Cole-Cole model whose frequency dependence is one can be used to decouple 'normal' high frequency EM coupling effects from lower frequency IP effects. Although parametric resolution is lost (Major and Silic, 1980) and this procedure is valid only before the phase maximum in simple geoelectric environments (Webster, 1980), it is a useful first approximation. 'Negative' coupling effects can be removed in a like manner by using two opposing Cole-Cole dispersions to model the coupling (Hallof and Pelton, 1980) (equations 15-16).

$$Z(\omega) = R_o \ (1 - m_1 f_1) \ (1 - m_2 f_2) \quad (15)$$

$$Z(\omega) = R_o \ (1 - m_1 f_1) \ \underbrace{(1 - m_2 f_2) \ (1 + m_3 f_3)}_{\text{coupling}} \quad (16)$$

Residual phases are calculated by decomposing the modeled real and imaginary parts of the spectrum from the real and imaginary (that is, cosine, sine) representation of the original spectrum.

Program description: Output

The program outputs: 1) inverted Cole-Cole parameters, 2) several statistical measures of the inversion's fit and the parameter's ambiguity and 3) a plot of the raw and inverted dispersions.

There are several measures of "goodness of fit". Bevington (1969) defines the functional variance (s^2) of a set of data (y_i) to a fitting function ($y(x_i)$) as (equation 17):

$$s^2 = 1/\nu \sum_i w_i \ [y_i - y(x_i)]^2 \quad (17)$$

where the weighting function, w_i (equation 18), is normalized in terms of the sample variances, σ_i (equation 19).

$$w_i = \frac{1/\sigma_i^2}{1/N \sum_i 1/\sigma_i^2} \tag{18}$$

$$\sigma_i = \frac{1}{N} \sum_i (y_i - \hat{y}_i)^2 \tag{19}$$

The reduced chi-square (χ_ν^2) (equation 20) is a measure of how well a fitting function matches a series of observations and is found by taking the ratio of the functional variance to the weighted average of the sample variances ($\sigma_i^2 = w_i \sigma_i^2$).

$$\chi_\nu^2 = 1/\nu \sum_i [y_i - y(x_i)]^2 = s^2/\bar{\sigma}_i^2 \tag{20}$$

Ideally, the reduced chi-square is unity-- larger values indicating a poor fit, smaller values indicating a larger uncertainty in the observations. If sample variance is unknown, $\sigma_i^2=1$, and the reduced chi-square is simply the least-squares error.

Parameter error can be estimated from the covariance matrix (equation 21).

$$\mathrm{cov}(p_{jk}) = [\ A^T A\ ]^{-1}\ \chi_\nu^2 \tag{21}$$

Diagonal elements of $\mathrm{cov}(p_{jk})$ indicate parameter variance (σ_j), non-diagonal elements indicate parameter covariance. For reasonable parameterizations, the normalized covariance or correlation matrix (equation 22) gives a measure of the correlation between parameters.

$$\mathrm{cor}(p_{jk}) = \frac{\mathrm{cov}\ p_{jk}}{\sqrt{\mathrm{cov}\ p_{jj}\ \mathrm{cov}\ p_{kk}}} \tag{22}$$

A correlation of zero indicates an independent parameter. Coefficients of +1 and -1 indicate a strong correlation or inverse correlation, respectively.

Following the statistics, raw and inverted dispersions are plotted. A plot of logarithmic phase lag and normalized (by R_0) amplitude versus logarithmic frequency is preferred because Cole-Cole parameters are estimated easily from this presentation. Data can be plotted on a normalized real-imaginary axis diagram (also known as an Argand or Cole-Cole diagram). For the benefit of those who prefer this format, the raw data are plotted on a reduced real-imaginary diagram also.

Program limitations

When comparing this data to other Cole-Cole modeled data, beware of slight but parametrically significant differences in the way multiple dispersions are defined.

Three dispersion models are adequate to fit most laboratory and field SIP data. If more complicated models are desired, the 9845A's 64K byte memory must be extended or advantage taken of overlapped processing.

The validity of inversion statistics depends upon: 1) spectral resolvability in terms of the Cole-Cole model and 2) linearity of the Taylor-series approximation. At best, statistics serve only as relative indications of parametric stability and correlation.

The Cole-Cole model only approximates EM coupling. As a result, parameters associated with high frequency polarization dispersions are ambiguous. In general, however, this is a good enough approximation so that parameters associated with low frequency dispersions are more characteristic of a coupling-free environment.

Procedural Guidelines

1) Weight to 0 those points that are not part of a smooth spectral curve. One 'bad' point, especially at a frequency extreme, can wreck convergence.

2) Determine how many polarization dispersions are present. Be sure to note models that do not truly correspond to a single dispersion (that is, where impulsive or transitory behavior is not fit well by a simpler model). Minimize the number of models.

3) Initialize the model. Hold unresolvable parameters constant. Refer to Figure 2 if you have difficulty relating spectral shape to the Cole-Cole parameters. In general, resolvable parameters are characterized by: 1) clear non-zero slopes (>4 pts) and 2) clearly distinguishable phase peaks (>50 mrad); unresolvable parameters are characterized by: 1) very low slopes and phase peaks (m,τ,c) and 2) phase maxima outside the measuring range (m,τ). First, minimize the number of constant parameters. Later, use insight gained from a set of spectra to determine which of the most ambiguous parameters to hold constant.

4) There will always be a better fitting model. When the data is fit adequately, stop inverting and start interpreting!

General Program Conventions and Notes

SYSTEM: Hewlett Packard (HP) 9845A/ 64K byte memory. Refer questions regarding HP BASIC syntax to the HP 9845 manuals: "Operating and programing" and "Graphics programing techniques".

UNITS: Unless otherwise noted, assume the following units: meters, milliradians, ohm-meters, volts, amperes.

DEFAULTS: All questions have default values preassigned. Pushing CONT (continue) executes these defaults. The default response is indicated by underlining.

VARIABLES: Variable names are usually mnemonics related to their function. In general, a given variable's definition is ubiquitous to the whole program. This program's most important variables are defined in table 2.

PROGRAM HEADERS: Definitions of parameters passed to each subroutine are found in its heading. Array dimensions are indicated in parenthesis. Descriptive line labels provide access to the head of each subroutine if this is required.

USER MODIFICATIONS: There are a few places where the user might want to set a default variable in the program. A few of the most likely modifications are described below.

Begin by pushing EDITLINE ...; finish by pushing STORE.

1) The internal printer is activated by changing the device code, Pdev, in line 390 from 16 to 0. Activating lines 2100 and 2250 will keep intermediate parameters from being printed out when Pdev=0.

2) A double logarithmic phase scale (to plot both positive and negative phases) is activated by changing the variable Log$ in line 400 from "YES" to "DBL".

3) Numeric, rather than analytic derivatives, will be calculated if the variable Analytic$ in line 1080 is changed from "YES" to "NO".

4) The default input device variable Device$ can be changed in line 6240 from ":C" (disk) to ":T14" (left-hand tape drive), for instance.

5) The log-log plot's base and frequency range are easily modified in lines 750 and 5300. These default to: base=10, f_{min}=-3 and f_{max}=5. Either plot can be suppressed by deactivating (with "!") the appropriate line (see lines 750, 770, 5300 or 5310). Note that if subroutine Crplt is not called before Argplt, the plotter must be initialized. Change line 8910 to: PLOTTER IS 13, "GRAPHICS"-- to accomplish this.

6) Occasionally, weak dispersions are incorrectly scaled in the final plot. This condition exists when a large DC offset is obvious between the raw and inverted normalized amplitudes. To replot the data correctly, temporarily modify Rk in line 7070 from 1 to R0.

Table 2--List of major variables

Below, the most important common, passed and calculated variables or arrays are defined. Default values are indicated in parentheses.

```
I=1,Nf
J=1         -raw data
 =1+x       -model x               if model=1
 =2+model   -inverted data           then J=2   -inverted
 =3+model   -residual
K=1,2*Nf
  1st half: amplitudes;   2nd half: phases
```

COMMON:

Nf -number of frequencies		(41)
model -number of Cole-Cole dispersions modeled		(2)
RO -DC apparent resistivity; initially set to Amp(1,1)		
F(1,I)	-frequency	
Amp(J,I)	-amplitude	
Phz(J,I)	-phase	
Constant(*)	-constant parameters 1=variable; 0=constant	(1)

PASSED:

Y(1,K)	-input or raw data	
Yfit(1,K)	-output or best fit	
Sigy(1,K)	-standard deviations	
Ywt(1,K)	-data weights	(1)
P(3,model)	Cole-Cole parameters	

CALCULATED:

A(I,model*3)	Derivative matrix
Dp(model*3,model*3)	Inverted incremental parameters

User Instructions: Program I/O

The following example uses the theoretically-calculated response of two RC circuits in series. Both circuits have a Warburg-like $(\sim\sqrt{f})$ capacitance. Below, a typical program run (listed in appendix A) is broken-down into 16 steps. System commands are capitalized. Commonly occurring program variables are offset by a double hyphen (--X).

A. Turn on the machine and insert the program tape into the right-hand (T-15) tape drive. Insert the data tape into the left-hand (T-14) tape drive.

B. Type GET "CRDBL", and press EXECUTE.

C. When the program is loaded, press RUN. Several prompting statements will appear successively on the screen and the user should type the appropriate responses as follows:

1) Today's date? (Mo/Da/Yr) End by pressing CONT. --Date$

2) Read from? : --Iofile$
Choose the proper subroutine to read in the data, CONT.

3) File name? : Enter the data file name, CONT. --Fnm$

4) Do you want to PLOT or ... CONT? : This allows the data to be previewed for the purpose of weighting points and choosing the initial inversion model. After generating the plot, the program waits three seconds and will continue automatically unless PAUSE is executed.

5) What do you want to invert? Type one number, CONT. --Iop

6) Choose one method of weighting the data? Four modes are possible:

Mode 1: Instrumental- use when measurement standard deviations are known.

Mode 2: Statistical- assumes a statistical relationship where errors are proportional to magnitude.

Mode 3: Selective- allows arbitrary weights to be assigned to each data point. In turn, amplitude and phase data are displayed, five points per line. A row of "1"'s, also offset in groups of five, will be displayed at the bottom of the screen. These are default weights. Use the <-- and --> keys to position the cursor under the data

point that is to be reweighted. CONT when finished.
Mode 4: None- assigns equal weights of "1" to each data point (default).

7) At this point, a tabulated summary of the data is printed to the device specified by Pdev (line 390). In this example, frequency (Hz), amplitude (ohm-m) and phase (mrad) are printed with their respective weights. Note that these are values of phase shift whereas phase lag is plotted.

8) How many dispersions ... do you want to model? --model
Refer to "Procedures". End with CONT.

9) Input M,T,C: --P(*)
Requests initial Cole-Cole parameters, CONT.

10) Are these correct? : Requests confirmation that parameters have been entered correctly. Typing NO branches back to step 9. CONT.

11) Do you want to hold one parameter constant?: Allows one parameter of current model to be held invariant during inversion. CONT.

12) "Which one?:" Requests parameter to be held constant. This loop must be passed through for each parameter held constant. CONT.

At this point, the computer starts inverting the data. If, for any reason, the inversion is terminated prematurely by pushing special function key "k8" (ie. wildly fluctuating or unrealistic parameters), the program should jump to step 13 (ie, statistics will be calculated for the last iteration). Unfortunately, if k8 is pushed during an intermediate calculation, the statistics and inverted plot might be invalid. Avoid this situation by pushing k8 immediately following the display of an intermediate result.

13) Intermediate results are printed on the device specified by Pdev following each iteration. This output consists of:

a) Fixed parameters, if any.
b) Free Cole-Cole parameters for each iteration. The column headings are:

It: iteration -iteration 0 lists initial parameters.
Lamda(λ): Marquardt stabilization factor.

Rchisqr: Reduced chi-square.
M,T,C: Cole-Cole parameters m, τ and c.

c) %Sigma= parameter standard deviations.

d) Correlation matrix. If inversion option (Iop) 1 or 3 was chosen (amplitude or both, respectively), an additional parameter (DC resistivity) is used to find parameter correlations. Its relationship to other parameters is found in the matrix's last row.

14) Type <u>YES</u> to plot your results. Type NO to change parameters, CONT. Program loops back to step 8 if this parameterization is not satisfactory. For the novice using this program, 'bad' data points are the cause of poor parameterizations more often than are unresolvable dispersions. If this is the case, start over from the beginning by pushing STOP twice and then RUN.

15) Which dispersion do you want to remove? Enter the dispersion or model number, CONT. This step is skipped if 0 is typed.

16) The final plot is automatically displayed on the screen. Type DUMP GRAPHICS and press EXECUTE, to get a hard copy.

WHAT IS PLOTTED:

Normalized amplitude (Z/R_o)	(0)
Inverted amplitude	- -- - -- -
Log_{10} phase lag	(+ or - ; # if w_i=0)
Inverted dispersions	-----------
Residual phase	- - - - - -

$R_{dc}=R_o$: Calculated zero frequency apparent resistivity

A real-imaginary axis diagram of the data is made in the lower left-hand corner. Each curve is normalized by R_o. The axes are negative upwards and to the left and are graduated at 10% increments and labeled every .5 units. Only the inverted dispersion is plotted (see USER MODIFICATIONS). Points weighted to zero are skipped. Plus (+) symbols mark every fourth frequency point.

REFERENCES CITED

Bevington, P.H., 1969, Data reduction and error analysis for the physical sciences: New York, McGraw-Hill, 330 p.

Cole, K.S. and Cole, R.H., 1941, Dispersion and absorption in dielectrics: Journal of Chemical Physics, V.9, p. 341.

Hallof, P.G. and Pelton, W.H., 1980, The removal of inductive coupling effects from spectral IP data: Paper presented at the 50th Annual SEG International Meeting, Houston, Nov. 16-20, 25 p.

Major, J. and Silic, J., 1981, Restrictions on the use of Cole-Cole dispersion models in complex resistivity interpretation: Geophysics, V.46(6), p. 916-931.

Marquardt, D.W., 1963, An algorithm for least-squares estimation of nonlinear parameters: Journal of Industrial and Applied Mathamatics, V.2(2), p. 431-441.

Pelton, W.H., 1977, Interpretation of induced polarization and resistivity data: Univ. of Utah, Ph. D. Thesis, 255 p.

Pelton, W.H., Ward, S.H., Hallof, P.G., Sill, W.R., and Nelson, P.H., 1978, Mineral discrimination and removal of inductive coupling with multifrequency I.P.: Geophysics, V.43(3), p. 588-609.

Washburne, J.C., 1982, Parameterization of spectral induced polarization data and in situ and laboratory spectral induced polarization measurements: West Shasta copper-zinc district, Shasta, CA.: Colorado School of Mines, M.S. Thesis, 443 p.

Webster, S.S., 1980, Implications of a spectral IP survey at Elura, in D.W. Emerson, ed., The geophysics of the Elura orebody, Cobar, New South Wales: Bulletin of the Australian Society of Exploration Geophysicists, V.11(4), p. 201-207.

Zonge, K.L., 1972, Electrical properties of rocks as applied to geophysical prospecting: University of Arizona, Ph. D. thesis, 153 p.

Appendix A

Example Program Listing

The following listing was made by assigning PRINT ALL IS 0 and depressing the PRINT ALL key. Each step is described in the User Instructions section of this report. Under normal operating conditions, only the two output sections indicated by a double line in the right-hand margin will be output to the thermal printer when Pdev=0.

```
1) Enter todays date(m/d/y):
   1/31/82
2) Read data from TEST, In1, or In2?
   TEST
3) TEST:Filename?
   mRCser
4) Do you want to PLOT and preview the spectrum or CONT?
   PLOT


   ******************************
        1) Just Amplitude
        2) Just Phase
        3) Both Amp and Phz
   ******************************
5) What do you want to invert?
   3

   ******************************
        1) Instrumental (1/sig^2)
        2) Statistical    (1/Y)
        3) Selective      (1-0)
        4) None            (1)
   ******************************
6) Choose one method of weighting the data?
   3

   AMPLITUDE

      1.97E+00   1.95E+00   1.91E+00   1.85E+00   1.77E+00
      1.68E+00   1.60E+00   1.54E+00   1.49E+00   1.43E+00
      1.36E+00   1.28E+00   1.19E+00   1.12E+00   1.07E+00
      1.04E+00   1.02E+00

   Edit the weighting function; then CONT.
    1, 1, 1, 1, 1| 1, 1, 1, 1, 1| 1, 1, 1, 1, 1| 1, 1

   PHASE

     -1.41E+01  -2.31E+01  -3.57E+01  -5.01E+01  -6.14E+01
     -6.41E+01  -5.91E+01  -5.33E+01  -5.35E+01  -6.18E+01
     -7.54E+01  -8.63E+01  -8.51E+01  -7.08E+01  -5.10E+01
     -3.32E+01  -2.03E+01

   Edit the weighting function; then CONT.
    1, 1, 1, 1, 1| 1, 0, 1, 1, 1| 1, 1, 1, 1, 1| 1, 1
```

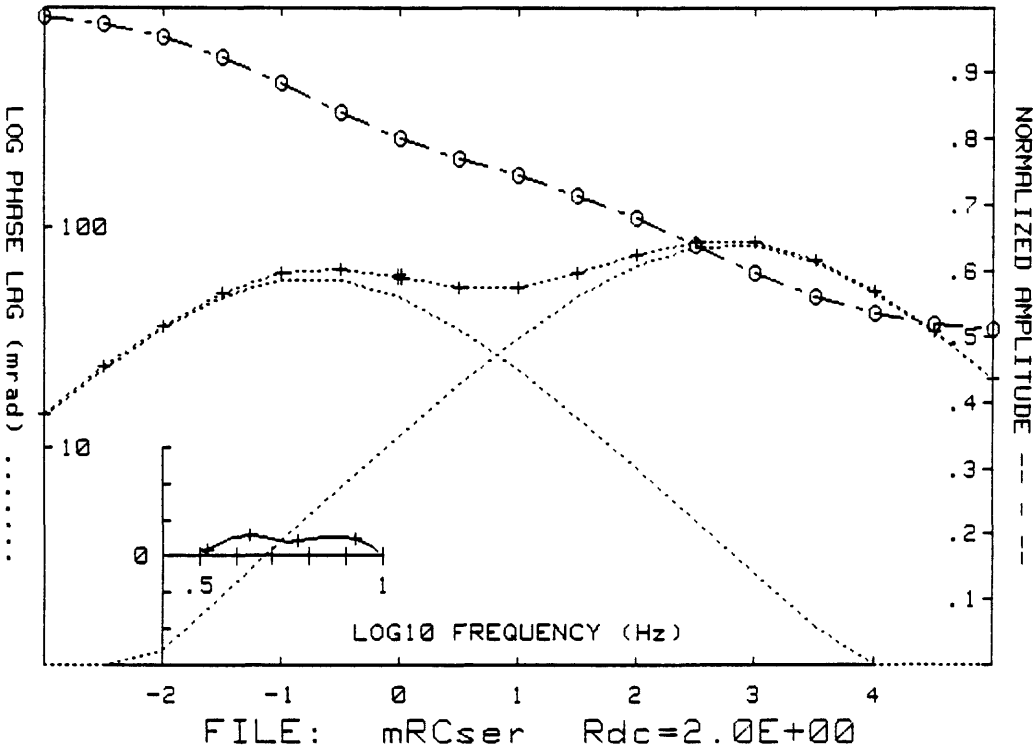
7)

FREQUENCY	AMP	WEIGHT	PHZ	WEIGHT
1.00E-03	1.97E+00	1.00	-1.41E+01	1.00
3.16E-03	1.95E+00	1.00	-2.31E+01	1.00
1.00E-02	1.91E+00	1.00	-3.57E+01	1.00
3.16E-02	1.85E+00	1.00	-5.01E+01	1.00
1.00E-01	1.77E+00	1.00	-6.14E+01	1.00
3.16E-01	1.68E+00	1.00	-6.41E+01	1.00
1.00E+00	1.60E+00	1.00	-5.91E+01	0.00
3.16E+00	1.54E+00	1.00	-5.33E+01	1.00
1.00E+01	1.49E+00	1.00	-5.35E+01	1.00
3.16E+01	1.43E+00	1.00	-6.18E+01	1.00
1.00E+02	1.36E+00	1.00	-7.54E+01	1.00
3.16E+02	1.28E+00	1.00	-8.63E+01	1.00
1.00E+03	1.19E+00	1.00	-8.51E+01	1.00
3.16E+03	1.12E+00	1.00	-7.08E+01	1.00
1.00E+04	1.07E+00	1.00	-5.10E+01	1.00
3.16E+04	1.04E+00	1.00	-3.32E+01	1.00
1.00E+05	1.02E+00	1.00	-2.03E+01	1.00

```
8)  How many dispersions?(default:2, max:3)
    2
    * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
    MODEL                    1
9)  INPUT M,T,C:
    .5,1,.5
     CHARGEABILITY= .500  TIME CONSTANT=1.00E+00  FREQUENCY DEPENDENCE= .500
    * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
10) Are these correct?(YES or NO)
    YES
11) Do you want to hold one parameter constant?(YES or NO)
    YES
    Enter one parameter you want to hold constant(1,2,or3):
12) 3
    Do you want to hold one parameter constant?(YES or NO)
    NO
    MODEL                    2
    INPUT M,T,C:
    .5,.001,.3
     CHARGEABILITY= .500  TIME CONSTANT=1.00E-03  FREQUENCY DEPENDENCE= .300
    * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
    Are these correct?(YES or NO)
    YES
    Do you want to hold one parameter constant?(YES or NO)
    NO
    Don't bother me, I'm computing . . . Press k8 to terminate inversion
```

```
13) CONSTANT PARAMETERS:C1: 5.00E-01
    It  Lamda   Rchisq      M1         T1         M2         T2         C2
     0  0E+00  3.96E-01  5.00E-01  1.00E+00  5.00E-01  1.00E-03  3.00E-01
     1  1E-01  1.03E-02  2.25E-01  9.55E-01  3.32E-01  5.99E-04  4.35E-01
     2  1E-02  2.52E-04  2.50E-01  1.24E+00  3.30E-01  3.69E-04  4.99E-01
     3  1E-03  4.41E-08  2.48E-01  1.20E+00  3.35E-01  4.00E-04  5.00E-01
    %Sigma=              3.86E-02  3.19E-02  3.20E-02  3.77E-02  2.92E-02
    CORRELATION MATRIX
                      1  1.000
                      2  -.732     1.000
                      3  -.650      .588     1.000
                      4  -.710      .616      .714     1.000
                      5   .587     -.451     -.508     -.677     1.000
                      6   .219     -.071      .038     -.009      .057     1.000

    The reduced chi square has converged or diverged.
    Type YES to plot your results. Type NO to change parameters.
14) PLOT?
    YES
    Which dispersion do you want to remove(0,1,2,3)?
15) 0
16) DUMP GRAPHICS
```



Appendix B

Source Program Listing

```
10    ! ************************************************************
20    !
30    ! PROGRAM "CRDBL": SPECTRAL IP INVERSION AND PLOTTING
40    !
50    !   SYSTEM: HP 9845A
60    !   AUTHOR: JIM WASHBURNE
70    !   ADDRESS: U.S. DEPARTMENT OF THE INTERIOR
80    !            GEOLOGICAL SURVEY
90    !            MAIL STOP 964
100   !            BOX 25046, DENVER FEDERAL CENTER
110   !            DENVER, CO   80225
120   !   LAST REVISION: 1/31/82
130   !
140   ! SUBROUTINE ACCESS ADDRESSES:
150   !       Crinv          10- 790
160   !       Curfit        800-4000
170   !         (Dy,Norm,Output)
180   !       Colem        4010-5070
190   !       Weight       5080-6120
200   !       Readin       6130-6520
210   !         (Test,In1,In2)
220   !       Crplt        6530-7290
230   !       Plots        7300-8450
240   !         (Rescale,Linplt,Xaxlbl,Yaxlbl,Title)
250   !       Argplt       8460-9210
260   ! ************************************************************
270 Crinv:!
280  OPTION BASE 1
290  COM Nf,Model,Pdev,Iop,Log$[3],Fnm$[6],Date$[8]
300  COM SHORT R0,F(1,41),Amp(6,41),Phz(6,41),Constant(3,3)
310  SHORT Y(1,82),Yfit(1,82),Sigy(1,82),Ywt(1,82)
320  DEFAULT ON                              !SUPRESS NON-FATAL ERRORS
330  EXIT GRAPHICS
340  PRINTER IS 16
350  PRINT PAGE
360  MAT Amp=ZER
370  MAT Phz=ZER
380  MAT Constant=ZER
390  Pdev=16                                 !PRINTER(16 or 0)
400  Log$="YES"                            !ABSOLUTE OR DOUBLE LOG SCALE
410  INPUT "Enter todays date(m/d/y):",Date$  !("YES" or "DBL")
420  Iofile$="TEST"
430 Read: INPUT "Read data from TEST, In1, or In2?",Iofile$
440     IF Iofile$="TEST" THEN CALL Test(Y(*),Sigy(*),Ywt(*))
450 !       * * *                                    * * *
460 !       * * * User supplied input programs * * *
470 !       * * *                                    * * *
480  Nf2=Nf*2
490  REDIM F(1,Nf),Y(1,Nf2),Yfit(1,Nf2),Sigy(1,Nf2),Ywt(1,Nf2)
500 Invert: CALL Curfit(Nf2,F(*),Y(*),Ywt(*),Yfit(*))
510  IF Model=1 THEN Plot
520 Decupl: Decupl$="N"
```

```
530   Ans=0
540   INPUT "Which dispersion do you want to remove(0,1,2,3)?",Ans
550   IF Ans=0 THEN Plot
560   Decupl$="Y"
570   M=Ans+1
580   Rk=1
590   IF (Amp(M,1)>1.5) OR (Amp(M,1)<.5) THEN Rk=R0
600   Last=Model+3
610   FOR I=1 TO Nf
620      Ro=Amp(1,I)/R0*COS(Phz(1,I)/1000)
630      Io=Amp(1,I)/R0*SIN(Phz(1,I)/1000)
640      Re=Amp(M,I)/Rk*COS(Phz(M,I)/1000)
650      Im=Amp(M,I)/Rk*SIN(Phz(M,I)/1000)
660      R2=1-Re                                          !RESIDUALS
670      I2=-Im
680      Dr=I2^2+(1-R2)^2
690      Rr=(Ro*(1-R2)-Io*I2)/Dr
700      Ir=(Ro*I2+Io*(1-R2))/Dr
710      Amp(Last,I)=R0*SQR(Rr^2+Ir^2)
720      Phz(Last,I)=1000*ATN(Ir/Rr)
730   NEXT I
740   !
750  Plot: CALL Crplt(10,-3,5,Decupl$,Ywt(*))
760   !
770   CALL Argplt(Ywt(*))
780   STOP
790   END
```

```
800  ! ***********************************************************
810  !
820 Curfit:!                                                 11/81
830  !     A LEAST-SQUARES FIT TO A NON-LINEAR FUNCTION WITH A
840  !           LINEARIZATION OF THE FITTING FUNCTION
850  ! AFTER:P.R.BEVINGTON,1969,DATA REDUCTION AND ERROR ANALYSIS
.
860  ! ***********************************************************
870  !
880  SUB Curfit(Nf2,SHORT X(*),Y(*),Sigy(*),Yfit(*))
890  !
900  !   X(*): independent variable           (1,Nf)
910  !   Y(*): dependent variable             (1,Nf2)
920  !   Sigy(*): standard deviations of Y    (1,Nf2)
930  !   Yfit(*): output, best fit to Y       (1,Nf2)
940  !
950  OPTION BASE 1
960  COM Nf,Model,Pdev,Iop,Log$[3],Fnm$[6],Date$[8]
970  COM SHORT R0,F(*),Amp(*),Phz(*),Constant(*)
980  SHORT P(3,3),Pp(3,3),Ro
990  SHORT Dy(1,Nf2),A(Nf2,10),At(10,Nf2),W(10,Nf2)
1000 SHORT One(10,1),Beta(1,10),Dp(1,10),Sigp(1,10)
1010 SHORT Alpha(10,10),Array(10,10),Anorm(10,10),Covar(10,10),Cor(
10,10),D(10,10)
1020 Initialize:!
1030 Nterms=3
1040 Lamda=.1
1050 Iteration=1
1060 Tol=1E-4
1070 IF Iop=1 THEN Tol=1E-5                       !ONLY FOR AMPLITUDE
1080 Analytic$="YES"          !ANALYTIC DERIVATIVES("YES" or "NO")
1090 P4=1                                    !INVERT R0?(0=NO, 1=YES)
1100 IF Iop=2 THEN P4=0
1110 Conv$="NO"
1120 ! Pdev=0  !REMOVE "!" FOR LISTING OF ONLY INITIAL/FINAL PARMS
1130 PRINTER IS 16
1140 MAT Yfit=(1)
1150 MAT Constant=(1)
1160 Parms: Model=2
1170 INPUT "How many dispersions?(default:2, max:3)",Model
1180 Nparms=Nparm=Model*Nterms+P4
1190 IMAGE 37("* ")
1200 IMAGE X,"CHARGEABILITY=",D.3D,2X,"TIME CONSTANT="D.DDE,2X,"FRE
QUENCY DEPENDENCE=",D.3D
1210 PRINT USING 1190
1220 FOR M=1 TO Model
1230    PRINT "MODEL ",M
1240    INPUT "INPUT M,T,C:",P(M,1),P(M,2),P(M,3)
1250    PRINT USING 1200;P(M,1),P(M,2),P(M,3)
1260    PRINT USING 1190
1270    Ans$="YES"
1280    INPUT "Are these correct?(YES or NO)",Ans$
```

```
1290     IF Ans$="NO" THEN 1230
1300     Ans$="NO"
1310     INPUT "Do you want to hold one parameter constant?(YES or
NO)",Ans$
1320     IF Ans$="NO" THEN 1370
1330     INPUT "Enter one parameter you want to hold constant(1,2,
or3):",J
1340     Constant(M,J)=0           !(0=CONSTANT, 1=VARIABLE(default))
1350     Nparm=Nparm-1
1360     GOTO 1300
1370 NEXT M
1380 Npts=Nf
1390 IF Iop=3 THEN Npts=Nf2
1400 Nfree=Npts-Nparm
1410 IF Nfree>0 THEN Calc
1420 DISP "Degrees of freedom are zero. Program has stopped."
1430 STOP
1440 Calc: ON KEY #8 GOTO Conv
1450 DISP "Don't bother me, I'm computing . . . Press k8 to termina
te inversion"
1460 REDIM P(Model,3),Pp(Model,3)
1470 Redim:REDIM A(Nf2,Nparm),At(Nparm,Nf2),W(Nparm,Nf2)
1480 REDIM One(Nparm,1),Beta(1,Nparm),Dp(1,Nparm),Sigp(1,Nparm)
1490 REDIM Alpha(Nparm,Nparm),Array(Nparm,Nparm),Anorm(Nparm,Nparm)
1500 REDIM Covar(Nparm,Nparm),Cor(Nparm,Nparm),D(Nparm,Nparm)
1510 GOSUB Ro
1520 Ro=R0
1530 IF P4=0 THEN R0=1
1540 FOR I=1 TO Nf
1550     CALL Colem(I,1,Analytic$,Yfit(1,I),Yfit(1,I+Nf),P(*),A(*))
1560 NEXT I
1570 R0=Ro
1580 GOSUB Ro
1590 Ro=R0
1600 IF P4=1 THEN R0=1
1610 GOSUB Rchisqr                              !EVALUATE INITIAL ERROR
1620 R0=Ro
1630 Rchisq0=Rchisq
1640 Zero:      MAT Beta=ZER                          !INVERSION
1650            MAT Alpha=ZER
1660            MAT One=(1)
1670 IF Conv$="YES" THEN Beta
1680 IF Analytic$="NO" THEN GOSUB Numeric      !NUMERIC DERIVATIVES
1690 Beta:      MAT Dy=Dy.Sigy
1700            MAT Beta=Dy*A
1710 Alpha:     MAT W=One*Sigy
1720            MAT At=TRN(A)
1730            MAT At=At.W
1740            MAT Alpha=At*A
1750 Norm:      CALL Norm(Nparm,Alpha(*),Anorm(*))
1760 Stabilize:MAT D=IDN
1770            MAT D=D*(Lamda)
```

```
1780            MAT Array=Anorm+D
1790 Invert:    MAT Array=INV(Array)
1800 IF Conv$="YES" THEN Stats
1810            MAT Array=Array.Anorm
1820            MAT Array=Array/Alpha
1830            MAT Dp=Beta*Array
1840 Jm=0                                    !CALCULATE NEW PARAMETERS
1850 FOR M=1 TO Model
1860  FOR J=1 TO Nterms
1870    Pp(M,J)=P(M,J)
1880    IF Constant(M,J)=0 THEN Nextj
1890    IF J=2 THEN Pp(M,J)=LGT(Pp(M,J))
1900    Jm=Jm+1
1910    Pp(M,J)=Pp(M,J)+Dp(1,Jm)
1920    IF J=2 THEN Pp(M,J)=10^Pp(M,J)
1930 Nextj:  NEXT J
1940 NEXT M
1950 Roo=Ro=R0
1960 IF P4=0 THEN R0=1
1970 IF P4=1 THEN R0=R0+Dp(1,Jm+1)
1980 FOR I=1 TO Nf
1990   CALL Colem(I,1,Analytic$,Yfit(1,I),Yfit(1,I+Nf),Pp(*),A(*))
2000 NEXT I
2010 IF P4=0 THEN R0=Ro
2020 IF P4=0 THEN GOSUB Ro
2030 Ro=R0
2040 IF P4=1 THEN R0=1
2050 GOSUB Rchisqr                                 !RE-EVALUATE ERROR
2060 R0=Ro
2070 Io:IF Iteration<>1 THEN Ckerr
2080 CALL Output(0,Model,Nterms,0,0,0,0,P(*))
2090 CALL Output(1,Model,Nterms,0,0,Rchisq0,5,P(*))
2100 ! Pdev=16  REMOVE "!" FOR LISTING OF INITIAL/FINAL PARMS ONLY
2110 Ckerr:IF (ABS(Rchisq/Rchisq0)<=Tol) OR (ABS(Rchisq0-Rchisq)<=T
ol) THEN Conv
2120        IF Rchisq<Tol/100 THEN Conv
2130 CALL Output(1,Model,Nterms,Iteration,Lamda,Rchisq,5,Pp(*))
2140 IF Lamda>=10000 THEN End
2150 Iteration=Iteration+1
2160 IF Rchisq0-Rchisq>=0 THEN GOTO Less
2170 More: Lamda=10*Lamda                       !ERROR HAS INCREASED
2180 IF P4=1 THEN R0=Roo
2190 GOTO Stabilize
2200 Less: Lamda=Lamda/10                       !ERROR HAS DECREASED
2210 MAT P=ABS(Pp)
2220 Rchisq0=Rchisq
2230 GOTO Zero
2240 Conv: Conv$="YES"                          !ERROR HAS CONVERGED
2250 ! Pdev=0   REMOVE "!" FOR LISTING OF INITIAL/FINAL PARMS ONLY
2260 CALL Output(1,Model,Nterms,Iteration,Lamda,Rchisq,5,Pp(*))
2270 Lamda=0
2280 MAT P=Pp
```

```
2290 IF P4=1 THEN GOTO Zero
2300 M=Model+2
2310 IF Model=1 THEN M=2
2320 Ct=0
2330 Dsum=0
2340 Shift=0
2350 IF Iop=2 THEN Shift=Nf
2360 FOR I=1 TO Nf
2370    IF Sigy(1,I+Shift)=0 THEN 2400
2380    Dsum=Dsum+LOG(Amp(1,I)/Amp(M,I))
2390    Ct=Ct+1
2400 NEXT I
2410 R0=EXP(Dsum/Ct)                          !AVERAGE R0 @ CONVERGENCE
2420 GOTO Zero
2430 Stats: REDIM One(1,Nparm)
2440        MAT One=(1)
2450        MAT Covar=Array*(Rchisq)
2460        MAT D=IDN
2470        MAT D=D.Covar
2480        MAT D=SQR(D)
2490        MAT Sigp=One*D
2500        MAT Sigp=Sigp*(100)
2510        CALL Norm(Nparm,Covar(*),Cor(*))
2520 CALL Output(2,Model,Nterms,0,0,0,22,Sigp(*)) !UNCERTAINTIES
2530 OUTPUT Pdev USING "18A";"CORRELATION MATRIX"
2540 CALL Output(3,Nparm,Nparm,0,0,0,19,Cor(*))
2550 GOTO End
```

```
2560 Numeric: DEF FNAsnh(SHORT X)=LOG(X+(X^2+1)^.5)
2570 SHORT Yfita,Yfitp
2580 Ro=R0
2590 IF P4=0 THEN R0=1
2600 FOR I=1 TO Nf                              !NUMERICAL DERIVATIVES
2610  Jm=0
2620  New=I+Nf
2630  FOR M=1 TO Model
2640   FOR J=1 TO Nterms
2650    IF Constant(M,J)=0 THEN 2750
2660    Jm=Jm+1
2670    Psave=P(M,J)
2680    P(M,J)=P(M,J)*1.1
2690    CALL Colem(I,1,"NO",Yfita,Yfitp,P(*),A(*))
2700    Dp=.1*Psave
2710    IF J=2 THEN Dp=.04139                   !LGT TRANSFORM
2720    IF Iop<>2 THEN A(I,Jm)=LOG(Yfita/Yfit(1,I))/Dp
2730    IF Iop<>1 THEN A(New,Jm)=(FNAsnh(Yfitp)-FNAsnh(Yfit(1,New))
)/Dp
2740    P(M,J)=Psave
2750   NEXT J
2760  NEXT M
2770  IF P4=0 THEN 2830
2780  Ro=R0
2790  R0=1.1*R0
2800  CALL Colem(I,1,"NO",Yfita,Yfitp,P(*),A(*))
2810  A(I,Jm+1)=LOG(Yfita/Yfit(1,I))*10/Ro
2820  R0=Ro
2830 NEXT I
2840 R0=Ro
2850 RETURN
2860 Ro: Shift=0
2870     IF Iop=2 THEN Shift=Nf
2880     FOR I=1 TO Nf
2890      Fix=Yfit(1,I)
2900      IF P4=1 THEN Fix=1
2910      R0=Amp(1,I)/Fix
2920      IF Sigy(1,I+Shift)<>0 THEN 2940
2930     NEXT I
2940  RETURN
2950 Rchisqr: DIM Dy2(1,2*Nf)                        !ERROR CALCULATION
2960          CALL Dy(Y(*),Yfit(*),Dy(*))
2970          MAT Dy2=Dy.Sigy
2980          MAT Dy2=Dy2.Dy2
2990          Chisq=SUM(Dy2)
3000          Rchisq=Chisq/Nfree
3010          RETURN
3020 End: Ans$="YES"
3030 PRINT "The reduced chi square has converged or diverged."
3040 PRINT "Type YES to plot your results. Type NO to change para
meters."
3050 INPUT "PLOT?",Ans$
3060 IF Ans$="NO" THEN Initialize
3070 SUBEND
```

```
3080 ! ************************************************************
3090 Dy:!                                                       11/81
3100  SUB Dy(SHORT Y(*),Yfit(*),Dy(*))
3110 !
3120 !    Y(*): Observed data
3130 !    Yfit(*): Calculated data
3140 !    Dy(*): Difference vector
3150 !
3160 OPTION BASE 1
3170 COM Nf,M,Pdev,Iop,Log$[3],Fnm$[6],Date$[8]
3180 COM SHORT R0,F(*),Amp(*),Phz(*),Constant(*)
3190 ON KEY #8 GOTO End
3200 DEF FNAsnh(SHORT X)=LOG(X+SQR(X^2+1))   !ARC-HYPERBOLIC SINE
3210 Amp: IF Iop=2 THEN Phz
3220 FOR I=1 TO Nf
3230    Dy(1,I)=LOG(Y(1,I)/(R0*Yfit(1,I)))
3240 NEXT I
3250 Phz:IF (Iop=1) OR (Iop=4) THEN End
3260 FOR I=1 TO Nf
3270    Dy(1,I+Nf)=FNAsnh(Y(1,I+Nf))-FNAsnh(Yfit(1,I+Nf))
3280 NEXT I
3290 End:SUBEND
3300 ! ************************************************************
3310 Norm:!                                        !NORMALIZATION  11/81
3320 SUB Norm(N,SHORT X(*),Y(*))
3330 !
3340 !    N: Dimensions of X and Y       NOTE: There may be problems
3350 !    X(*): Input array              if Xii gets 'close' to zero
3360 !    Y(*): Output array
3370 !
3380 OPTION BASE 1
3390 SHORT I(N,N),D(N,N)
3400 ON KEY #8 GOTO End
3410     MAT Y=(1)
3420     MAT I=IDN
3430     MAT D=(1)/X
3440     MAT D=I.D
3450     MAT I=D*Y
3460     MAT Y=I*D
3470     MAT Y=ABS(Y)
3480     MAT Y=SQR(Y)
3490     MAT Y=Y.X
3500 End: SUBEND
```

```
3510 ! ***********************************************************
3520 Output:!                                    INVERSION OUTPUT  11/81
3530 SUB Output(Form,Ix,Jx,K1,K2,K3,Tab1,SHORT X(*))
3540 !
3550 !    Form: format type
3560 !    Ix: first index limit
3570 !    Jx: second index limit
3580 !    K1: \
3590 !    K2:   > constants
3600 !    K3: /
3610 !    Tab1: TAB argument
3620 !    X(*): output matrix   (Ix,Jx)
3630 !
3640 OPTION BASE 1
3650 COM Nf,Model,Pdev,Iop,Log$[3],Fnm$[6],Date$[8]
3660 COM SHORT R0,F(*),Amp(*),Phz(*),Constant(*)
3670 DIM Header$[30],List$[94],Clist$[80],Col$(9)
3680 ON KEY #8 GOTO End
3690 DATA "M1  ","T1  ","C1  ","M2  ","T2  ","C2  ","M3  ","
T3  ","C3  "
3700 MAT READ Col$
3710 PRINTER IS Pdev,WIDTH(94)
3720 Ji=0
3730 FOR I=1 TO Ix
3740  IF Form=3 THEN Jx=I
3750  FOR J=1 TO Jx
3760   IF Form<>3 THEN 3790
3770    OUTPUT Plist$ USING "#,MD.DDD,4X";X(I,J)
3780    GOTO Sum
3790    IF (Form<>0) OR (Constant(I,J)<>0) THEN 3830
3800    OUTPUT Alist$ USING "#,X,4A,A";Col$((I-1)*3+J),":"
3810    OUTPUT Plist$ USING "#,MD.DDE,X";X(I,J)
3820    Clist$=Clist$&Alist$&Plist$
3830    IF Constant(I,J)=0 THEN 3890
3840    Ji=Ji+1
3850    IF Form=0 THEN OUTPUT Plist$ USING "#,4X,4A,4X";Col$((I-1)*
3+J)
3860    IF Form=1 THEN OUTPUT Plist$ USING "#,MD.DDE,X";X(I,J)
3870    IF Form=2 THEN OUTPUT Plist$ USING "#,MD.DDE,X";X(1,Ji)
3880 Sum: List$=List$&Plist$
3890  NEXT J
3900  IF Form=3 THEN PRINT TAB(Tab1);I;List$
3910  IF Form=3 THEN List$=""
3920 NEXT I
3930 IF Form=3 THEN End
3940 IF Form=2 THEN PRINT "%Sigma=";TAB(Tab1);List$
3950 IF Form=1 THEN OUTPUT Header$ USING "#,DE,2X,D.DDE,2X";K2,K3
3960 IF Form=1 THEN PRINT K1;TAB(Tab1);TRIM$(Header$&List$)
3970 IF Form=0 THEN PRINT "CONSTANT PARAMETERS:";TRIM$(Clist$)
3980 IF Form=0 THEN PRINT TRIM$("It   Lamda   Rchisq   "&List$)
3990 End: PRINTER IS 16,WIDTH(80)
4000 SUBEND
```

```
4010 ! ***********************************************************
4020 !
4030 Colem:!    FORWARD AND DERIVATIVE COLE-COLE ALGORITHMS  11/81
4040 !                         MULTIPLICATIVE
4050 ! ***********************************************************
4060 !
4070 SUB Colem(I,It,Deriv$,SHORT Afit,Pfit,P(*),A(*))
4080 !
4090 !    I: current index from calling program
4100 !    It: seed for output files
4110 !    Deriv$: calculate analytic derivatives?, "YES" or "NO"
4120 !    Afit: best fit amplitude
4130 !    Pfit: best fit phase
4140 !    P(*): input parameter matrix   (M,1)=M,(M,2)=T,(M,3)=C
4150 !    A(*): output derivative matrix   (I,Model)
4160 !
4170 OPTION BASE 1
4180 COM Nf,Mm,Pdev,Iop,Log$[3],Fnm$[6],Date$[8]
4190 COM SHORT R0,F(*),Amp(*),Phz(*),Constant(*)
4200 SHORT C(0:1),Q(0:1),Z(0:Mm-1,2),Pd(0:Mm*3-1,2),Dadw(Mm)
4210 P2=PI/2
4220 Nterms=3
4230 C(0)=1
4240 C(1)=0
4250 IF (Log$="DBL") AND (Mm<>1) THEN Cupl$="YES"
4260 FOR M=1 TO Mm                                !EVALUATING EQUATION
4270    It=It+1                                   !FOR Re AND Im PARTS
4280    N=M-1
4290    Sgn=-1
4300    IF (Cupl$="YES") AND (M=Mm) THEN Sgn=-Sgn
4310    W=2*PI*F(1,I)
4320    Wt=W*ABS(P(M,2))
4330    Wc=Wt^P(M,3)
4340    W2c=Wc^2
4350    A=Wc*COS(P2*P(M,3))
4360    B=Wc*SIN(P2*P(M,3))
4370    K1=A+W2c
4380    D1=1+2*A+W2c
4390    Re=Sgn*P(M,1)*K1/D1
4400    Z(N,1)=1+Re
4410    Im=Sgn*B*P(M,1)/D1
4420    Z(N,2)=Im
4430    !                                         FINDING Amp AND Phz
4440    Amp(It,I)=Afit=R0*(Z(N,1)^2+Z(N,2)^2)^.5
4450    Phz(It,I)=Pfit=1000*ATN(Z(N,2)/Z(N,1))      !Phz in mrad's
4460    Cr=C(0)
4470    Ci=C(1)
4480    C(0)=Cr*Z(N,1)-Ci*Z(N,2)                     !COMPLEX PRODUCTS
4490    C(1)=Cr*Z(N,2)+Ci*Z(N,1)
4500    !
4510    IF Deriv$="NO" THEN 4630               !PARTIAL DIFFERENTIALS
4520    Inc=N*Nterms
4530    Kc=Sgn*P(M,1)/D1^2
4540    Kt=Kc*P(M,3)/P(M,2)
```

```
4550    Pd(Inc,1)=Sgn*K1/D1
4560    Pd(Inc,2)=Sgn*B/D1
4570    Ret=2*W2c+A*(1+W2c)
4580    Pd(1+Inc,1)=Kt*Ret
4590    Imt=B*(1-W2c)
4600    Pd(1+Inc,2)=Kt*Imt
4610    Pd(2+Inc,1)=Kc*(Ret*LOG(Wt)-P2*Imt)
4620    Pd(2+Inc,2)=Kc*(P2*Ret+Imt*LOG(Wt))
4630 NEXT M
4640 !                                                  !Amp AND Phz FOR
4650 Tan=C(1)/C(0)                                      !TOTAL RELAXATION
4660 Rsave=C(0)
4670 IF Mm=1 THEN 4740
4680  Pextra=0
4690  IF C(0)<0 THEN Pextra=Sgn*3141.59                !SECOND&THIRD QUAD
4700 It=It+1
4710 Amp(It,I)=Afit=R0*(C(0)^2+C(1)^2)^.5
4720 Phz(It,I)=Pfit=1000*ATN(Tan)+Pextra
4730 !
4740 IF Deriv$="NO" THEN End
4750 D3=R0*SQR(1/(1+Tan^2))
4760 D4=1000/(Rsave*(1+Tan^2))
4770 D3=D3/Afit                                         !Log TRANSFORM
4780 D4=D4/SQR(1+Pfit^2)                                !Asinh TRANSFORM
4790 New=I+Nf
4800 Jm=0
4810 FOR M=1 TO Mm                                      !DERIVATIVES
4820    C(0)=1
4830    C(1)=0
4840  FOR K=1 TO Mm
4850    IF M=K THEN Skip
4860    Cr=C(0)
4870    Ci=C(1)
4880    C(0)=Cr*Z(K-1,1)-Ci*Z(K-1,2)                   !PARTIAL PRODUCTS
4890    C(1)=Cr*Z(K-1,2)+Ci*Z(K-1,1)
4900 Skip: NEXT K
4910  FOR J=1 TO Nterms
4920    N=M-1
4930    Id=J-1+N*Nterms
4940    Q(0)=C(0)*Pd(Id,1)-C(1)*Pd(Id,2)              !PARTIAL DIFFERENTIAL
4950    Q(1)=C(0)*Pd(Id,2)+C(1)*Pd(Id,1)
4960    IF Constant(M,J)=0 THEN 5020
4970    Jm=Jm+1
4980    A(I,Jm)=D3*(Q(0)+Q(1)*Tan)
4990    IF J=2 THEN A(I,Jm)=2.3025*P(M,2)*A(I,Jm)
5000    A(New,Jm)=D4*(Q(1)-Q(0)*Tan)
5010    IF J=2 THEN A(New,Jm)=2.3025*P(M,2)*A(New,Jm)
5020  NEXT J
5030 NEXT M
5040 IF R0=1 THEN End
5050 A(I,Jm+1)=1/R0                                   !ie. Afit/(R0*Afit)
5060 A(New,Jm+1)=0
5070 End: SUBEND
```

```
5080 ! ************************************************************
5090 !
5100 Weight:!    DATA WEIGHTING                                11/81
5110 !
5120 ! ************************************************************
5130 !
5140 SUB Weight(SHORT Y(*),Sigy(*),Ywt(*))
5150 !
5160 !   Y(*): input data vector                     (1,2*Nf)
5170 !   Sigy(*): standard deviation vector          (1,2*Nf)
5180 !   Ywt(*): output weight vector                (1,2*Nf)
5190 !
5200 OPTION BASE 1
5210 COM Nf,Model,Pdev,Iop,Log$[3],Fnm$[6],Date$[8]
5220 COM SHORT R0,F(*),Amp(*),Phz(*),Constant(*)
5230 DIM List$[123],Punc$[1],Comma$[1],Break$[1]
5240 MAT Ywt=(1)
5250 R0=Amp(1,1)
5260 Ans$="CONT"
5270 INPUT "Do you want to PLOT and preview the spectrum or CONT?
",Ans$
5280 IF Ans$="CONT" THEN Inv_op
5290 Preview: Model=0
5300 CALL Crplt(10,-3,5,"N",Ywt(*))
5310 CALL Argplt(Ywt(*))
5320 WAIT 3000
5330 EXIT GRAPHICS
5340 Inv_op: Iop=3
5350 PRINT "******************************"
5360 PRINT "      1) Just Amplitude"
5370 PRINT "      2) Just Phase"
5380 PRINT "      3) Both Amp and Phz"
5390 PRINT "******************************"
5400 INPUT "What do you want to invert?",Iop
5410 PRINT
5420 Wt_op: Wop=4
5430 PRINT "******************************"
5440 PRINT "      1) Instrumental (1/sig^2)"
5450 PRINT "      2) Statistical    (1/Y)"
5460 PRINT "      3) Selective      (1-0)"
5470 PRINT "      4) None            (1)"
5480 PRINT "******************************"
5490 INPUT "Choose one method of weighting the data?",Wop
5500 MAT Ywt=ZER
5510 PRINT PAGE
5520 Alist: IF Iop=2 THEN Plist
5530     Plist$="AMPLITUDE"
5540     Shift=0
5550     GOSUB Wt
5560 Plist: IF Iop=1 THEN End
5570     Plist$="PHASE"
5580     Shift=Nf
5590     GOSUB Wt
```

```
5600 GOTO End
5610 Wt: ON Wop GOTO Inst,Stat,Sel,None
5620 Inst: !
5630    FOR I=1 TO Nf
5640    IF Sigy=0 THEN Sigy=1
5650    Ywt(1,I+Shift)=1/Sigy^2
5660    NEXT I
5670 RETURN
5680 Stat: !
5690    FOR I=1 TO Nf
5700    Y=Y(1,I+Shift)
5710    IF Y=0 THEN Y=1
5720    Ywt(1,I+Shift)=1/ABS(Y)
5730    NEXT I
5740 RETURN
5750 Sel: !
5760    GOSUB List
5770    Comma$=","
5780    Break$="|"
5790    FOR I=1 TO Nf-1
5800    Ct=Ct+1
5810    Punc$=Comma$
5820    IF Ct=5 THEN Punc$=Break$
5830    IF Ct=5 THEN Ct=0
5840    OUTPUT Plist$ USING "#,3A";" "&"1"&Punc$
5850    List$=List$&Plist$
5860    NEXT I
5870    List$=List$&" 1 "
5880 EDIT "Edit the weighting function; then CONT.",List$
5890    FOR I=1 TO Nf
5900    Ywt(1,I+Shift)=VAL(List$[1,2])
5910    List$=List$[4]
5920    NEXT I
5930 RETURN
5940 None: !
5950    FOR I=1 TO Nf
5960    Ywt(1,I+Shift)=1
5970    NEXT I
5980 RETURN
5990 List: Ct=0
6000    PRINT LIN(1),Plist$,LIN(1)
6010    FOR I=1 TO Nf
6020    Ct=Ct+1
6030    OUTPUT Plist$ USING "#,2X,MD.DDE";Y(1,I+Shift)
6040    List$=List$&Plist$
6050    IF I=Nf THEN Print
6060    IF Ct<>5 THEN 6100
6070 Print: PRINT List$
6080    List$=""
6090    Ct=0
6100    NEXT I
6110 RETURN
6120 End: SUBEND
```

```
6130 Readin:! ****************************************************
6140 !
6150 Test:!    FORWARD MODELED DATA I/O                         11/81
6160 !
6170 ! ******************************************************
6180 !
6190 SUB Test(SHORT Y(*),Sigy(*),Ywt(*))
6200 !
6210 OPTION BASE 1
6220 COM Nf,Model,Pdev,Iop,Log$[3],Fnm$[6],Date$[8]
6230 COM SHORT R0,F(*),Amp(*),Phz(*),Constant(*)
6240 Device$=":C"                                   !REMOVABLE DISK
6250 Input: INPUT "TEST:Filename?",Fnm$
6260 ASSIGN #1 TO Fnm$&Device$
6270 I=0
6280 Readloop: ON END #1 GOTO Reform
6290    I=I+1
6300    READ #1;F(1,I),Amp(1,I),Phz(1,I)
6310    Nf=I
6320 GOTO Readloop
6330 Reform: FOR I=1 TO Nf
6340         Y(1,I)=Amp(1,I)
6350         Y(1,I+Nf)=Phz(1,I)
6360         NEXT I
6370 CALL Weight(Y(*),Sigy(*),Ywt(*))
6380 Io: PRINTER IS Pdev
6390 PRINT LIN(1),"FREQUENCY    AMP    WEIGHT      PHZ
WEIGHT"
6400 FOR I=1 TO Nf
6410    PRINT USING "D.DDE,2X,2(MD.DDE,3X,D.DD,3X)";F(1,I),Amp(1,I)
,Ywt(1,I),Phz(1,I),Ywt(1,I+Nf)
6420 NEXT I
6430 End: PRINTER IS 16
6440 SUBEND
6450 ! ******************************************************
6460 !
6470 ! USER SUPPLIED INPUT PROGRAMS
6480 !
6490 In1: SUB In1(SHORT Y(*),Sigy(*),Ywt(*))
6500 !
6510 In2: SUB In2(SHORT Y(*),Sigy(*),Ywt(*))
6520 !
```

```
6530 ! ************************************************************
6540 !
6550 Crplt:!    PLOTS SPECTRAL IP DATA                         11/81
6560 !
6570 ! ************************************************************
6580 !
6590 SUB Crplt(Base,Xmn,Xmx,Decupl$,SHORT Ywt(*))
6600 !
6610 !   Base: base system for frequency scale
6620 !   Xmn: minimum log10 frequency
6630 !   Xmx: maximum log10 frequency
6640 !   Decupl$: plot decupled phase?(Y or N)
6650 !   Ywt(*): data weights              (1,2*Nf)
6660 !
6670 OPTION BASE 1
6680 COM Nf,Model,Pdev,Iop,Log$[3],Fnm$[6],Date$[8]
6690 COM SHORT R0,F(*),Amp(*),Phz(*),Constant(*)
6700 SHORT X(1,Nf),Z(Model+3,Nf)
6710 DIM Title$[40],Sym$(Nf)[1],Blank$(Nf)[1]
6720 P2=PI/2
6730 J1=2
6740 M=Model+2
6750 IF Model=0 THEN M=1
6760 IF Model=0 THEN J1=1
6770 IF Model=1 THEN M=2
6780 IF Decupl$="Y" THEN M=M+1
6790 FOR I=1 TO Nf
6800    X(1,I)=LOG(F(1,I))/LOG(Base)
6810    Blank$(I)=""
6820    OUTPUT Sym$ USING "SD";-SGN(Phz(1,I))
6830    Sym$(I)=Sym$[1,1]
6840    IF Ywt(1,I+Nf)=0 THEN Sym$(I)="#"
6850  FOR J=1 TO M
6860    Phz=-Phz(J,I)
6870    Z(J,I)=SGN(Phz)*LGT(ABS(Phz))
6880    IF (Phz>-1) AND (Phz<1) THEN Z(J,I)=0
6890  NEXT J
6900 NEXT I
6910 PLOTTER IS 13,"GRAPHICS"
6920 GRAPHICS
6930 LOCATE 5,118,20,99
6940 Phase: !
6950 Ymx=3
6960 Ymn=0
6970 Dely=1
6980 IF Log$="DBL" THEN Ymn=-3
6990 SCALE Xmn,Xmx,Ymn,Ymx
7000 AXES 1,ABS(Dely),Xmn,Ymn
7010 CALL Yaxlbl(2,-P2,3.3,Ymn,Ymx,Dely,1,Xmn,.25,"LOG PHASE LAG (m
rad) .......")
```

```
7020 FOR J=J1 TO M
7030     CALL Linplt(Nf,3,1,J,Blank$(*),X(*),Z(*))
7040 NEXT J
7050     CALL Linplt(Nf,2,1,1,Sym$(*),X(*),Z(*))
7060 !
7070 Rk=1
7080 IF (Amp(M,1)>1.2) OR (Amp(M,1)<.2) THEN Rk=R0
7090 IF Decupl$="Y" THEN M=M-1
7100 FOR I=1 TO Nf
7110     Sym$(I)="O"
7120     IF Ywt(1,I)=0 THEN Sym$(I)="#"
7130     Z(M,I)=Amp(M,I)/Rk
7140     Z(1,I)=Amp(1,I)/R0
7150 NEXT I
7160 Amplitude: !
7170 Ymx=1
7180 Ymn=0
7190 Dely=.1
7200 SCALE Xmn,Xmx,Ymn,Ymx
7210 AXES 1,Dely,Xmx,Ymx
7220 CALL Yaxlbl(8,-P2,3.3,Ymn,Ymx,Dely,1,Xmx,.25,"NORMALIZED AMPLI
TUDE -- - --")
7230     CALL Linplt(Nf,4,1,M,Blank$(*),X(*),Z(*))
7240     CALL Linplt(Nf,2,1,1,Sym$(*),X(*),Z(*))
7250 Labels: !
7260 CALL Xaxlbl(6,0,3.3,Xmn,Xmx,1,1,Ymn,.05,"LOG10 FREQUENCY (Hz)"
)
7270 OUTPUT Title$ USING "7A,X,6A,XX,4A,D.DE";" FILE: ",Fnm$,"Rdc="
,R0
7280 CALL Title((Xmx-Xmn)/2+Xmn,-.1,0,5,Title$)
7290 End: SUBEND
```

```
7300 ! ***********************************************************
7310 !
7320 Plots:!     PLOTTING SUBROUTINE PACKAGE                    11/81
7330 !
7340 ! ***********************************************************
7350 Rescale:!                       RESCALES ARGAND AXIS TO LOOK GOOD
7360 SUB Rescale(Del,Mn,Mx)
7370 Sc=1/Del
7380 IF Mx>0 THEN Mx=(INT(Sc*Mx)+SGN(Mx))*Del
7390 IF Mx<0 THEN Mx=(INT(Sc*Mx)-SGN(Mx))*Del
7400 ! IF Mx=0 THEN Mx=Del
7410 IF Mn<0 THEN Mn=INT(Sc*Mn)*Del
7420 IF Mn>0 THEN Mn=(INT(Sc*Mn)-SGN(Mn))*Del
7430 ! IF Mn=0 THEN Mn=-Del
7440 SUBEND
7450 !
7460 ! ***********************************************************
7470 Linplt:!                                               LINE PLOT
7480 SUB Linplt(Nf,Lintyp,Jx,Jy,Sym$(*),SHORT X(*),SHORT Y(*))
7490 !
7500 !    Nf: # of data points
7510 !    Lintyp: LINE TYPE argument
7520 !    Jx: constant index on X(*)
7530 !    Jy: constant index on Y(*)
7540 !    Sym$(*): plotting symbol      (Nf)
7550 !    X(*): independent variable  (Jx,Nf)
7560 !    Y(*): dependent variable    (Jy,Nf)
7570 !
7580 LORG 5
7590 Ipen=-2
7600    FOR I=1 TO Nf
7610    LINE TYPE Lintyp
7620    PLOT X(Jx,I),Y(Jy,I),Ipen
7630    IF Sym$(I)="" THEN 7680
7640    WHERE X,Y
7650    LINE TYPE 1
7660    LABEL USING "K";Sym$(I)
7670    MOVE X,Y
7680    Ipen=-1
7690    IF I=Nf THEN 7710
7700    IF X(Jx,I)>X(Jx,I+1) THEN Ipen=-2
7710    NEXT I
7720 LINE TYPE 1
7730 End:SUBEND
7740 !
7750 ! ***********************************************************
7760 !
```

```
7770 !   Lorg: LORG argument
7780 !   Ldir: LDIR argument
7790 !   Csize: CSIZE argument
7800 !   (X,Y)mn: minimum axis value
7810 !   (X,Y)mx: maximum axis value
7820 !   Del(x,y): axis STEP
7830 !   (X,Y)scale: multiplicative axis label scaling factor
7840 !   (Y,X): conjugate axis intersection
7850 !   (Y,X)shift: conjugate axis offset
7860 !   Lable$: axis label
7870 !
7880 Xaxlbl:!                                              X-AXIS LABEL
7890 SUB Xaxlbl(Lorg,Ldir,Csize,Xmn,Xmx,Delx,Xscale,Y,Yshift,Label$
)
7900 CSIZE Csize
7910 LORG Lorg
7920 Isgn=1                             !Lorg=4
7930 IF Lorg=6 THEN Isgn=-1             !Lorg=6
7940    FOR I=Xmn+Delx TO Xmx-Delx STEP Delx
7950    MOVE I,Y
7960    SETGU
7970    WHERE Xx,Yy
7980    MOVE Xx,Yy+Isgn*2.5
7990    SETUU
8000    LABEL USING "K";I*Xscale
8010    NEXT I
8020 LDIR Ldir
8030 LORG 5
8040 MOVE Xmn+(Xmx-Xmn)/2,Y-Isgn*Yshift
8050 LABEL USING "K";Label$
8060 LDIR 0
8070 SUBEND
8080 !
8090 Yaxlbl:!                                              Y-AXIS LABEL
8100 SUB Yaxlbl(Lorg,Ldir,Csize,Ymn,Ymx,Dely,Yscale,X,Xshift,Label$
)
8110 CSIZE Csize
8120 LORG Lorg
8130 Isgn=1                             !Lorg=2
8140 IF Lorg=8 THEN Isgn=-1             !Lorg=8
8150    FOR I=Ymn+Dely TO Ymx-Dely STEP Dely
8160    MOVE X,I
8170    Ii=I
8180    IF I=0 THEN Ii=1
8190    IF Lorg=2 THEN Yscale=SGN(I)*10^ABS(I)/Ii
8200    LABEL USING "X,K,X";I*Yscale
8210    NEXT I
8220 LDIR Ldir
8230 LORG 5
8240 CSIZE 3.3
8250 MOVE X-Isgn*Xshift,Ymn+(Ymx-Ymn)/2
8260 LABEL USING "K";Label$
8270 LDIR 0
8280 SUBEND
```

```
8290 !
8300 ! ***********************************************************
8310 Title:!                                              !PRINTS TITLE
8320 SUB Title(X,Y,Ldir,Csize,Label$)
8330 !
8340 !   X,Y: title position
8350 !   Ldir: LDIR argument
8360 !   Csize: CSIZE argument
8370 !   Label$: plot title
8380 !
8390 LORG 5
8400 LDIR Ldir
8410 CSIZE Csize
8420 MOVE X,Y
8430 LABEL USING "K";Label$
8440 CSIZE 3.3
8450 SUBEND
```

```
8460 ! ************************************************************
8470 !
8480 Argplt:!    REAL AND IMAGINARY ARGAND PLOT                    11/81
8490 !
8500 ! ************************************************************
8510 !
8520 SUB Argplt(SHORT Ywt(*))
8530 OPTION BASE 1
8540 COM Nf,M,Pdev,Iop,Log$[3],Fnm$[6],Date$[8]
8550 COM SHORT R0,F(*),A(*),P(*),Constant(*)
8560 SHORT Re(5,Nf),Im(5,Nf)
8570 ! Rmn=Imn=0                        ! FORCES FULL NORMALIZED SCALE
8580 ! Rmx=Imx=0                        !      REMOVE R,Iflag
8590  Rflag=Iflag=0
8600 FOR J=1 TO 1! M+1                  ! JUST PLOT OBSERVED DATA
8610 Rk=1
8620 IF (A(J,1)>1.2) OR (A(J,1)<.2) THEN Rk=R0
8630 FOR I=1 TO Nf
8640    Re(J,I)=A(J,I)/Rk*COS(P(J,I)/1000)
8650    IF (Iop<>2) AND (Ywt(1,I)=0) THEN 8710
8660    IF (Iop=2) AND (Ywt(1,I+Nf)=0) THEN 8710
8670    IF Rflag=0 THEN Rmn=Rmx=Re(J,I)
8680    Rflag=1
8690    Rmx=MAX(Rmx,Re(J,I))
8700    Rmn=MIN(Rmn,Re(J,I))
8710    Im(J,I)=A(J,I)/Rk*SIN(P(J,I)/1000)
8720    IF (Iop<>1) AND (Ywt(1,I+Nf)=0) THEN 8780
8730    IF (Iop=1) AND (Ywt(1,I)=0) THEN 8780
8740    IF Iflag=0 THEN Imn=Imx=Im(J,I)
8750    Iflag=1
8760    Imx=MAX(Imx,Im(J,I))
8770    Imn=MIN(Imn,Im(J,I))
8780 NEXT I
8790 NEXT J
8800 Delr=.1
8810  Deli=.1
8820  CALL Rescale(Delr,Rmn,Rmx)
8830  CALL Rescale(Deli,Imn,Imx)
8840  Nr=(Rmx-Rmn)/Delr
8850  Ni=(Imx-Imn)/Deli
8860  Nmx=MAX(Nr,Ni)
8870  IF Nr<>Nmx THEN Rmn=Rmn-INT((Nmx-Nr)/2)*Delr
8880  IF Ni<>Nmx THEN Imn=Imn-INT((Nmx-Ni)/2)*Deli
8890  Rmx=Rmn+Nmx*Delr
8900  Imx=Imn+Nmx*Deli
```

```
8910  PLOTTER 13 IS ON
8920  GRAPHICS
8930  LOCATE 19.1,45.43,20,46.33
8940  LORG 5
8950  SCALE Rmn,Rmx,Imx,Imn
8960  AXES Delr,Deli,Rmn,0
8970  CALL Xaxlbl(6,0,3.3,INT(2*Rmn)/2,INT(2*Rmx)/2+.5,.5,1,0,0,"")
8980  CALL Yaxlbl(8,-PI/2,3.3,INT(2*Imn)/2,INT(2*Imx)/2+.5,.5,1,Rmn
,0,"")
8990  FOR J=1 TO M+1
9000  Lintyp=3
9010  IF J=1 THEN Lintyp=1
9020  LINE TYPE Lintyp
9030  MOVE Re(J,1),Im(J,1)
9040  Ct=0
9050  FOR I=1 TO Nf
9060    Ct=Ct+1
9070    Ipen=-1
9080    IF J<>1 THEN 9190
9090    IF (Iop<>2) AND (Ywt(1,I)=0) THEN 9120
9100    IF ((Iop=2) OR (Iop=3)) AND (Ywt(1,I+Nf)=0) THEN 9120
9110    PLOT Re(J,I),Im(J,I),Ipen
9120    WHERE X,Y
9130    IF (Ct<>4) OR (J<>1) THEN 9190
9140    LINE TYPE 1
9150    LABEL USING "K";"+"
9160    LINE TYPE Lintyp
9170    MOVE X,Y
9180    Ct=0
9190  NEXT I
9200  NEXT J
9210  SUBEND
```